



GUI Manual

Amsterdam Modeling Suite 2025.1

www.scm.com

Apr 04, 2025

CONTENTS

1	General	1
1.1	Introduction	1
1.2	Release 2025.1	1
1.3	Release 2024.1	2
1.4	Release 2023.1	5
1.5	Release 2022.1	7
2	Set up	13
2.1	Mouse	13
2.2	Running Remotely	13
2.3	OpenGL3.2+, fallback mode	13
2.4	Font issues	14
2.5	AMSjobs Queues	14
2.6	Quantum ESPRESSO	14
3	SCM menu	15
4	Background processes with no interface	17
4.1	scmd: communication between GUI modules	17
4.2	scmjobd: monitor running jobs	17
5	Viewing your molecule	19
5.1	Molecule style: Balls and Sticks, Sticks, or Wireframe	19
5.2	Camera position: view direction	19
5.3	3D	20
5.4	Fly to selection	20
5.5	Reset View	20
6	Mouse Interaction	21
6.1	Rotate, Translate and Zoom	21
6.2	Selecting	22
7	Keyboard Shortcuts	23
7.1	AMSinput	23
7.2	AMSview	25
7.3	AMSmovie	25
7.4	AMSlevels	26
7.5	AMStail (Logfile)	26
7.6	AMSoutput	26
7.7	AMSspectra	26
7.8	AMSbandstructure	27

7.9	AMSdos	27
7.10	AMSjobs	27
8	Selecting	29
9	AMSPreferences	31
9.1	General	31
9.2	Restore defaults	31
9.3	Panels	31
9.3.1	General	32
9.3.2	Atoms and Bonds	32
9.3.3	Colors	32
9.3.4	Atom Colors	32
9.3.5	Bond Colors	32
9.3.6	AMSview	32
9.3.7	AMSjobs	32
9.3.8	AMSlevels	32
9.3.9	Environment	32
10	AMSinput	33
10.1	Input: options	33
10.1.1	Normal Options	33
10.1.2	Expert Options	34
10.1.3	Units	34
10.2	Input: buttons	34
10.2.1	Searching	34
10.2.2	The “>” details buttons	35
10.2.3	Using the “?” info buttons	35
10.3	Searching	35
10.3.1	Panel search details	36
10.3.2	Key search details	36
10.3.3	Documentation search details	37
10.3.4	Molecule search details	37
10.4	Crystals and Slabs	37
10.4.1	Periodicity	37
10.4.2	Building crystals	38
10.4.3	Build slabs	38
10.4.4	Crystal menu	38
	Generate Super Cell...	38
	Map Atoms To (-0.5..0.5)	39
	Map Atoms To (0..1)	39
	Set (0.5, 0.5, 0.5)	39
10.5	Builder (packmol)	39
10.5.1	The five starting modes of the Builder	39
	Note on non-orthorhombic systems	40
10.5.2	The Builder has flexible input	40
10.6	Building molecules	41
10.6.1	Bonding mode	41
10.6.2	Atom Details: Connectors (valency) and Lone Pairs, Atom mass	41
10.6.3	Pre-optimization	41
10.6.4	Geometry adjustments	42
	Sliders	42
	Edit box	42
	Centroids	42

