# CellMuncher - A tool for super-cell manipulations

Copyright © L. Houben, ER-C, Research Centre Jülich GmbH
[www.er-c.org](http://www.er-c.org)

CellMuncher is a perl script that converts super-cell files and that manipulates the atom list in many ways. It uses an internal list representation. Super-cell data can be supplied in two different formats used by popular electron microscopy image simulation packages, namely MacTempas and EMS. The same formats are supported for export, besides CIF and VRML format and formats for the structure-viewer programs VESTA, CrystalMaker and JSV.

# Installation

### Unix flavoured machines

On a Unix flavour machine, determine the installation path of the perl binary by the shell command

$ which perl

The default installation path is /usr/local/bin. If your perl installation path deviates from the default one, open CellMuncher in an editor and edit the binary path for perl on the first line of the script. Finally, copy CellMuncher into a directory where binaries are looked for, e.g.

$ sudo cp CellMuncher /usr/local/bin

You probably need super user rights to do this.

### Windows

There are two options to use CellMuncher in Windows:

(1)  Install the pre-compiled binary CellMuncher.exe. CellMuncher.exe can be executed in cmd.exe

(2)  Install a Windows implementation of Perl, like Strawberry Perl. CellMuncher can then be executed in a perl shell.

# Usage

CellMuncher is a command line driven program. To process a super-cell file called mycell.cel the command has the general form

$ CellMuncher [OPTIONS]

Important OPTIONS are

--input=<name of the input file>

--output=<name of the outputfile>

A complete listing of the available options is given below. A short help page is presented when the --help option is chosen:

$ CellMuncher --help

A simple example:

$ CellMuncher --input=mycell.cel --output=mycell_copy.cel

copies the Stadelmann EMS file mycell.cel into mycell_copy.cel. (Note that it is not an exact copy since the formatting of the file is checked upon copying!)


# Options


Options can be given in a long form (--option[=value]) or (for almost all options) in a short form of a single character (-o [=value]). Only a few option are exclusively available in the long form. Note that the equal sign = is not mandatory, a space will do also.

Full list of options

Note: <> symbolize that the argument within the brackets is a mandatory argument.

--attach-cell=<value>

--align-z-axis-non-periodic|S=<value>

--align-z-axis-periodic|D=<value>

--cif

--centre-atoms=<value>

--centre-at-zero|N

--centre-rotate-axis|O=<value>

--correct-dimension|u=<value>

--create-block|b=<value>

--cut-above|c=<value>

--cut-above-plane|a=<value>

--cut-at-distance=<value>

--cut-below|C=<value>

--cut-below-plane|A=<value>

—cut-below-distance=<value>

--cut-cell-dimension=<value>

--decorate=<value>

--debug

—delete-duplicate | —unique | -d

--delete-element|Y=<value>

--dice-pair-site-occupancy=<value>

--dice-site-occupancy

--expand-structure=<value>

--extend-cell|e=<value>

--extend-cell-centre|E=<value>

--extend-to-size=<value>

--frozen-lattice|F=<value>

--frozen-lattice-fix-margin=<value>

--help|h

—input-file | —input | -in | —f=<value>

--invert-axis|I=<value>

--merge-file|H=<value>

--mirror-atom|m=<value>

--mirror-axis|M=<value>

—output-file |—output | —out | -o=<value>

--override

--periodic|p=<value>

--point-defect-displacements=<value>

--preview=<value>

--prompt

--read-format

—input-format | r=<value>

--remove-close-atoms|v=<value>

--repeat|j=<value>

--repeat-to-size=<value>

--replace=<value>

--reverse

--rotate-axis|R=<value>

—rotate-non-periodic|Q=<value>

—search-command=<value>

—search-help=<value>

--set-dw-factor|B=<value>

--set-occupancy=<value>

--slice|L=<value>

--slice-by-list|l=<value>

--shuffle

--screw-dislocation=<value>

--sort|s=<value>

--swap-axes|x=<value>

--translate-atom|t=<value>

--translate-axis|T=<value>

—unformatted

—unique | —delete-duplicate

--wedge-cut-above|k=<value>

--wedge-cut-below|K=<value>

--write-format|--output-format|w=<value>

## Detailled Description

Values are given as a comma separated list. The following symbols are used:

s: string or character, e.g. x

i: integer value, e.g. 2

f: floating point value, e.g. 3.14159.

