Definition of QM and Active Region via pdb file

The QM and active region can be defined in the pdb file as shown in the following example:

```
%qmmm
 Use_QM_InfoFromPDB true      # get the QMAtoms and ActiveAtoms definition from
 Use_Active_InfoFromPDB true  # the pdb file. The default for both is false.
                             # If InfoFromPDB is set to true, the QMAtoms or
                             # ActiveAtoms input is ignored
end
*pdbfile 0 1 ubq.pdb
```

If `Use_QM_InfoFromPDB` or `Use_Active_InfoFromPDB` is set to true, a pdb file should be used for the structural input. QM atoms are defined via 1 in the occupancy column, MM atoms via 0. Active atoms are defined via 1 in the B-factor column, non-active atoms via 0.

```
ubq.pdb:
...
ATOM    327  N   ASP A  21      29.599  18.599   9.828  0.00  0.00      P1   N
ATOM    328  HN  ASP A  21      29.168  19.310   9.279  0.00  0.00      P1   H
ATOM    329  CA  ASP A  21      30.796  19.083  10.566  0.00  1.00      P1   C
ATOM    330  HA  ASP A  21      31.577  18.340  10.448  0.00  1.00      P1   H
ATOM    331  CB  ASP A  21      31.155  20.515  10.048  2.00  1.00      P1   C
ATOM    332  HB1 ASP A  21      30.220  21.082   9.865  2.00  1.00      P1   H
ATOM    333  HB2 ASP A  21      31.754  21.064  10.801  2.00  1.00      P1   H
ATOM    334  CG  ASP A  21      31.923  20.436   8.755  1.00  1.00      P1   C
ATOM    335  OD1 ASP A  21      32.493  19.374   8.456  1.00  1.00      P1   O
ATOM    336  OD2 ASP A  21      31.838  21.402   7.968  1.00  1.00      P1   O
ATOM    337  C   ASP A  21      30.491  19.162  12.040  0.00  0.00      P1   C
ATOM    338  O   ASP A  21      29.367  19.523  12.441  0.00  0.00      P1   O
...
```

> **ⓘ Note**
>
> Contrary to the hybrid36 standard of PDB files, ORCA writes non-standard pdb files as:
>
> - atoms 1-99,999 in decimal numbers
>
> - atoms 100,000 and beyond in hexadecimal numbers, with atom 100,000 corresponding to index 186a0.
>
> This ensures a unique mapping of indices. If you want to select an atom with an index in hexadecimal space (index larger than 100,000), convert the hexadecimal number into decimals when choosing this index in the ORCA input file. Note also, that in the pdb file, counting starts from 1, while in ORCA counting starts from zero.

6.1.4 Active Atoms in Combination with Optimization, Frequency Calculation, MD

The systems of multiscale calculations can become quite large with tens and hundreds of thousands of atoms. In multiscale calculations the region of interest is often only a particular part of the system, and it is common practice to restrict the optimization to a small part of the system, which we call the active region = active part of the system. For info on how to define the active region see *QM and Active Region Definition*. Usually this active part consists of hundreds of atoms, and is defined as the QM region plus a layer around the QM region. The same definition holds for frequency calculations, in particular since after optimization non-active atoms are not at stationary points, and a frequency calculation would lead to artifacts in such a scenario. MD calculations on systems with hundreds of thousands of atoms are not problematic, but there are applications where a separation in active and non-active parts can be important (e.g. a solute in a solvent droplet, with the outer shell of the solvent frozen).

> **ⓘ Note**
>
> If no active atoms are defined, the entire system is treated as active.

> **ⓘ Note**
>
> The active region definitions also apply to MM calculations, but have to be provided via the `%qmmm` block.

Optimization in redundant internal coordinates

In ORCA's QM/MM geometry optimization only the positions of the active atoms are optimized. The forces on these active atoms are nevertheless influenced by the interactions with the non-active surrounding atoms. In order to get a smooth optimization convergence for quasi-Newton optimization algorithms in internal coordinates, it is necessary that the Hessian values between the active atoms and the directly surrounding non-active atoms are available. For that reason the active atoms are extended by a shell of surrounding non-active atoms which are also included in the geometry optimization, but whose positions are constrained, see Fig. 6.3. This shell of atoms can be automatically chosen by ORCA. There are three options available:

Table 6.2: Options to extend the active region.

Keyword	Options	Description
EXTENDAC	DIST.	(Default) The parameter `Dist_AtomsAroundOpt` controls which non-active atoms are included in the extension shell, i.e. non-active atoms that have a distance of less than the sum of their VDW radii plus `Dist_AtomsAroundOpt` are included.
	COV_	All (non-active) atoms that are covalently bonded to active atoms are included.
	NO	No non-active atoms are included

The user can also provide the atoms for the extension shell manually. This will be discussed in section *Frequency Calculation*.
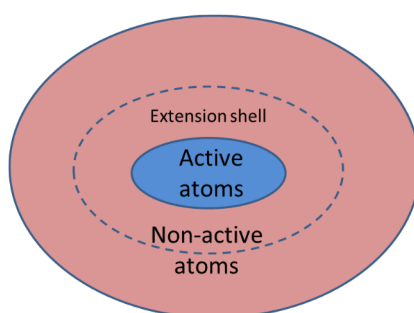

Fig. 6.3: Active and non-active atoms. Additionally shown is the extension shell, which consists of non-active atoms close in distance to the active atoms. The extension shell is used for optimization in internal coordinates and PHVA.

The set of active atoms is called the 'activeRegion', and the set of active atoms plus the surrounding non-active atoms is called 'activeRegionExt'. During geometry optimization the following trajectories are stored (which can be switched off):

Table 6.3: Files containing trajectories during optimization.

Filename	Description
`basename_trj.xyz`	Entire QM/MM system
`basename.QMonly_trj.xyz`	Only QM region
`basename.activeRegion_trj.xyz`	Only active atoms
`basename.activeRegionExt_trj.xyz`	Active atoms plus extension

The following files are stored containing the optimized structures - if requested:

Table 6.4: Files containing optimized structures.