	Move an atom (possibly perpendicular to the screen)	43
	Rotate or translate the selection	43
	Coordinates	43
	Z-Matrix	43
	Set Center Of Mass	43
	Set Origin	44
	Symmetry	44
	Rotate 90	44
	Align	44
	Mirror	45
	Update coordinates via AMSmovie	45
10.6.5	Bonds	45
10.7	Tuning Geometry: translation, rotation, position with respect to plane	45
10.7.1	Atoms to move	46
10.7.2	Translate atoms, direction	46
10.7.3	Translate atoms, target	46
10.7.4	Rotate atoms	47
10.7.5	Position atoms with respect to a plane	47
10.8	Importing and exporting your molecule	48
10.8.1	Importing your molecule	48
10.8.2	Copy/Paste between GUI modules, XYZ coordinates, SMILES or InChI strings	49
10.8.3	Exporting your molecule	49
10.9	Regions	50
10.10	Proteins (from PDB or mol2 files)	51
10.11	Presets and Defaults	51
10.11.1	When are presets used	52
10.11.2	Saving presets	52
10.11.3	Matching presets with modules / tasks	53
10.11.4	Default values used	53
10.11.5	Color Code	53
10.11.6	Input options remarks	54
	Spin and Occupation	54
10.12	Update geometry dialog	54
10.12.1	Meaning of restart options	55
11	AMSjobs	57
11.1	Jobs	57
11.1.1	Job list: per folder or global	57
11.1.2	Job status (including WARNING and ERROR info)	57
11.1.3	Selecting Jobs	58
11.1.4	Running a job: .run and .job	58
11.1.5	Job files	58
	Local files	58
	Remote files	59
11.1.6	Clean Up	59
11.1.7	Archive jobs	59
11.1.8	Import / Export jobs	59
11.1.9	Find job files	60
11.1.10	Test Job	60
11.2	Queues	60
11.2.1	Interactive Queue	61
11.2.2	Sequential Queue	61
11.2.3	Setting up your own Queues	61
11.2.4	Cloud computing	63

11.2.5	Dynamic queues	65
11.2.6	Example Queue configurations for SGE and PBS	66
11.3	Tools: set up many jobs and collect data from many jobs	67
11.3.1	Prepare: set up many jobs at once	67
11.3.2	Report: collect data from many jobs	68
11.4	Series of jobs depending on each other	68
12	AMsview	69
12.1	Visualization methods	69
12.1.1	Spinor: spin magnetization density	70
12.2	Fields	70
12.2.1	Steric Interaction	70
12.3	Properties	71
12.4	Comparing data from several molecules	71
12.5	Temporary Files	72
12.6	Calculating Fields	72
13	AMsmovie	73
13.1	Geometries	73
13.2	Scalar Atomic Properties	73
13.3	Atomic Vectors	73
13.4	Graphs	74
13.5	Temperature Profile	74
13.6	Trajectory Analysis	74
13.7	Export trajectories to other formats	75
14	KFbrowser	77
14.1	Non-expert mode	77
14.2	Expert mode	77
15	Spreadsheets (.xlsx)	79
15.1	Export AMS results to spreadsheet format	79
15.1.1	Create a .xlsx file for a single AMS result	79
15.1.2	Generate spreadsheet by default for all jobs	80
15.1.3	Compare multiple AMS calculations in a spreadsheet	80
15.1.4	Units	80
15.1.5	Distributions and spectra	81
15.1.6	Evaluate formulas with LibreOffice	81
15.2	Results available in spreadsheet format	81
15.2.1	Engines	82
15.2.2	Tasks	82
15.2.3	Single-point properties	82
15.2.4	Geometry optimization properties	83
15.2.5	NEB properties	84
15.2.6	MD properties	84
15.2.7	System	85
15.2.8	General	86
16	Conformers	87
17	Fukui Function	89
18	Minimum Energy Crossing Point	91
19	Quantum ESPRESSO	93