Note:  <> symbolize that the argument within the brackets is a mandatory argument. [] symbolizes optional arguments, multiple means that one command line may contain more than one of these options

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --help | display a help page | - | ✘ | --help |
| —search-command | search a string within the command options | s<br><br>s: search string | ✘ | --search-command=sort<br><br>search command options for the occurrence of 'sort'.<br><br> displays all option keys if multiple commands match,<br><br>displays the help description if the result is a unique command |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| —search-help | search a string in the command descriptions | s<br><br>s: search string | ✗ | --search-help=sort<br><br>search c for the occurrence of 'sort'.<br><br> displays all option keys if multiple commands match,<br><br>displays the help description if the result is a unique command |
| --input-file=<value><br>—input=<value><br>--in=<value><br>-f <value> | file name of the input super-cell | s<br><br>filename | ✗ | --input-file=mycell.cel<br><br>input data will be read from mycell.cel |
| --output-file=<value><br>--output=<value><br>--out=<value><br>-o <value> | file name of the output super-cell | s<br><br>filename<br><br>Note: the suffix will be appended automatically, a base name is sufficient | ✗ | —write-format=XMS,CIF —output-file=my2ndcell<br><br>output data will be written to from my2ndcell.cel and my2ndcell.cif<br><br>if not given, then CellMuncher will try to construct the output filename from the input filename, i.e. a file conversion from EMS to MacTempas format will work with the two options<br><br>—write-format=MT —input=mycell.cel<br><br>to produce mycell.at |
| --read-format=<value><br>--input-format=<value><br>-r=<value> | file format of the input super-cell | s<br><br>XMS=EMS format<br><br>MT=MacTempas format<br><br>XYZ=Ascii format | ✗ | --read-format=XMS<br><br>input data is an EMS data file<br><br>XYZ ascii data file format:<br>line #1: comment line<br>line #2: a b c alpha beta gamma<br>line #3: number of atoms N<br>line #4: Atom Symbol X Y Z<br>line #N+3: - " -<br>X,Y, and Z are in absolute coordinates (Angstrom) |