Filename	Description
`basename.pdb`	Optimized QM/MM system in pdb file format
`basename.xyz`	Optimized QM/MM system
`basename.QMonly.xyz`	Only QM region
`basename.activeRegion.xyz`	Only active atoms
`basename.activeRegionExt.xyz`	Active atoms plus extension

Optimization with the Cartesian L-BFGS Minimizer

For very large active regions the quasi-Newton optimization in internal coordinates can become costly. It can be advantageous to use the `L-Opt` or `L-OptH` feature, see section *Geometry Optimizations using the L-BFGS optimizer*. In the context of multiscale simulations it can also be useful to treat parts of the system or full molecules of the system as rigid. E.g. you can optimize the hydrogen positions of the protein and water molecules, and at the same time relax non-standard molecules, for which no exact bonded parameters are available, as rigid bodies. For available options in that context see *Optimization Assuming a Rigid Body*. An example is shown here:

```
!QMMM L-OptH
%qmmm
 QMAtoms {22397:22423} end
 ORCAFFFilename "DNA_hexamer.ORCAFF.prms"
end
*pdbfile 0 1 DNA_ligand.pdb

%frag
 Definition
  1 {22307:22396} end # cofactor
  2 {22397:22423} end # ligand
 end
end

%geom
 RigidFrags {1 2} end # treat cofactor and ligand as individual rigid bodies
end
```

Frequency Calculation

If all atoms are active, a regular frequency calculation is carried out when requesting `!NumFreq`. If there are also non-active atoms in the QM/MM system, the *Partial Hessian Vibrational Analysis* (PHVA) is automatically selected. Here, the PHVA is carried out for the above defined `activeRegionExt`, where the extension shell atoms are automatically used as 'frozen' atoms. Note that the analytic Hessian is not available due to the missing analytic second derivatives for the MM calculation. Note that in a new calculation after an optimization it might happen that the new automatically generated extended active region is different compared to the previous region before optimization. This means that when using a previously computed Hessian (e.g. at the end of an optimization or a NEB-TS run) the Hessian does not fit to the new extended active region. ORCA tries to solve this problem by fetching the information on the extended region from the `.hess` file. If that does not work (e.g. if you distort the geometry after the Hessian calculation) you should manually provide the list of atoms of the extended active region. This is done via the following keyword:

```
%qmmm
 OptRegion_FixedAtoms {27 1288:1290 4400} end  # manually define the
                                               # activeRegionExt atoms.
end
```

Note that ORCA did print the necessary information in the output of the calculation in which the Hessian was computed:

```
Fixed atoms used in optimizer        ... 27 1288 1289 1290 4400
```

Nudged Elastic Band Calculations

NEB calculations (section *Nudged Elastic Band Method*) can be carried out in combination with the multiscale features in order to e.g. study enzyme catalysis. When automatically building the extension shell at the start of a Multiscale-NEB calculation, not only the coordinates of the main input structure ('reactant'), but also the atomic coordinates of the 'product' and, if available, of the 'transition state guess' are used to determine the union of the extension shell of the active region. For large systems it is advised to use the active region feature for the NEB calculation.

> **ⓘ Note**
>
> The atomic positions of the non-active atoms of reactant and product and, if available, transition state guess, should be identical.

Molecular Dynamics

If there are active and non-active atoms in the multiscale system, only the active atoms are allowed to propagate in the MD run. If all atoms are active, all atoms are propagated. For more information on the output and trajectory options, see section *Regions*.

6.1.5 Boundary Region

This section is important for the QM/MM, QM/QM2 and QM/QM2/MM methods. For the latter method two boundary regions are present (between QM and QM2 as well as between QM2 and MM region), and both can go through covalent bonds. In the following we will only discuss the concept for the boundary between QM and MM, but the same holds true for the other two boundaries.

Link Atoms

ORCA automatically generates link atoms based on the information of the QM region and on the topology of the system (based on the `ORCAFF.prms` file). ORCA places link atoms on the bond between QM and MM atoms.

> **⓿ Important**
>
> When defining the QM, QM2 and MM regions, make sure that you only cut through single bonds (not aromatic, double, triple bonds, etc.).

The distance between QM atom and link atom is determined via a **bond length scaling factor** (in relation to the QM-MM bond length) that is computed as the ratio of the equilibrium bond length between QM and hydrogen atom (d_{QM-H}) and the equilibrium bond length between QM and MM atom (d_{QM-MM}).

For the **equilibrium bond lengths to hydrogen**, ORCA uses tabulated standard values for the most common atoms involved in boundary regions (C, N, O), which can be modified via keywords as defined further below. ORCA stores these values on-the-fly in a simple file (`basename.H_DIST.prms`). If necessary, the user can modify these values

atom-specific or add others to the file and provide this file as input to ORCA (see also paragraph *Get standard hydrogen bond lengths: getHDist*). For QM/QM2 and QM/QM2/MM methods the equilibrium bond lengths to hydrogen are explicitly calculated.

```
%qmmm
# standard equilibrium bond lengths with hydrogen can be modified
 Dist_C_HLA 1.09          # d0_C-H
 Dist_O_HLA 0.98          # d0_O-H
 Dist_N_HLA 0.99          # d0_N-H
# file can be provided which provides the used d0_X-H values specific to all atoms
 H_Dist_FileName "QM_H_dist.txt"
end
```

Bonded Interactions at the QM-MM Boundary

The MM bonded interactions within the QM region are neglected in the additive coupling scheme. However, at the boundary, it is advisable to use some specific bonded interactions which include QM atoms. By default ORCA neglects only those bonded interactions at the boundary, where only one MM atom is involved, i.e. all bonds of the type QM1-MM1, bends of the type QM2-QM1-MM1, and torsions of the type QM3-QM2-QM1-MM1. Other QM-MM mixed bonded interactions (with more than two MM atoms involved) are included. The user is allowed to include the described interactions, which is controlled via the following keywords:

```
%qmmm
 DeleteLADoubleCounting true      # Neglect bends (QM2-QM1-MM1) and torsions
                                  # (QM3-QM2-QM1-MM1). Default true.
 DeleteLABondDoubleCounting true # Neglect bonds (QM1-MM1)
end
```

Charge Alteration

If QM and MM atoms are connected via a bond (defined in the topology file), the charge of the close-by MM atom (and its neighboring atoms) has to be modified in order to prevent overpolarization of the electron density of LA and QM region. This charge alteration is automatically taken care of by ORCA in `%qmmm` block. ORCA provides the most popular schemes that can be used to prevent overpolarization.