20	VASP via AMS	95
20.1	VASP support in the Amsterdam Modeling Suite	95
20.1.1	General	95
20.1.2	VASP as an external engine	95
20.1.3	Tutorials	95
20.2	Setting up a VASP calculation	95
20.2.1	VASP executable and number of processes (parallelization)	95
20.2.2	Pseudopotentials and PAW potentials (POTCAR)	96
20.2.3	k-point sampling (KPOINTS)	96
20.2.4	Other settings (INCAR)	96
20.3	Running VASP via AMS	97
20.3.1	Running VASP with AMSjobs	97
20.3.2	Preprocessing the input	97
20.3.3	Scripting support	97
20.4	VASP via AMS without the GUI	97
21	ParAMS	107
21.1	General	107
21.1.1	Start ParAMS	107
21.1.2	Overview	107
21.1.3	File handling	108
	Open	108
	Merge	108
	Save	108
	Revert	108
21.1.4	Tables	108
	Header menus: sort and filter	109
	Search	109
	Selection	109
21.2	Import reference calculations	109
21.2.1	Jobs	110
21.2.2	Training set	110
21.2.3	Validation Set	112
21.2.4	Mouse interaction	112
21.2.5	Editing	112
21.2.6	Warnings and errors	112
21.2.7	Reference data	113
	Running reference jobs	113
	Importing results from reference jobs	113
21.3	Parameters	113
21.3.1	ReaxFF parameters	113
	lgDispersion, ACKS2 or eReaxFF	113
	Add a ReaxFF block of parameters	113
	Initialize ReaxFF parameters	113
	Edit ReaxFF header	114
	Editing parameters and optimization details	114
21.3.2	ASE Parameters	114
21.3.3	No Parameters	114
21.4	Graphs	114
21.5	Settings	114
21.6	Results	115
21.7	Info	115
22	AMScrs: COSMO-RS	117

22.1	Menu Commands	117
22.1.1	File menu	117
22.1.2	Compounds menu	118
22.1.3	Method menu	119
22.1.4	Properties menu	119
22.1.5	Solvent Optimizations menu	121
22.1.6	Analysis menu	121
22.1.7	Graph menu	121
22.1.8	Help Menu	122
23	AMSkinetics (MKMCXX and Zacros)	123
23.1	MKMCXX Microkinetics	123
23.2	Zacros Kinetic Monte Carlo	123
24	GUI Environment Variables	125
25	AMSpackages	129
25.1	General	129
25.1.1	Installing packages	130
25.1.2	Removing packages	130
25.1.3	Reinstalling packages from a previous version of AMS	130
25.1.4	Selecting an alternative package source	131
26	Required Citations	133
26.1	Authors and General References	133
26.2	External programs and Libraries	133
	Index	135

GENERAL

1.1 Introduction

This document will provide background information for the Graphical User Interface, part of the AMS package.

The purpose of this document is to give more detailed background information on how the GUI operates. It will not explain how to use it in detail. For that reason, we strongly suggest that before reading this document, you first check the [GUI tutorials](#).

1.2 Release 2025.1

- General:
 - Open JupyterLab notebooks (.ipynb files) via SCM menu
 - License Info showing the dirac check output in the Help menu
 - Command to start Terminal on a Mac now directly sets the environment (no need to copy/paste it any more)
 - Improved display of coordinate axes, now showing at origin and /or in the left-bottom corner. The choice (in the View menu) will be remembered.
 - Improved display of lattice vectors: added an option to change the thickness (in the View Periodic menu). The choice will be remembered.
- AMSjobs:
 - Open JupyterLab notebooks (.ipynb files) by clicking on them in the AMSjobs window
 - Run Jupyterlab notebooks without GUI either locally or remote (like any other job), results in the .results folder
 - Add # dependency lines to your ipynb notebook file to use a dependency (like the dependency lines in the AMS run scripts)
 - Editing a queue in AMSjobs now has a Save As... option
- AMSinput:
 - Editing of the main pages for the engines is no longer possible
 - Startup much faster
 - Back and forward buttons on panels (arrows on the top left of the panels)
 - Input options panel (the right half of the AMSinput window) resizable and improved dynamic layout with font changes

- Menu commands with shortcuts to adjust layout: Molecule only, Input options only, or both
- Arrow up and down jump to previous/next non-default item in Expert panels
- ROKS support
- ROSE support
- ASE calculators: option to analyze a selected calculator .py file, and use the defaults if desired.
- QE work function support.
- QE interface now supports slabs and chains as well.
- Builder now uses packmol via the python packmol implementation, the GUI for this has changed.
- Region manipulation now also part of the undo stack.
- Support analytical phonons with QE.
- Support CREST metadynamics.
- Support PLUMED.
- AMSmovie:
 - MD Analysis: support setting charges and masses per atom for MSD.
 - MD molecule counts: now also try to use Time on the horizontal axes, if ‘Try To Use Time On X-axes’ is checked in the Graph menu.
- ParAMS GUI:
 - Back and forward buttons on panels (arrows on the top left of the panels).
 - SCM_PARAMS_RESULTS_ONLY_SIZE env var: if input file size bigger than this value, offer to load results only (for speed). A value of 0 is the same as not setting it (and thus always loading the input).
 - Save the Reference or ParAMS run script for selected jobs. The run scripts can be opened by AMSinput.
- KFBrowser:
 - Plots in the ams.rkf Plot section now are available in the non-expert mode as plots.