| option | description | value format/ choices | mult iple | example/ explanation/ comment |
|---|---|---|---|---|
| --write-format=\<value\><br>--output-format=\<value\><br>-w=\<value\> | file format of the output super-cell | s1[,s2[,s3[,…]]]<br><br>where s1,s2,s3 are<br><br>XMS=EMS format (enhanced precsion 10F6, if --unformatted is not set)<br><br>EMS=EMS format (standard precision 8F4, if --unformatted is not set)<br><br>MT=MacTempas format<br><br>CIF=CIF format<br><br>VESTA=for Vesta structure viewer, identical with CIF format<br><br>CM=Crystal Maker text format, also for Vesta<br><br>ALI=Crystal Maker text format<br><br>JSV=JSV structure viewer format<br><br>WRL=VRML format | ✘ | —write-format=XMS<br><br>output data is an EMS data file with enhanced accuracy (10F6 instead of 8F4 for the atom positions)<br><br>—write-format=CIF,XMS,MT<br><br>output data is in CIF, EMS and MacTempas format |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --attach-cell=<value> | attach a super cell file to the input file at the input cell upper face | s1,s2,s3<br><br>s1:filename<br><br>s2:EMS,XMS,MT,XYZ,JSV<br><br>s3: dimension along which the cell will be attached, i.e. x, y, or z | ✔ | --attach-cell=attachme.cel,XMS,x<br><br>attaches the cell attachme.cel in EMS format to the super cell in the input file, the attached atoms will be placed at the upper face along the x dimension of the input cell<br><br>Note: other cell dimensions have to match |
| --merge-file=<value><br><br>-H <value> | merges a super cell file to the input file | s1,s2[,f1,f2,f3]<br><br>s1:filename<br><br>s2:EMS,XMS,MT,XYZ,JSV<br><br>f1,f2,f3: position in relative coordinates, optional | ✔ | --merge-file=mymerge.cel,XMS<br><br>merges the cell mymerge,cel in EMS format to the super cell in the input file,the merged atoms will be centred in the middle of the cell<br><br>--merge-file=mymerge.cel,XMS,0.2,0.3,0.6<br><br>same, but the atoms in mymerge.cel will be centred around the relative position 0.2,0.3,0.6<br><br>Note: so far the cell of the merge file needs to be smaller than the cell of the input file |
| --delete-element=<value><br><br>-Y <value> | delete an element from the list | s<br><br>element symbol | ✔ | --delete-element=Mo<br><br>delete all molybdenum atoms from the list |
| --set-dw-factor=<value><br><br>-B <value> | set the Debye-Waller factor for one element | s,f<br><br>element symbol, DW factor in nm | ✔ | --set-dw-factor=Sr,0.0046<br><br>set the Debye-Waller factor of strontium atoms to 0.0046 nm -responds to $8*pi^2*u^2$ where $u^2$ is the isotropic mean quadratic displacement |
| --set-occupancy=<value> | set the site occupancy for one element | s,f<br><br>element symbol, site occupancy | ✔ | --set-occupancy=Ga,0.6<br><br>set the site occupancy of gallium atoms to 0.6 |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --dice-pair-site-occupancy=<value> | create a specific site occupancy configuration for a site that may be occupied by two elements A and B | s1,s2,f<br><br>s1: element symbol for A<br>s2: element symbol for B<br>f: maximum distance between A and B sites in Angstrom | ✔ | --dice-pair-site-occupancy=Pd,Al,0.4<br><br>for each of the Pd and Al sites that are closer than 0.4 Å: set the occupancy of the Pd atom to 1 with a probability, that corresponds to the occupancy on that site, the occupancy of Al is set to 0 if the site is occupied by Pd and 1 otherwise<br><br>note that vacant sites are deleted from the atom list! |
| --dice-site-occupancy | create a specific site occupancy configuration | - | ✘ | --dice-site-occupancy<br><br>set the occupancy for each atom site to 1 or 0;  the probability for site occupation corresponds to the occupancy parameter<br><br>note that occupancies are either 1 or 0 afterwards, vacant sites are deleted from the atom list! |
| --repeat=<value><br><br>-j <value> | repeat the super cell along one direction | s,i<br><br>direction: x,y or z, integer factor | ✔ | --repeat=x,2<br><br>repeat the super cell 2 times along x |
| --repeat-to-size=<value> | scale the super cell along one direction to a certain size in Ångstrom and fill with atoms | s,f<br><br>direction: x,y or z, size in Ångstrom | ✔ | --repeat-to-size=x,200<br><br>scale the super cell in x-direction to 200 Ångstrom |
| --wedge-cut-above=<value><br><br>-k <value> | cut a wedge from the top of the super cell | s1,f1,f2<br><br>s1: x or y<br>f1: angle in °<br>f2: offset in relative coordinates<br><br>an atom is removed, when $z > s1*\tan(f1) + f2$ | ✔ | --wedge-cut-above=y,30,0<br><br>create a 30°-wedge in the yz-plane, top part of the cell is vacuum |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --wedge-cut-below=<value>  -K <value> | cut a wedge from the bottom of the super cell | s1,f1,f2  s1: x or y f1: angle in ° f2: offset in relative coordinates  an atom is removed, when $z < s1*tan(f1) + f2$ | ✔ | --wedge-cut-below=x,30,0  create a 30°-wedge in the xz-plane, bottom part of the cell is vacuum |
| --cif | Shortcut for converting an input EMS or XMS file to CIF | - | ✘ | --cif —input=mycell.cel  creates CIF file with the name mycell.cif |
| --cut-above=<value>  -c <value> | cut along one direction above a certain value | s1,f [,s2]  s1: x or y f: cut coordinate [or atom index] s2: index, indicates that f is the number of atom from where to cut (optional) | ✔ | --cut-above=x,0.7  cut all atoms with x > 0 .7  --cut-above=x,123,index  cut all atoms with index > 123 in the list sorted for x |
| --cut-below=<value>  -C <value> | cut along one direction below a certain value | s1,f [,s2]  s1: x or y f: cut coordinate [or atom index] s2: index, indicates that f is the number of atom from where to cut (optional) | ✔ | --cut-below=x,0.3  cut all atoms with x < 0 .3  --cut-below=x,12,index  cut all atoms with index < 12 in the list sorted for x |
| --cut-above-plane=<value>  -a <value> | cut atoms above a certain plane in a distance from the origin | f1,f2,f3,f4  f1,f2,f3: plane indices f4: distance from origin in Angstrom | ✔ | --cut-above-plane=1,1,1,25  cut all atoms above the 111-plane in a distance of 25 Angstrom |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --cut-below-plane=&lt;value&gt;<br><br>-A &lt;value&gt; | cut atoms below a certain plane in a distance from the origin | f1,f2,f3,f4<br><br>f1,f2,f3: plane indices<br>f4: distance from origin in Angstrom | ✔ | --cut-below-plane=-1,-1,-1,25<br><br>cut all atoms below the 111-plane in a distance of 25 Angstrom |
| --cut-at-distance=&lt;value&gt; | cut atoms beyond a certain distance from the origin | s,f<br><br>s: axis, x,y or z or xy,yz or xz or xyz<br>f: distance from origin in Angstrom | ✔ | --cut-at-distance=xy,25<br><br>cut a cylinder with the z-axis as the cylinder axis and a radius of 25 Angstrom<br><br>--cut-at-distance=xyz,25<br><br>cut a sphere with a radius of 25 Angstrom<br><br>Note: Only for orthogonal lattices |
| --cut-below-distance=&lt;value&gt; | cut atoms below a certain distance from the origin | s,f<br><br>s: axis, x,y or z or xy,yz or xz or xyz<br>f: distance from origin in Angstrom | ✔ | --cut-below-distance=xyz,15<br><br>cut atoms within a a sphere with a radius of 15 Angstrom<br><br><br>Note: Only for orthogonal lattices |
| --cut-cell-dimension=&lt;value&gt; | cut cell along one direction below and above a certain relative coordinate | s1,f1,f2<br><br>s1: x or y or z<br>f1, f2: relative cut coordinate, lower bound and upper bound | ✔ | --cut-cell-dimension=x,0.2,0.8<br><br>cut all atoms with x < 0 .2 and x > 0.8 and correct unit cell dimension |