Table 6.5: Options to modify the charge at the QM-MM boundary in order to prevent overpolarization.

Keyword	Options	Description
ChargeAltera	CS	(Default) Charge Shift - Shift the charge of the MM atom away to the MM2 atoms so that the overall charge is conserved
	RCD	Redistributed Charge and Dipole - Shift the charge of the MM atom so that the overall charge and dipole is conserved
	Z0	Keep charges as they are. This MM atom will probably lead to overpolarization
	Z1	Delete the charge on the MM1 atom (no overall charge conservation)
	Z2	Delete the charges on the MM1 atom and on its first (MM2) neighbors (no overall charge conservation)
	Z3	Delete the charges on the MM1 atom and on its first (MM2) and second (MM3) neighbors (no overall charge conservation)
Scale_RCD	real	For RCD scheme defines where on the bond between MM1 and MM2 the shifted charge is positioned. Default: 0.5, i.e. midpoint.
Scale_CS	real	For CS scheme defines where on the bond between MM2 and MM1/MM3 the shifted charge is positioned. Default: 0.06 x bond.

6.1.6 Embedding Types

There are two embedding schemes available for treating the electrostatic interaction between high level and low level region. In the electrostatic embedding scheme the electrostatic interaction between QM and MM system is computed at the QM level. Thus, the charge distribution of the MM atoms (or other low level methods) can polarize the electron density of the QM region. The following embeding schemes are available:

Table 6.6: Options for embedding scheme to treat the electrostatic interaction between high and low level region.

Keyword	Options	Description
Embedding	Electrostatic	(Default) The electrostatic interaction between QM and MM system is computed at the QM level.
	Mechanical	The electrostatic interaction between QM and MM system is computed at the MM level.

> ⏸ **Important**
>
> In QMMM the LJ interaction between QM and MM system is always computed at the MM level.

In the scheme of electrostatic embedding, the evaluation of the electrostatic potential generated by the MM part can be accelerated by using the FMM algorithm (described in reference [797]). This will speed up the building of the Fock Matrix. The default recommended setup can be called using the FMM-QMMM keyword directly in the keywords line. However, more details about the algorithm parameters and all input options can be found in (see also paragraph *Fast Multipole Method*)

```
! FMM-QMMM
```

It is recommended to use that option whenever the MM part is composed of more than 10,000 atoms (or point charges for ECM methods).

6.1.7 Additional Keywords

Here we list additional keywords and options that are accessible from within the `%qmmm` block and that are relevant to QM/MM calculations. Some of these keywords can also be accessed via the `%mm` block, see section *Available Keywords for the MM module*.

```
! QMMM
%qmmm

# The extended active region, activeRegionExt, contains the atoms surrounding the
# active atoms.
 ExtendActiveRegion distance  # rule to choose the atoms belonging
                              # to activeRegionExt.
                              # no - do not use activeRegionExt atoms
                              # cov_bonds - add only atoms bonded covalently to
                              # active atoms
                              # distance (default) - use a distance criterion (VDW
                              # distance plus Dist_AtomsAroundOpt)
 Dist_AtomsAroundOpt 0.0      # in Angstrom (Default 0). Meaning only freeze atoms
                              # which touch the active atoms by their VDW spheres.
                              # only needed for ExtendActiveRegion distance
 OptRegion_FixedAtoms {27 1288:1290 4400} end  # manually define the
                                               # activeRegionExt atoms.
                                               # Default: empty list.

# Printing options. All are true by default.
```

```
PrintLevel 1                # Can be 0 to 4, 0=nothing, 1=normal, ...
PrintOptRegion true         # Additionally print xyz and trj for opt region
PrintOptRegionExt false     # Additionally print xyz and trj for extended
                            # opt region
PrintQMRegion true          # Additionally print xyz and trj for QM region
PrintFullTRJ true           # Print trajectory of full system. Default true.
PrintPDB true               # Additionally print pdb file for entire system, is
                            # updated every iteration for optimization

# Computing MM nonbonded interactions within non-active region.
Do_NB_For_Fixed_Fixed true # Compute MM-MM nonbonded interaction also for
                            # non-active atom pairs. Default true.
# Treats all active water molecules that are treated on MM level as rigid bodies
#  in optimization and MD simulation, see section "Rigid Water".
Rigid_MM_Water false        #  Default false.

end

*pdbfile 0 1 ubq.pdb  # structure input via pdb file, but also possible via
                      # xyz file
```

Table 6.7: Additional keywords in the `%qmmm` block, which are also accessible from the `%mm` block.

Keyword	Options	Description
ORCAFFFilen	"<filen	Filename of the ORCA Force Field prms file.
Do_NB_For_F	true/fa.	Compute MM-MM nonbonded interaction also within non-active region, i.e. for non-active atom pairs. Default `true`. For `GOAT` and `Crystal-QMMM` default is `false`.
RIGID_MM_WA	true/fa.	Optimization and MD of active MM waters. Default `false`.
ExtendActiv	distanc	Scheme to extend the active region to activeRegionExt. See also *here*. `distance`: Use a distance criterion (VDW distance plus `Dist_AtomsAroundOpt`).
	cov_bor	Add only atoms bonded covalently to active atoms.
	no	Do not add any atoms to activeRegionExt.
Dist_AtomsA	<real>	Distance to be added to VDW distance for `ExtendActiveRegion`. In Angstrom. Default `0`.
OptRegion_F	{ints}	List of atoms to define the extended active region. Default: empty list.
PrintOptReg	true/fa.	Additionally print xyz and trj for opt region. Default `true`.
PrintOptReg	true/fa.	Additionally print xyz and trj for extended opt region. Default `false`.
PrintPDB	true/fa.	Additionally print pdb file for entire system, is updated every iteration for optimization. Default `false`.