1.3 Release 2024.1

This release no longer supports old style ReaxFF input and output (the non-AMS version with the .rxkf result files). This release only supports QE via AMS, the old QE interface has been removed.

In comparison to AMS-GUI 2023.1, the 2024.1 release offers many bug fixes and the following new changes or main functionality:

- General:
 - Show Info: Axes And Planes menu in View menu (if present): show info about the vector defined by selecting 2 atoms:
 - * length of vector
 - * angle with x, y and z-axis
 - * projection on x, y and z-axis
 - * also angle and projection on user defined vector and user defined plane (user specifies normal vector)
 - * for periodic systems, similar info for angles and projections on the lattice vectors

- * for periodic systems, also show result using shortest image convention
- In Show Info view: pin information so more info can be seen at the same time.
- The Show Info view will update in AMSmovie when playing a movie.
- AMSjobs:
 - Command to copy paths for local or remote jobs, in the Jobs menu, so they can easily be used in a terminal.
- AMSinput:
 - Export a job to a PLAMS Python script (only applicable to single AMS jobs, not more complicated tasks like COSMO-RS Compound or Conformers).
 - No longer supports opening old style ReaxFF input (the non-AMS version)
 - SCRF support removed (as ADF no longer supports it)
 - Support QE engine (instead of the old non-engine QE)
 - Improved selection of ReaxFF force fields (note double click in table is easiest to use)
 - Support reaxff.charges option to define charge per atom (edit in AtomDetails).
 - Support defining TS reaction coordinate using an initial and final system
 - Support Repulsion engine add-on
 - DFTB Single Point more button links to Fragment panel
 - Builder now puts new molecules in region Builder instead of the confusing Auto_Generated region
 - Import molecule(s) from file(s): use filename or formula as name of the imported molecule (if importing into an empty molecule space). Note the molecule name is visible only when multiple molecules are present (via Edit -> New Molecule for example, or when importing multiple files.).
 - Import molecules from files: now you can select multiple files to be imported at the same time, each in its own molecule space.
 - Import molecule using SMILES: import multiple SMILES (separated by spaces) as multiple molecule. And use SMILES or formula as name of the imported molecule (if importing into an empty molecule space). Note the molecule name is visible only when multiple molecules are present (via edit -> New Molecule for example, or when using multiple SMILESs).
 - Import molecule from a forcefield.rkf file: read forcefield types, charges and patch file as well
 - Use LoadSystem key via Coordinates panel
 - When showing multiple molecules in a grid, clicking in an empty grid space will create a new molecule
 - Powder X-Ray Diffraction (XRD) available from the View menu. This requires pymatgen which can be installed from SCM → Packages
 - Delete All Molecules: same as New, but only for the molecules (leaves the changed settings intact)
 - Add TSRC distance term using a dummy atom
 - BAND restart DOS and band structure support
 - Paste multiple values in Atom details fields: depending on number of pasted values update all atoms in atom selection (also make sure current field is in selection), all atoms, or only current field.
 - Improve reading of run scripts (especially picking up input options for non-unique keys)
 - Paste engine block only (starting with Engine and ending with EndEngine)