| --decorate=&lt;value&gt; | decorates the positions in a super cell with a point group from a file | s1,s2[,s3]<br><br>s1: filename<br>s2: file format: EMS,XMS,MT, XYZ,JSV<br>s3: orientation, optional values: "rndrot" | ✔ | --decorate=molecule.cel,XMS,rndrot<br><br>decorates all the positions in the input file with the molecule data in molecule.cel. The molecule will be rotated in a random fashion. Note that the centre of the molecule in molecule.cel should be at 0,0,0. |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --extend-cell=<value> <br><br> -e <value> | extend unit cell without translating elements | s,f <br><br> s: axis, x,y or z <br> f: scaling factor | ✔ | --extend-cell=x,1.15 <br><br> extends the unit cell along x by a factor of 1.15 |
| --extend-cell-centre=<value> <br><br> -E <value> | extend unit cell, translate atoms so that the old cell is centred in the new cell | s,f <br><br> s: axis, x,y or z <br> f: scaling factor | ✔ | --extend-cell-centre=y,1.2 <br><br> extends the unit cell along y by a factor of 1.2, then translates along y by 0.1 |
| --extend-to-size=<value> | extend unit cell to a certain size in Ångstrom, translate atoms so that the old cell is centred in the new cell | s,f <br><br> s: axis, x,y or z <br> f: size in Å | ✔ | --extend-to-size=z,100 <br><br> extends the unit cell along z to a size of 100 Ångstrom and translates the atoms into the centre around z=50 Ångstrom |
| --expand-structure=<value> | expand a non-periodic structure e.g. a nanoparticle by a factor | f <br><br> f: factor | ✘ | --expand-structure=1.05 <br><br> expands a structure by 5 % |
| --align-z-axis-periodic=<value> <br><br> -D <value> <br><br> BROKEN! | align the z axis to a direction uvw for a periodic object, tries to find new period | f1,f2,f3 <br><br> f1,f2,f3: uvw indices | ✘ | --align-z-axis-periodic=1,2,1 <br><br> Aligns the z-axis to the 121 viewing direction <br><br> Works only for orthogonal lattices! Use the --periodic option to fold the cell periodic. Use the --remove-close-atoms option to remove close atoms |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --align-z-axis-non-periodic=<value>  -S <value> | align the z axis to a direction uvw for a periodic object | f1,f2,f3  f1,f2,f3: uvw indices | ✗ | --align-z-axis-non-periodic=1,2,1  Aligns the z-axis to the 121 viewing direction.  Works only for orthogonal lattices! |
| --rotate-non-periodic=<value>  -Q <value> | rotate a non-periodic object around an axis uvw through the centre of a super-cell | f1,f2,f3,f4  f1,f2,f3: uvw indices f4: angle in ° | ✔ | --rotate-non-periodic=1,2,2,14  Rotates by 14° around the 122 direction, the rotation axis goes through the centre of the super-cell.  Works only for orthogonal lattices! |
| --rotate-axis=<value>  -R <value> | rotate around an axis uvw through 0,0,0 | f1,f2,f3,f4  f1,f2,f3: uvw indices f4: angle in ° | ✔ | --rotate-axis=2,1,0,10  Rotates by 10° around the 210 direction, the rotation axis goes through the 000 corner of the super-cell.  Works only for orthogonal lattices! |
| --centre-rotate-axis=<value>  -O <value> | rotate around an axis uvw through the centre of the super-cell | f1,f2,f3,f4  f1,f2,f3: uvw indices f4: angle in ° | ✔ | --centre-rotate-axis=2,1,0,10  Rotates by 10° around the 210 direction, the rotation axis goes through centre of the super-cell.  Works only for orthogonal lattices! |
| --orthogonalize-plane=<value>  -g <value> | orthogonalize a plane with non-orthogonal base vectors | s,i  s: xy, xz or yz  i: maximum increase in cell volume | ✗ | --orthogonalize-plane=xz,10  Orthogonalize xz-plane, maximum increase in cell volume is 10. The algorithm tries to find a new periodic supercell.  Works only for xz so far (beta is not 90°), xy and yz are not implemented yet!  Note: it depends on the angle and base vector lengths whether a solution can be found. The program will stop if there is no exact solution within the given bound. In that case it is better to create a non-periodic orthogonal cell using the --create-block option. |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --create-block=<value><br><br>-b <value> | repeat a unit cell to an orthogonal block of a certain size and with a certain viewing direction | f1,f2,f3,f4,f5,f6,f7,f8,f9<br><br>f1-f3: x,y,z components of the viewing direction relative to the unit cell axes; this direction will be the new z direction<br><br>f4-f6: x,y,z components relative to the unit cell axes of a direction that has an orthogonal component to the viewing direction; this direction defines the new b axis of the resulting super-cell<br><br>f7-f9: the dimensions x,y, and z of the new cell in nm | ✘ | --create-block=1,1,0,-1,1,0,5,5,3<br><br>create a block with a size of 5x5x3 nm3 and a z-direction that corresponds to 110, the new y-direction is parallel to -1,1,0 |
| --invert-axis=<value><br><br>-I <value> | invert atom positions on an axis | s<br><br>s: x,y or z | ✔ | --invert-axis=z<br><br>Invert atom positions on the z-axis. |
| --mirror-axis=<value><br><br>-M <value> | mirror atom positions on an axis | s,f<br><br>s: x,y or z<br>f: mirror position in relative coordinates | ✔ | --mirror-axis=x,0.3<br><br>Mirror atom positions on the x-axis, mirror point is the relative coordinate 0.3 |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --mirror-atom=\<value\><br><br>-m \<value\> | mirror single atom on an axis | s,i,f<br><br>s: x,y or z<br>i: atom index<br>f: mirror position in relative coordinates | ✔ | --mirror-atom=y,121,0.6<br><br>Mirror atom 121 on the y-axis, mirror point is the relative coordinate 0.6 |
| --translate-axis=\<value\><br><br>-T \<value\> | tranlate atom positions on an axis | s,f<br><br>s: x,y or z<br>f: translation in relative coordinates | ✔ | --translate-axis=x,0.3<br><br>Translate all the atoms on the x-axis by value 0.3 in relative coordinates |
| --translate-atom=\<value\><br><br>-t \<value\> | translate single atom on an axis | s,i,f<br><br>s: x,y or z<br>i: atom index<br>f: translation in relative coordi-nates | ✔ | --translate-atom=z,12,0.1<br><br>Translate atom 12 on the z-axis by 0.1 relative to the z-dimension of the unit cell. |
| --frozen-lattice=\<value\><br><br>-F \<value\> | generate a frozen lattice configuration | s<br><br>s: x,y or z or a combination like x,y or x,z or x,y,z | ✔ | --frozen-lattice=x,y<br><br>create a frozen lattice configuration<br><br>takes for each atom the DW factor and creates a random displacement in x,y<br><br>displacements along one crystallographic direction follow a normal distribution with standard deviation Sqrt(1/(3*8Pi^2)*DW) and are usually about 4-6 pm on average |
| --frozen-lattice-fix-mar-gin=\<value\> | generate a frozen lattice configuration, keep atoms at the boundary fixed | s,f1,f2,f3<br><br>s: x,y or z<br><br>f1,f2,f3: margin width in x,y and z respectively in relative supercell size | ✔ | --frozen-lattice-fix-margin=x,0.1,0.2,0.0<br><br>create a frozen lattice configuration for atoms in 0.1 <= x <= 0.9 and 0.2 <= y <= 0.8 and 0 <= z <=1<br><br>takes for each atom the DW factor and creates a random displacement in x,y<br><br>displacements along one crystallographic direction follow a normal distribution with standard deviation Sqrt(1/(3*8Pi^2)*DW) and are usually about 4-6 pm on average |