6.2 Multiscale Simulations of molecular systems

ORCA features three methods suitable for molecular systems such as proteins, DNA, large molecules, explicit solvation, depending on their size and depending on the availability of force field parameters.

- *Additive QM/MM* is a pure additive QMMM method.

- *QM1/QM2* is a subtractive (2-layered) ONIOM method.

- *QM1/QM2/MM* is a 3-layered method, combining a subtractive method between QM1 and QM2, and an additive QMMM method for QM2/MM.

6.2.1 Additive QMMM

In the additive QMMM scheme force field parameters are used to describe the interactions in the MM region, as well as bonded interactions and LJ interactions between QM and MM atoms. Thus, force field parameters are required for the MM part of the calculation. ORCA currently supports the popular **CHARMM**, **AMBER**, and **Open Force Field** force fields, as discussed in *Molecular Mechanics Force Field in ORCA*, and provides tools to *convert* their force field parameter files into the ORCA force field file format.

Basic Usage

The minimum input necessary for an additive QM/MM calculation is shown in the following example. The force field filename is provided via the keyword ORCAFFFilename:

```
!QMMM
%qmmm
 QMAtoms {0 1 2 27 28} end
 ORCAFFFilename "UBQ.ORCAFF.prms"
end
```

> **ⓘ Note**
>
> ORCAFF.prms files need to be provided for QM/MM, QM/QM2/MM and IONIC-CRYSTAL-QMMM calculations, but not for QM/QM2 and MOL-CRYSTAL-QMMM.

6.2.2 ONIOM Methods: General Overview

For the simulation of large systems with up to 10000 atoms, or for large QM regions in biomolecules, ORCA provides the QM/QM2 and QM/QM2/MM methods. The specifics of the two different methods are discussed further below. Here we are presenting the common concepts and keywords of both methods. For subtractive methods, we use a high level (QM) and a low level (QM2) of accuracy for different parts of the system. The advantages of this - in contrary to QM-MM methods - are as follows:

- QM2 methods are polarizable, the interaction with the high level region is more accurate.

- No MM parameters are needed for the atoms that are described at the QM2 level.

Available QM2 Methods

ORCA has several built in QM2 methods:

- Semiempirical methods (AM1, PM3)

- Tight-binding DFT (XTB0, XTB1, XTB (or XTB2))

- Composite methods (HF-3c, PBEh-3c, r2SCAN-3c)

- User-defined QM2 methods

The individual keywords for these methods are:

Table 6.8: Options for QM2 for QM/QM2 and QM/QM2/MM methods available via simple keywords.

QM/QM2	QM/QM2/MM
!QM/XTB	!QM/XTB/MM
!QM/XTB1	!QM/XTB1/MM
!QM/XTB0	!QM/XTB0/MM
!QM/GFN-FF	!QM/GFN-FF/MM
!QM/HF-3C	!QM/HF-3C/MM
!QM/PBEH-3C	!QM/PBEH-3C/MM
!QM/R2SCAN-3C	!QM/R2SCAN-3C/MM
!QM/PM3	!QM/PM3/MM
!QM/AM1	!QM/AM1/MM

Users can define their own low-level methods in the following way

```
!QM/QM2         or   !QM/QM2/MM
%qmmm
  QM2CUSTOMMETHOD "B3LYP"
  QM2CUSTOMBASIS "def2-SVP def2/J"
end
```

Alternatively, a custom QM2 method / basis set file can be provided:

```
!QM/QM2  or  !QM/QM2/MM
%qmmm
  QM2CustomFile "myQM2Method.txt" # File should be available
                                 # in working directory
end
```

The custom QM2 method file can contain any desired input, as e.g. the file myQM2Method.txt:

```
!cc-pVDZ HF TightSCF NOSOSCF KDIIS
%basis
  NewAuxJKGTO Mg "AutoAux" end
end
```

> **ⓘ Note**
>
> Only add methods (including convergence settings) and basis sets for the QM2Custom options. Everything else (parallelization, memory, solvation, etc.) is taken care of by ORCA itself.

Electrostatic Interaction between high and low level

By default we are using electrostatic embedding, i.e. the high level system sees the atomic point charges of the low level (QM2) system. These point charges are derived from the full system low level (QM2) calculation. The following methods for determining these charges are available:

Table 6.9: Options for determining the atomic charges of the QM2 atoms.

Keyword	Option
Charge_Method	Hirshfeld (default)
	MBIS
	CHELPG
	Mulliken
	Loewdin (default if QM2 is AM1 or PM3)

The QM2 point charges can be scaled with the following keyword.

```
%qmmm
  Scale_QM2Charges_MMAtom 1. # default is 1.
end
```

Boundary Region

The boundary between high and low level part of the system can contain covalent bonds. For the detection and realistic treatment of these covalent bonds, a topology of the large QM2 system is generated at calculation start using the following keyword.

Table 6.10: Options for creating an initial topology for the full (QM/QM2) or medium (QM/QM2/MM) system.

Keyword	Option
AutoFF_QM2_Method	Simple_Eq (default)
	XTB
	XTB1
	XTB0
	GFNFF
	HF3C
	PBEH3C
	R2SCAN3C
	PM3
	AM1
	QM2

> **ⓘ Note**
>
> If you use any of the XTB methods as AutoFF_QM2_Method make sure to have the otool_xtb binary in your ORCA PATH, see *Extended Tight-Binding: GFN0-xTB, GFN-xTB, GFN2-xTB*.

6.2.3 Subtractive QM/QM2 Method (ONIOM2)

The QM/QM2 method is a very convenient way of running multiscale calculations without the need to prepare any parameters. This method is a subtractive QM-QM method, in which we treat a part of interest on a higher level of accuracy, and the remainder of the system on lower level of accuracy. The implementation follows similar works as e.g. described in reference [798].