- Import Engine Settings (similar to pasting the engine block only), from .run scripts, ams.rkf (the userinput variable is used), or engine.txt as made by params. Intended use is with MD Active Learning, to use the resulting engine after training in another job (just use Import Engine Settings and select the ams.rkf file from the active learning job). The command lives in the File menu.
- When using LoadEngine, switch to a dummy LoadEngine module to make it clear for the user and to prevent engine settings to be changed. LoadEngine could be used when importing a run script that is using it.
- Offer to try to read run script when pasting an AMS job script (recognized by something like ams << eor and matching eor at end of input, as in all example run scripts and run scripts made by AMSinput)
- Support MD Reaction Boost
- MD Active Learning module (which is Simple Active Learning), see also Import Engine Settings
- Support pNMR
- Perturb atoms randomly by 0.01 Angstrom via the Edit→Perturb option.
- Undo/Redo work only on the molecule structure, and now also only when the focus is on that panel, thus while editing the molecule or after clicking in it, and NOT when editing input options.
- AMSmovie:
 - PES Scans: improve labels below molecule
 - PES Scans: option to reorder all frames by PES points, useful if some PES points are redone at the end
 - No longer support .rxkf files
 - Analysis MSD supports properties, making it possible to compute ionic conductivity
 - Active learning trajectories: in File -> Related Files the trajectories of related trajectories are available (stepX_attempt_Y_simulation, and final simulation).
- AMSview:
 - NCI diagram support for ADF results
 - Improve ease of positioning planes (contour planes and colored planes), via explicit setting of normal vectors and a point on the plane
 - Add visualization along a line: the values of the field, and these values integrated along the line. The resulting line graph may be saved as image or as XY values.
 - Add visualization along a line: now also the values integrated over a plane perpendicular to the line, and these may also be integrated over the line (thus integrating 3D space);
 - Add special Grid Cell that covers at least one cell spanned by the lattice vectors (offset to match the cell as shown by AMSview).
- AMSspectra:
 - NMR: support visualization of pNMR results
- AMSdos:
 - Support display of DOS info from QE using QE as AMS engine, older QE stuff no longer supported.
- AMSbands:
 - Support display of DOS and BAND structure info from QE using QE as AMS engine, older QE stuff no longer supported.
- ParAMS GUI:
 - Improved selection of ReaxFF force fields

- Generate validation set now by JobID as an option
- Option to load a parameter (interface) yaml file (it will replace any parameter data already loaded).
- Support Machine Learning training, input and visualization
- When opening a MD Active Learning job (simple active learning) the last training result will be shown.
- Option to open AMSinput with an engine with training results (based on the engine.txt files)

1.4 Release 2023.1

In comparison to AMS-GUI 2022.1, the 2023.1 release offers many bug fixes and the following new main functionality:

- General:
 - Two new periodic view options are added. The three available options are now:
 - * (old) Repeat unit cells
 - * (new) A layer of x (default = 2.0) Angstrom around the unit cell
 - * (new) Complete the molecules that are partially inside the unit cell
 - The 2.0 Angstrom layer is the new periodic view default, and the periodic display threshold was increased.
 - New atom selection methods are added
 - * Select atoms by index
 - * Select atoms with an interactive box widget and box parameters
 - * Select atoms by region
- AMSjobs:
 - Apply to all check when doing something to multiple jobs
 - Report to tsv file now always includes filename from which results are collected
- AMSinput:
 - Presets have been changed: check the documentation for details.
 - Builder supports explicit bonds from .in files
 - ACE Reaction Network support
 - ASE engine support
 - Improve import of job from run scripts
 - Show molecule formula per region on Regions panel
 - Sort coordinate table by coordinate or element type
 - PES scan per x/y/z coordinate supported
 - Multiple NEB intermediates
 - Dynamic factors in hybrid engine
 - MD reactor (Martinez nanoreactor)
 - GW for BAND
 - MP2, RPA and double hybrids for BAND