| option | description | value format/ choices | mult iple | example/ explanation/ comment |
|---|---|---|---|---|
| --sort=<value><br><br>-s <value> | sort atom list | s<br><br>s: x, y, z, e, all | ✔ | --sort=x --sort=e<br><br>first sort according to the x-coordinate then according to the element |
| --correct-dimension=<value><br><br>-u <value> | correct unit cell dimensions (deletes vacuum space) | s<br><br>s: x,y or z or a combination like x,y or x,z or x,y,z | ✔ | --correct-dimension=x,y<br><br>correct dimensions in x and y, after correction, the minimum (maximum) x ynd y relative atom coordinate is 0 (1) |
| --centre-atoms=<value> | centre atoms within the volume of the unit cell | s<br><br>s: x,y or z or a combination like x,y or x,z or x,y,z | ✔ | --centre-atoms=x,y<br><br>centre atoms in x and y<br><br>hint: use the --periodic option before! |
| --centre-at-zero | centre atoms around the origin (0,0,0) of the coordinate system | | ✘ | --centre-at-zero<br><br>centre atoms around x=0, y=0, z=0<br><br>hint: use the --periodic option before! |
| --screw-dislocation=<value> | create a displacement field of a srew islocation, including Eshelby twist | f1,f2,f3<br><br>f1,f2: x position in relative coordinates f3:magnitude of Burgers vector in Ångstrom | ✘ | --screw-dislocation=0.5,0.5,1.5<br><br>creates a screw dislocation in the centre of the super cell<br>Burgers vector has a magnitude of +1.5 Angstrom<br>atoms are shifted according to the Eshelby equation<br><br>hint: expand unit cell before in order to avoid wrap-arounds |
| --swap-axes=<value><br><br>-x <value> | swap two axes | s<br><br>s: xy or xz or yz | ✔ | --swap-axes=xy<br><br>swap x-axis with y-axis |