The method can be used in a similar way as a regular QM calculation. Let us have a look at the proton transfer in propionic acid, which can be modeled as follows:

```
!QM/XTB BP86 def2-TZVP def2/J
!Fast-NEB-TS NumFreq
!pal8
%qmmm
  QMAtoms {0:3} end
end
%neb
 product "propionicAcid_prod.xyz"
 preopt true
end
*xyz 0 1
H       -0.738352472      0.000000000     -5.836214279
O       -0.738352472     -0.587240971     -5.061536853
O       -0.738352472      1.434717404     -4.069730302
C       -0.738352472      0.227304724     -3.975502162
C       -0.738352472     -0.566448428     -2.687358498
H        0.133951528     -1.231202352     -2.710760176
H       -1.610656472     -1.231202352     -2.710760176
C       -0.738352472      0.318369069     -1.443687014
H       -0.738352472     -0.294739868     -0.538164669
H        0.142397528      0.965221387     -1.423275731
H       -1.619102472      0.965221387     -1.423275731
*
```

with the product structure file (propionicAcid_prod.xyz):

```
11
C3H6O2
H       -0.738352472      1.628728096     -5.020130139
O       -0.738352472     -0.587240971     -5.061536853
O       -0.738352472      1.434717404     -4.069730302
C       -0.738352472      0.227304724     -3.975502162
C       -0.738352472     -0.566448428     -2.687358498
H        0.133951528     -1.231202352     -2.710760176
H       -1.610656472     -1.231202352     -2.710760176
C       -0.738352472      0.318369069     -1.443687014
H       -0.738352472     -0.294739868     -0.538164669
H        0.142397528      0.965221387     -1.423275731
H       -1.619102472      0.965221387     -1.423275731
```

As can be seen from the input, the only difference to a regular calculation is the necessity to define the high level region via the `QMAtoms` or `QMCoreAtoms` keyword.

System charges and multiplicities

The two subsystems can have different (integer) charges and multiplicities. Defining the correct charges and multiplicities is important. The charge and multiplicity defined via the coordinate section defines the charge and multiplicity of the high level region (QMAtoms). The user still needs to define the charge and multiplicity of the total system (corresponding to the sum of the charge of the high level and low level parts, and corresponding to the overall multiplicity).

```
%qmmm
  QMAtoms {0:3} end # high level region
  Charge_Total  0   # charge of the full system. Default 0.
  Mult_Total    1   # multiplicity of the full system. Default 1.
end
*xyz 0 1            # charge and mult. of the high level region, i.e. atoms 0 to 3
```

Solvation

Implicit Solvation effects can be included in QM/QM2 calculations. On the one hand, for QM/XTB calculations, one can adopt the analytical linearized Poisson-Boltzmann (ALPB) solvation model, the domain decomposition COSMO (ddCOSMO), or the extended conductor-like polarizable continuum model (CPCM-X), and on the other hand, if no XTB is requested, ORCA uses the C-PCM. The user just needs to add the following tags in the ORCA input file, XTB for the QM2 region:

```
!QM/XTB ALPB(Water)

or

!QM/XTB DDCOSMO(Water)

or

!QM/XTB CPCMX(Water)
```

No XTB for the QM2 region:

```
!QM/HF-3c CPCM(Water)
```

If the ddCOSMO (XTB) or the C-PCM (non-XTB) are requested, there are two possible ONIOM/implicit-solvation methods [799].

- **C-PCM/B:** The effect of the solvent is, in the first place, included in the calculation for the large QM2 system. Once this calculation finishes, the solvation charges located on the surface of the cavity for the large system are used as point charges for the subsequent low-level and high-level calculations for the small system.

- **C-PCM/C:** The effect of the solvent is only included in the calculation for the large QM2 system.

The user can choose one scheme or the other via the tag "solv_scheme" in the "qmmm" block:

```
%qmmm
solv_scheme CPCM_B   # CPCM_B (default)
                     # CPCM_C
end
```

If the ALPB model or the CPCM-X are requested (within QM/XTB methods), the solvation effect is just included in the calculation for the large QM2 system (as one does for the C-PCM/C scheme).

6.2.4 QM/QM2/MM Method (ONIOM3)

The QM/QM2/MM method uses a combination of the subtractive scheme for the QM-QM2 part, and the additive scheme for the (QM-QM2) - (MM) interaction. It can be used if very large QM regions are required for biomolecules and explicitly solvated systems. The system is divided into a high level (QM), low level (QM2), and MM region (MM).

QM2 Atoms

QM2 atoms need to be defined for QM/QM2/MM calculations. They can be defined either directly

```
%qmmm
 QMAtoms {0:4} end           # list of QM atoms (counting starts from 0) or
 QM2Atoms {5:22} end         # list of QM2 atoms
end                          # an atom should not occur in both lists
*pdbfile 0 1 ubq.pdb
```

or via the occupancy column of a pdb file (see *QM and Active Region Definition*).

> **ⓘ Note**
>
> QM2 atoms can be defined via 2 in the occupancy column. In this case `Use_QM2_InfoFromPDB` must be set to true.

System charges and multiplicities

The high and low level subsystems can have different (integer) charges and multiplicities. Defining the correct charges and multiplicities is important. The charge and multiplicity defined via the coordinate section defines the charge and multiplicity of the high level region (`QMAtoms`). The user still needs to define the charge and multiplicity of the medium system (corresponding to the sum of the charge of the high level and low level regions, and corresponding to the overall multiplicity of the combined high and low level region). The charge of the MM region is determined based on the MM parameters provided by the force field.

```
%qmmm
  QMAtoms {0:3} end # high level region
  Charge_Medium 0   # charge of the medium system. Default 0.
  Mult_Medium   1   # multiplicity of the medium system.  Default 0.
end
*xyz 0 1             # charge and mult. of the high level region, i.e. atoms 0 to 3
```

Basic Usage

An example for a QM/QM2/MM calculation looks as follows:

```
!QM/HF-3c/MM Opt B3LYP def2-TZVP def2/J NumFreq
%qmmm
 ORCAFFFilename "peptideChain.ORCAFF.prms"
 QMAtoms {16:33 68:82} end
 QM2Atoms {0:12 83:104} end
 ActiveAtoms { 0:38 65:120} end
 Charge_Medium 0
end
*pdbfile -1 1 peptideChain.pdb
```

6.3 CRYSTAL-QMMM

For the simulation of extended systems ORCA provides the CRYSTAL-QMMM features:

- *MOL-CRYSTAL-QMMM* for molecular crystals.
- *IONIC-CRYSTAL-QMMM* for semiconductors and insulators.