- Conformers support, both generating conformers and using conformers with further calculations using the conformers.rkf file
 - MD analysis jobs supported as part of the main job
 - MD analysis: automatically run over all elements for RDF and ACF if requested
 - Save As: menu shortcut
 - Preoptimizer now optionally honors constraints
- AMSmovie:
 - The graphs use now Time on the X-axis instead of Frame, if Time is available for plotting (as in MD results)
 - Analysis of MD trajectories now remember settings and shows last result, if present
 - Analysis: multiple calculations for one trajectory of same type (multiple RDF etc) now supported
 - Analysis: calculated the RDF/ACF/MSD/Histogram in a separate job, possibly on some remote system
 - ACE reaction support
 - ChemTraYzer2 support: handle the molecule population and event files, producing graphs in addition to the tables
- AMSspectra:
 - Support Conformers via amsbatch.rkf
 - Support BAND GW
- AMSbands:
 - Option to shift the Fermi energy to zero.
- AMSdos:
 - Use stacked curves for the partial DOS (similar to the DOS in AMSbands).
 - Option to shift the Fermi energy to zero.
 - Option to mirror beta dos.
- AMSkinetics:
 - Import PES Exploration binding sites result to obtain a complete Zacros system setup
- AMSview:
 - Visualization of atom vector properties
 - Visualization of NBO natural charges
 - Visualization of RISM Guv RDF
- AMSoutput:
 - Added some DFTB menu shortcuts
 - Added generic Header menu with shortcuts to all headers (any text between bars)
- ParAMS GUI:
 - GLOMPO support
 - GUI to set up the job options (in the Settings tab)

1.5 Release 2022.1

In comparison to AMS-GUI 2021.1, the 2022.1 release offers many bug fixes and the following new main functionality:

- General:
 - All GUI modules transfer job to ParAMS GUI via menu command (cmd/ctrl-T)
 - (Linux only) Better dependency handling. Incompatible libraries from \$AMSBIN/lib no longer need to be removed.
 - (Windows only) Option to use a different method of GUI inter-communication. The default uses sockets that can be blocked by security software. Set the environment variable SCM_SCMD_PIPE to use the new alternative that uses named pipes.
- AMSinput:
 - Open .run files (also importing many options)
 - Import system block from .run files
 - Open and export system block (as in run scripts) as .in files
 - AMS Lattice scan
 - GFNFF engine
 - Lennard-Jones engine
 - APPLE&P Force Field type
 - FCF support updated
 - GW support updated
 - DFTB fragments
 - QTAIM / IQA support updated
 - FQFMu forcefield for FQQM.
 - FQQM and FDE combination
 - Workflows target with OLED tasks
 - Region constraints
 - Constraint for all bonded pairs in selection
 - Constraints work with shortest PBC distance
 - Conceptual DFT / Fukui improvements by Laurent Joubert
 - Autografs update SBU database
 - Single Point option for COSMO-RS compound
 - Crystal slicer automatically converts to conventional cell and aligns axes
- AMSjobs:
 - Better support for running PLAMS jobs
 - Improved queue settings interface
 - Option to test your remote queue settings
 - Better support for larger runscripts for queuing systems

- AMSspectra:
 - Support for GW improved
- Preferences:
 - Added region display radius setting
- AMSlevels:
 - DFTB fragment orbitals
- AMSview:
 - Visualization of OLED properties
 - Histograms of atom properties
 - DFTB fragment orbitals
- AMSbands:
 - Support for spin-orbit coupling results
- AMSmovie:
 - Bond source can be chosen. Choices are: auto (prefers stored bonds), guessed, initial, or no bonds.
 - PES Exploration/EON energy landscape. Addition of Arrhenius prefactors to info and Fragmented states.
 - Acerxn support
 - ChemTraYzer 2:
 - * Input
 - setup input (based on json description and .def file)
 - AMSinput style unit changes and revert to default (right click)
 - when opening the GUI, update the input fields with values as found in the chemtrayzer2/job file, if it exists
 - * Running
 - run ChemTraYzer 2, job
 - output and results are in chemtrayzer2 folder next to the ams.rkf file
 - progress dialog while running, cancel button to interrupt
 - errors from run are reported
 - * Output
 - automatically show the results in tables when available
 - right bottom of table: button with name of table, click it to open by the OS (on MacOS open the .csv file in Numbers for example)
 - tables can be sorted per column
 - tables can be searched, with some advanced search options. use help balloon to get info on search syntax
 - feedback on search in status line (number of items found, column searched)
 - click, shift-click, control (or right) click to adjust selection in tables
 - selection in upper table searches for matching evens in lower table