| option | description | value format/ choices | multiple | example/ explanation/ comment |
|---|---|---|---|---|
| --periodic=<value><br><br>-p <value> | fold periodic | s<br><br>s: x or y or z | ✔ | --periodic=x<br><br>fold x axis periodic, that means all relative x-coordinates will be translated to the interval [0,1[, presuming that the structure is periodic |
| --delete-duplicate<br><br>-d | delete duplicate atoms | - | ✘ | --delete-duplicate<br><br>delete duplicate atoms in the list |
| --remove-close-atoms=<value><br><br>-v <value> | removes atoms that are closer than a certain distance to an atom that appeared earlier in the list | f<br><br>f: distance in Angstrom | ✘ | --remove-close-atoms=0.8<br><br>removes atoms that are closer than 0.8 Angstrom: the atom wit the lower index in the list will remain, use together with the --sort, --shuffle, --reverse options on elements to re-arrange the list and determine which atom or element will be first and remain |
| --debug | switch on debugging code | - | ✘ | --debug |
| --override | don't ask if output-file exists, just override it | - | ✘ | --override |

| option | description | value format/ choices | mult iple | example/ explanation/ comment |
|---|---|---|---|---|
| --point-defect-displace-ments=\<value\> | calculate an exponentially decaying displacement with the distance around a given centre, including a cosine modulation of the radial displacement | f1,f2,f3,f4,f5,f6,f7,f8<br><br>f1: point defect centre x, relative coordinates<br>f2: point defect centre y, relative coordinates<br>f3: point defect centre z, relative coordinates<br>f4: displacement amplitude (A)<br>f5: sigma width (A)<br>f6: cosine ampli-tude, fraction of f4 as a relative number between 0 and 1, 0 means pure exponential, 1 means cosine with exponential-ly dampened amplitude<br>f7: cosine period width (A)<br>f8: cut-off radius (A) | ✗ | |
| --preview=\<value\> | generate a preview file and display the preview in CrystalMaker or Vesta<br><br>(Mac OS X so far) | s<br><br>s: CM or VESTA or Safari | ✗ | --preview=VESTA<br><br>sends output to Vesta, choose CM for CrystalMaker preview and Safari for a Safari preview |