In this section we present the concepts and keywords that are common to both methods. ORCA is a molecular code and cannot carry out periodic calculations, but ORCA has been used for modeling selected properties for ionic solids using embedded cluster models already in the past (see e.g reference [599]). With ORCA 5 we provide the CRYSTAL-QMMM method, which simplifies setting up and running these types of embedded cluster calculations. The user needs to provide the structure with a (large enough) supercell representative for the crystal, and provide a list of QM atoms. The QM region should be located in the central part of the supercell. The QM atoms are embedded in the remainder of the supercell, the surrounding MM atoms, which are represented by atom-centered point charges and influence the QM calculations. How these charges are obtained, is outlined in the next paragraph.

6.3.1 Charge Convergence between QM and MM region

The core idea of the CRYSTAL-QMMM method is to reach charge convergence between the QM and the MM atoms. For this purpose, the MM charges are updated self-consistently with the atomic (CHELPG) charges of the QM atoms. This idea follows the work of reference [599] for the `IONIC-CRYSTAL-QMMM` and of reference [800] for the `MOL-CRYSTAL-QMMM`.

Table 6.11: Keywords for charge convergence settings for `!IONIC-CRYSTAL-QMMM` or `MOL-CRYSTAL-QMMM`

Keyword	Options	Description
`Conv_Charges`	`true`/`false`	Use charge convergence scheme. Default `true`.
`Conv_Charges_M`	`<int>`	Maximum number of cycles for charge convergence scheme. Default `30` for MOL-CRYSTAL-QMMM. Default `50` for IONIC-CRYSTAL-QMMM.
`Conv_Charges_C`	`<real>`	Convergence threshold (default `0.01`) for maximum charge change for atom type between two subsequent charge convergence cycles.
`Scale_FormalCh`	`<real>`	MM atomic charges used in QM part of the CRYSTAL-QMMM calculation are scaled by this factor. Default `1.`.

During optimizations, the charge convergence scheme can be used (i) only at the very beginning, (ii) every N geometry steps, or (iii) for each geometry step, using the following keyword:

```
%geom
  ReConvCharge 1 # default is 1. Redo charge convergence scheme every N steps.
end
```

After charge convergence, the converged charges are stored in the file `basename.convCharges.ORCAFF.prms`. It can be used for a later calculation (with the same or different electronic structure settings) with the following keyword combination:

```
!IONIC-CRYSTAL-QMMM   or   !MOL-CRYSTAL-QMMM
%qmmm
  Conv_Charges false
  ORCAFFFilename "firstjob.convCharges.ORCAFF.prms"
end
```

6.3.2 MM-MM Interaction

Unlike with the other multiscale methods (QMMM, QM/QM2, QM/QM2/MM) the MM region is only treated as a perturbation to the QM region. The (bonded and nonbonded) MM-MM interaction is not computed - since it would not affect the QM-MM interaction - and thus the active region (optimizations, frequencies, …) in CRYSTAL-QMMM calculations should be restricted to atoms of the QM region.

6.3.3 Preparing CRYSTAL-QMMM Calculations

The input structures and input files for the CRYSTAL-QMMM calculations can be prepared with the *orca_crystalprep* module.

6.3.4 MOL-CRYSTAL-QMMM

Conceptually we use an additive QM/MM calculation, in which the QM region interacts with the MM region only via nonbonded interactions. Lennard-Jones interaction between QM and MM region is computed using van der Waals parameters from the UFF force field. The charge convergence treatment between QM and MM region starts with zero charges on the MM atoms, and is iterated until convergence of the QM atomic (CHELPG) and MM point charges is achieved.

The `MOL-CRYSTAL-QMMM` method expects as structural input a supercell that consists of repeating, identical molecular subunits, i.e. the atom ordering of the molecules should always be the same, and one molecular subunit should follow the next one. The minimum input necessary for a `MOL-CRYSTAL-QMMM` run looks as follows.

```
%qmmm
  NUnitCellAtoms 16       # provide the number of atoms per molecular subunit
  QMAtoms {160:175} end   # QM atoms, should be in central part of the supercell.
                          # Must be at least one complete molecular subunit.
end
```

> **ⓘ Note**
>
> For molecular crystals it is assumed that the entire supercell consists of repeating units which each have zero charge. QM regions should consist of one or multiple of such charge-neutral units.

Extending the QM Region

ORCA can extend the QM region (which we call QM core region, defined by the original `QMAtoms` input) by one or multiple layers of molecular subunits using the `HFLayers` keyword. If the `HFLayers` keyword is used, the molecular subunits of the defined number of layers around the QM core region is added to the QM region. The layers of molecular subunits around the QM core region are detected via distance criteria.

```
%qmmm
  HFLayers 0                          # default 0
  HFLayerGTO "LANL2DZ"                # default. Use this basis set for the
                                      # HFLayer atoms.
  HFLayerECP "HayWadt"                # default. Use these ECPs for the HFLayer atoms.
  Conv_Charge_UseQMCoreOnly true      # only use the QM core region for the charge
                                      # convergence scheme
end
```

The HFLayer can be seen as a buffer region between the molecular subunit of interest (original QM atoms) and the surrounding subunits. The different possible reasons for `HFLayers` are as follows:

- For DLPNO calculations with `HFLayers`, the DLPNO multilevel feature is automatically switched on. The molecules of the `HFlayer` form an own fragment which is treated on HF level only, i.e. these molecules are not included in the Post-HF treatment.

- The `HFLayers` can also be used for non-DLPNO calculations. In such cases, the HFLayer molecules are in the same way included in the QM calculation as the other QM molecules. But their definition is a convenient way to choose a different basis set (and ECPs) for those molecules.

- It can be assumed that the QM charges computed for the QM core region are more realistic than the charges of the HFLayer atoms near the MM atomic charges. Thus, using only the QM atomic charges of the QM core region represent more realistic charges for the charge convergence scheme.

- If necessary, the atoms of the HFLayer can be defined by the user. Make sure that complete molecular subunits are used here.

```
%qmmm
  HFLayerAtoms {0:15} end
end
```

> **ⓘ Note**
>
> The term `HFLayers` might be misleading. Strictly speaking, only for `MOL-CRYSTAL-QMMM` calculations with `DLPNO` methods (e.g. `DLPNO-CCSD(T)`, `DLPNO-MP2`, `DLPNO-B2PLYP`) the HF layer atoms are treated on HF level. For other methods (e.g. DFT) the HF layer atoms are treated with the same electronic structure method as the QM core region atoms.

Basic Usage

An example for a `MOL-CRYSTAL-QMMM` calculation looks as follows:

```
! MOL-CRYSTAL-QMMM
! PBE def2-SVP Opt NumFreq
%qmmm
 NUnitCellAtoms 13
 QMAtoms {221:233} end
 ActiveAtoms {221:233} end
end
*xyzfile 0 1 Ala_SC.xyz
```

6.3.5 IONIC-CRYSTAL-QMMM

Conceptually we use an embedded cluster calculation, in which the QM region interacts only with the MM atomic point charges of the surrounding. Unlike with `MOL-CRYSTAL-QMMM`, the Lennard-Jones interaction between QM and MM region is not computed. The charge convergence treatment between QM and MM region starts with the initial MM charges (the charges initially read from the ORCAFF.prms file) and is iterated until convergence of the QM atomic (CHELPG) and MM point charges is achieved.

Boundary Region

For ionic crystals a boundary region needs to be introduced between the QM region and the atomic point charges of the MM atoms in order to avoid spurious electron leakage from the clusters and to treat dangling bonds on the surface of the QM region. This boundary region consists of capped effective core potentials (cECPs). These cECPs are placed on the position of the MM atoms that are directly adjacent to the QM region. ORCA analyzes the connectivity of the atoms of the supercell and can thus detect the layers of atoms around the QM region, where the first layer consists of the atoms directly bonded to the QM atoms, the second layer consists of the atoms directly bonded to the atoms of the first layer and so on. The charges of these cECPs are determined in the same way as the MM atomic charges during the charge convergence scheme.

```
%qmmm
  ECPLayerECP "SDD"                 # cECPs used for the boundary region
  ECPLayers 3                       # Number of cECP layers around the QM region.
                                    # Default is 3.
  Scale_FormalCharge_ECPAtom 1.     # default is 1. Charges on cECPs are scaled by
                                    # this factor
end
```

Extending the QM Region

ORCA can extend the QM region (which we call QM core region, defined by the original QMAtoms input) by one or multiple layers of atoms using the `HFLayers` keyword. If the `HFLayers` keyword is used, the atoms of the defined number of layers around the QM core region is added to the QM region. The layers of atoms around the QM core region are detected in the same way as defined above for the ECPLayers.

```
%qmmm
  HFLayers 0                        # default 0
  HFLayerGTO "LANL2DZ"              # default. Use this basis set for the
                                    # HFLayer atoms
  HFLayerECP "HayWadt"             # default. Use these ECPs for the HFLayer atoms.
  Conv_Charge_UseQMCoreOnly true # only use the QM core region for the charge
                                    # convergence scheme
end
```

The HFLayer can be seen as a buffer region between the actual atoms of interest (original QM atoms) and the surrounding cECP and MM point charge environment.The different possible reasons for `HFLayers` are as follows:

- For DLPNO calculations with `HFLayers`, the DLPNO multilevel feature is automatically switched on. The atoms of the HFLayer form an own fragment which is treated at HF level only, i.e. these atoms are not included in the Post-HF treatment.

- It can be assumed that the QM charges computed for the QM core region are more realistic than the charges of the HFLayer atoms near the cECPs and MM atomic charges, in particular for highly charged systems. Thus, using only the QM atomic charges of the QM core region represent more realistic charges for the charge convergence scheme.

- If necessary, the atoms of the HFLayer can be defined by the user:

```
%qmmm
  HFLayerAtoms {29:35} end
end
```

- The charge of the HFLayer is automatically computed based on the formal charges of its atoms. If necessary, the HFLayer charge can be provided by the user.

```
%qmmm
  Charge_HFLayer 10 # by default it is computed automatically based on the formal
                    # charges from the ORCAFF.prms file
end
```

> **ⓘ Note**
>
> The term `HFLayers` might be misleading. Strictly speaking, only for `IONIC-CRYSTAL-QMMM` calculations with DLPNO methods (e.g. `DLPNO-CCSD(T)`, `DLPNO-MP2`, `DLPNO-B2PLYP`) the HF layer atoms are treated on HF level. For other methods (e.g. DFT) the HF layer atoms are treated with the same electronic structure method as the QM core region atoms.

Net Charge of the Entire Supercell

For ionic crystals, the QM region (as well as the entire supercell) can consist of an unequal number of atoms of each atom type. As a result of that, the QM region may have non-zero net charge. Consequently, when assigning the atomic point charges to the MM atoms during the charge convergence scheme, the sum of the charge of the QM region, of the ECP layer, and of the atomic charges of the MM atoms, can deviate from the ideal charge of the supercell. In order to enforce the ideal charge of the supercell, the total charge of the entire system is enforced by equally shifting the charges of all MM (including boundary) atoms.

```
%qmmm
  Charge_Total  0          # default 0. Total charge of the supercell
  EnforceTotalCharge true # enforce the total charge by shifting MM charges
end
```

The charge shifting procedure can be restricted to the MM atoms on the outer parts of the supercell by defining the number of `OuterPCLayers`. If `OuterPCLayers` is set to 1, only the atoms of the most outer layer of the supercell (recognized based on the connectivity of the atoms) are included in the charge shifting procedure. If `OuterPCLayers` is larger than 2, the atoms connected to the most outer layer are additionally included in the charge shifting procedure, etc.

```
%qmmm
  OuterPCLayers 0          # default 0, i.e. charge shifting for all MM atoms
end
```

> **ⓘ Note**
>
> The charge of the QM core region can be chosen to be assigned automatically. This overwrites the charge provided in the xyz or pdb section.
>
> ```
> %qmmm
>   AutoDetectQCCharge false # default is false
> end
> ```

Example Input

A minimal example for an `IONIC-CRYSTAL-QMMM` calculation looks as follows:

```
! IONIC-CRYSTAL-QMMM
! NMR
! PBE pcSseg-2 def2/JK def2-svp/C
%qmmm
  QMAtoms {0:6} end
  ORCAFFFilename "nacl6.ORCAFF.prms"
  CHARGE_TOTAL    0
end
*xyzfile -5 1 nacl6.xyz
```

Different QM and MM regions Stored in the PDB file

The stored PDB file contains the following entries in its occupancy column.