| option | description | value format/ choices | mult iple | example/ explanation/ comment |
|---|---|---|---|---|
| [—preview-file=<value>]<br><br>obsolete in v2.5:<br><br>use —write-format | | | | |
| --prompt | wait for user input after each preview output | - | ✗ | --prompt |
| --reverse | put atom list in reverse order | - | ✗ | --sort=x --reverse<br><br>first sort according to ascending x-coordinate, then reverse output |
| --shuffle | shuffle atom list | - | ✗ | —shuffle |
| --unformatted | write unformatted output in combination with EMS or XMS output | - | ✗ | --unformatted  —write-format=XMS |
| --slice=<value><br><br>-L <value> | slice the super-cell in z direction | i<br><br>i: number of slices | ✗ | —slice=10 —write-format=XMS,CIF<br><br>create 10 slices in z direction with the thickness dz=0.1 in relative coordinates, slices are labelled by a running counter starting from 001,<br><br>specify only the prefix for the output, e.g. -o slc will lead to the output of slc001.cel, slc002.cel, …, slc001.cif, slc002.cif, … for —output-format=XMS,CIF |

# Further Notes

CellMuncher does not rely on strictly formatted input. Since most operations do not commute, it is safe to perform them one by one in a sequential manner.

# Examples

**Example 1:** Create a rectangular cell of Hübnerite ($MnWO_4$) in two steps.

Download an ICSD structure file from the ICSD database and convert it to an EMS super cell using BuildCell. Note: You may need to check the occupancy values and the Debye-Waller factors in the EMS file after conversion!

```
BuildCell --cif=MnWO467906.cif --output=MnWO467906.cel
```
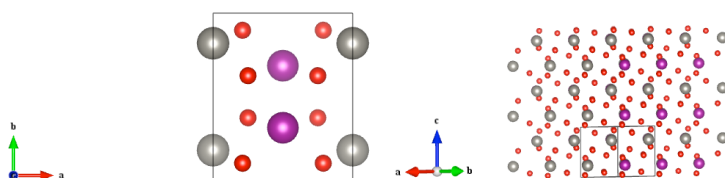


Fig.: P2/c unit cell of $MnWO_4$ (converted from ICSD file No. 467906 using BuildCell) and a repeated cell in <110> viewing direction. The monoclinic unit cell dimensions are a=4.83000, b=5.76030, c=4.99400, alpha=90.0000, beta=91.1420, gamma=90.0000.

Then use the --create-block option to make a rectangular super-cell of a size of 3x3x6 $nm^3$ with the <110> direction parallel to the z direction of the unit CellMuncher. Note that a, b, and c are all different and that beta is not 90°! To construct a rectangular cell you need the new z-direction, <110> in this case, and one orthogonal direction. One possible direction is $<|a||c|/|a|^2\cos(a,c),0,1>=$ <0.020607064,0,1>, which is parallel the plane normal to the ab-plane. CellMuncher will suggest a direction close to the chosen one in case it is not orthogonal. In fact, any direction which is not parallel to the new z-direction will do. Therefore, we can choose <0,0,1> and let CellMuncher do the rest:

```
CellMuncher --preview-file=VESTA --create-block=1,1,0,0,0,1,3,3,6 --input=MnWO467906.cel --output=Mn-WO467906-110-super-cell.cel
```
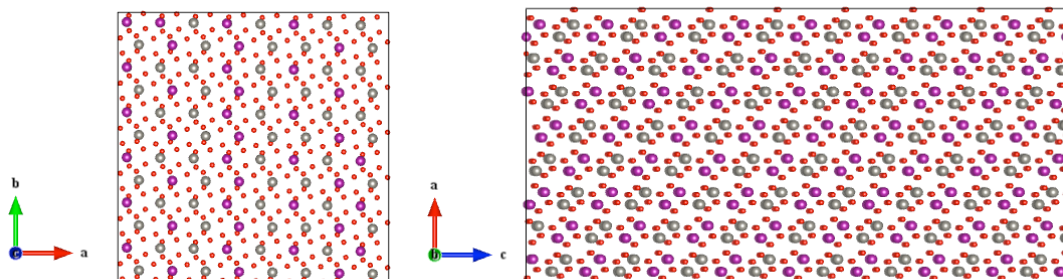
Fig.: Super-cell of MnWO$_4$ with the z-direction parallel to the <110> viewing direction of the original unit cell.

**Example 2:** A rectangular cell of hexagonal GaAs in more steps

Create an EMS super cell containing a unit cell of hexagonal GaAS and name the file GaAs_wurtzite_unitcell.cel

```
# GaAs wurtzite unit cell data
 0 0.624 0.382 0.382 120 90 90
Ga 0.0    0.0       0.0      1.0    0.006  0.0    0.0    0.0
Ga 0.5    0.666667  0.33333  1.0    0.006  0.0    0.0    0.0
As  0.375  0.0       0.0      1.0    0.0055 0.0    0.0    0.0
As 0.875  0.666667  0.33333  1.0    0.0055 0.0    0.0    0.0
*
```

Now create a shell script with the CellMuncher commands

```
#!/bin/bash

# repeat hexagonal cell, open preview in CrystalMaker and prompt user to continue
CellMuncher --repeat=x,2 --repeat=y,5 --repeat=z,5 --override  --prompt --preview=CM --input-file=GaAs_wurtzite_unitcell.cel --output-
file=tmp001.cel

# swap axes, preparation for orthogonalisation, only beta can be orthogonalized,
# open preview in CrystalMaker and prompt user to continue
CellMuncher --swap-axes=xy --override  --prompt --preview=CM --input-file=tmp001.cel --output-file=tmp002.cel

# orthogonalisation, only beta can be orthogonalized
# open preview in CrystalMaker and prompt user to continue
CellMuncher --orthogonalize-plane=xz,10 --override  --prompt --preview=CM --input-file=tmp002.cel --output-file=tmp003.cel

# repeat in x-direction, make 20 nm thick, y and z are periodicm
# open preview in CrystalMaker and prompt user to continue
CellMuncher --debug --delete-duplicate --repeat=x,10  --override  --prompt --preview=CM --input-file=tmp003.cel --output-
file=tmp004.cel

# swap axes, so that z is the large dimension and x > y,
# open preview in CrystalMaker and prompt user to continue
CellMuncher --swap-axes=xz --override  --prompt --preview=CM --input-file=tmp004.cel --output-file=tmp005.cel
```
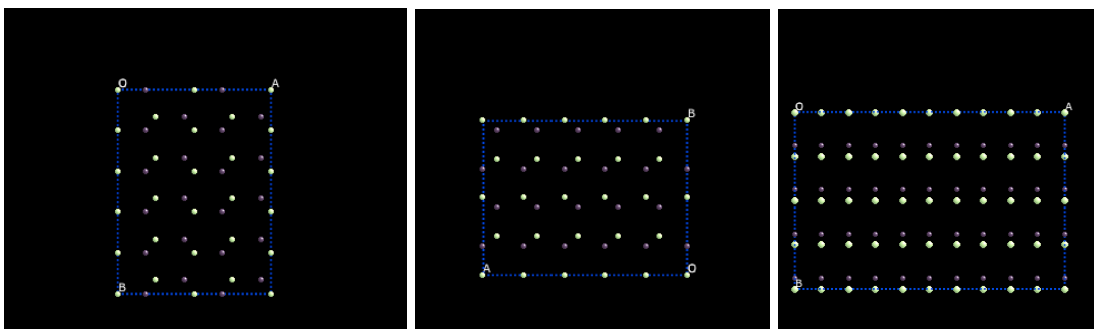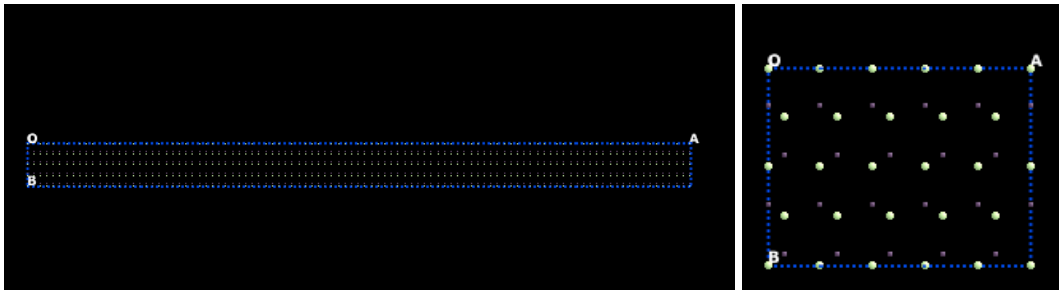
then execute the script file. You may need to change the path to the bash shell in the first line of the script according to your installation path.

CrystalMaker will show the preview of the 5 steps:

## Known Bugs

Option processing is order sensitive! It is wise to do only one operation at a time.
Leave a blank between option specifier and argument for the single character short option.

## Author/Contact

CellMuncher version 2.5, Nov 2014, Copyright © L. Houben
CellMuncher comes with ABSOLUTELY NO WARRANTY.

mail to the author