Table 6.12: Definition of entries stored in the pdb file for Ionic-Crystal-QMMM calculations.

Value	Description
0	MM atoms mimicked by surrounding point charges
1	QM core region atoms
2	HFLayer atoms
3	cECPs
4	OuterPCLayer atoms (subset of MM atoms) if defined, are the only atoms that are used in the charge shift procedure for enforcing the total charge

6.4 QM/MM via Interfaces to ORCA

ORCA is easy to interface as a QM engine in pretty much any QM/MM environment, as it will accept a set of point charges from an external file (see section *Inclusion of Point Charges*) and it will return, in a transparent format, all the required information for computing energies and gradients to the driving program. In our research group we have experience with four different QM/MM environments: ChemShell, Gromacs, pDynamo and NAMD. Each of the interfaces is described in the following section. It is beyond the scope of the manual to be exhaustive, and only minimal working examples are going to be presented. These will cover mainly the technical aspects with respect to the QM part of the QM/MM calculation. For the initial preparation of the system the user is referred to the documentation of the driving program.

6.4.1 ORCA and Gromacs

In cooperation with the developers of Gromacs software package we developed an interface to ORCA. The interface is available starting with Gromacs version 4.0.4 up to version 4.5.5.

As mentioned above, the initial setup has to be done with the Gromacs. In the QM/MM calculation Gromacs will call ORCA to get the energy and the gradients of the QM atoms. The interface can be used to perform all job types allowed by the Gromacs software package, e.g. optimizations, molecular dynamics, free energy. In addition, for geometry optimizations we have implemented a microiterative scheme that can be requested by setting the keyword `bOpt = yes` in the Gromacs `.mdp` file. This will cause ORCA to perform a QM geometry optimization every time it is called by Gromacs. During this optimization ORCA will also compute the Lennard-Jones interaction between the QM and MM atoms, and freeze the boundary atoms. This microiterative scheme can also be used to perform a transition state optimization. If you are looking for a transition state and have a good initial guess for the structure, you can carry out an optimization of the MM system and at the same time perform a transition state optimization of the QM system with ORCA via the microiterative scheme. Since it is expensive to calculate the Hessian for each microiterative step, the user can tell ORCA to use the (updated) Hessian matrix of the previous step via `read_temp_Hess` in the ORCA input.

In order to allow full flexibility to the user, the information for the QM run are provided in a separate plain text file, called `GromacsBasename.ORCAINFO`. When Gromacs writes the input for the ORCA calculation, it will merge this file with the information on the coordinates, point charges, Lennard-Jones coefficients and type of calculation (`EnGrad`, `Opt`, `TSOpt`).

Below is a typical example of an input file created by Gromacs, where for each Gromacs optimization step, a full optimization of the QM-part will be performed by ORCA, instead of just doing the energy and gradient calculation.

```
# Optimization step in the Lennard-Jones and point charges field of the MM atoms
! QMMMOpt

# file containing the Lennard Jones coefficients for the Lennard-Jones interaction
%LJCoefficients "temp.LJ"
```

(continues on next page)

```
# file containing the point charges for the electrostatic interaction
%pointcharges "temp.pc"

%geom
    # calculate the exact Hessian before the first optimization step
    Calc_Hess true
    # in case of a TS optimization the updated Hessian of the previous
    # TS optimization run is read instead of calculating a new one
    read_temp_Hess true
 end
```

> **ⓘ Note**
>
> - If you are using `bOpt` or `bTS` you have to take care of additional terms over the boundary. Either you can zero out the dihedral terms of the Q2-Q1-M1-M2 configurations, or you can fix the respective Q2 atoms.
>
> - If you want to use the ORCA constraints, you have to also put them in the Gromacs part of the calculation.
>
> - If there are no bonds between the QM and MM systems, the `bOpt` optimization of the QM system might have convergence problems. This is the case if the step size is too large and thus QM atoms come too close to MM atoms. The convergence problems can be circumvented by adding extra coordinates and Hessian diagonal values for Cartesian coordinates of selected QM atoms to the set of internal coordinates. This should be done for only few atoms that are in the boundary region.

```
%geom
    # add the Cartesian position of atoms 2 and 4 to the
    # set of internal coordinates with a diagonal Hessian value of 0.1
    Hess_Internal
    {C 2 D 0.1}
    {C 4 D 0.1}
    end
end
```

6.4.2 ORCA and pDynamo

The interface with the pDynamo library is briefly documented here. It uses the same plain text files to exchange information between the pDynamo library and ORCA. In order to have simiar functionality as presented above, we have extended the interface to generate more complex ORCA input files by accepting verbatim blocks of text. We have also implemented in pDynamo the capability of writing files containing Lennard-Jones parameters.

A simple example which calculates the QM/MM energy for a small designed peptide in which the side chain of one amino acid is treated QM is ilustrated below. For this example, a complete geometry optimization is going to be performed during the ORCA call.

```python
import os
import pCore
import pBabel
import pMolecule
import pMoleculeScripts

# Read the initial coordinates from the .pdb file.
system = pMoleculeScripts.PDBFile_ToSystem(
        "1UAO.pdb", modelNumber=1, useComponentLibrary=True)

# Instantiate the required models.
mmmodel = pMolecule.MMModelOPLS("protein")
nbmodel = pMolecule.NBModelORCA()
qcmodel = pMolecule.QCModelORCA(
```