- selection in lower table shows corresponding atoms (hiding other atoms) in corresponding (first) frame in the Movie window
- left-right arrow keys to skip through movie frames
- up-down arrow keys to move selection up or down (thus go through different events for example)
- escape clears selection, or the search if it has focus
- if no selection, escape clears the search
- Mean Square Displacement (MSD) via analysis program
- ParAMS GUI:
 - General
 - * ‘All’, ‘Jobs’, ‘Training Set’, ‘Validation Set’, ‘Engines’: table with training data
 - * Molecule window: show system for selected entries
 - * ‘Info’: detailed info for any entry in job collection or training data
 - * ‘Parameters’: table with parameters, ranges and other details to be optimized
 - * ‘Params Configuration’: show and set up the ParAMS job details
 - * ‘Graphs’: results of optimization in graphs
 - * ‘Results’: text output files
 - * import old style reaxff geo/params files
 - * revert option to ditch changes due to editing or handling results
 - Running
 - * run the ParAMS job via AMSjobs
 - * opening a ParAMS job will automatically read the results if available
 - * live update of the results
 - * running and live update also work for running on remote machines (as all other jobs in AMSjobs)
 - * restart a previous run when using the CMA optimizer
 - Training Data
 - * different tabs to show all of these or just one subset (‘All’, ‘Jobs’, Training Set’, ‘Validation Set’, ‘Engines’)
 - * filter by type of entries (Energies, Bonds, PES, Jobs, Engines, ...)
 - * when multiple systems are relevant for an entry, use arrow left-right to move through them
 - * add jobs from inside the Params GUI, or via any other GUI module (cmd/ctrl - T)
 - * use the ParAMS importers to generate training set entries when importing a job
 - * select atoms to set the proper index when creating some expressions
 - * menu commands to add most common expressions in the training set
 - * edit all (non-output) items in all tables
 - * editing via AMSinput for jobs and engines
 - * smart editing of multiple items in many cases (expressions for multiple items like relative energies, or deactivate all selected parameters for example)
 - * tab-completion to quickly enter Job IDs

- * energy expressions may be balanced automatically (as they correspond to a reaction typically)
- * set up training set and validation set (manual or random)
- * merge training sets / job collections
- * edit all JobIDs at once: prefix or change
- * when results are available the contribution to errors is shown with the matching item
- * results are live-updating
- Parameters
 - * ReaxFF, xTB and Lennard-Jones parameters supported
 - * detailed information about meaning of parameter and optimization range etc
 - * adjust parameter optimization range Min and Max
 - * generate new ReaxFF blocks, initialize ReaxFF blocks
 - * parameters are live-updating
- ParAMS Configuration
 - * set up the params.conf.py file to use for running
 - * predefined common options via menus
 - * fully editable by user if desired
- Graphs
 - * relevant graphs are shown automatically when running
 - * graphs are live-updating
 - * validation and training set graphs together, compare with first iteration
 - * graphs of PES data (like energy vs distance or angle etc)
 - * interaction in graphs (click on an item to select matching training set entries, and select relevant atoms in molecule display if applicable)
 - * save graphs as image or xy from File menu
- Results
 - * show text output easily accessible via menus
 - * text is live-updating
- Tables
 - * tables can be sorted per column
 - * tables can be searched. use help balloon to get info on search syntax
 - * feedback on search in status line (number of items found, column searched)
 - * click, shift-click, control (or right) click to adjust selection in tables
 - * up-down arrow keys to move selection up or down
 - * escape clears selection, or the search if it has focus
 - * if no selection, escape clears the search
- AMSknetics

- Renamed module from Microkinetics
- Added support for the Kinetic Monte-Carlo program Zacros.

2.1 Mouse

A three-button mouse is also very convenient for using the ADF-GUI, and on a Mac you can use a Magic mouse for this purpose. To get three buttons (instead of the standard one or two), download and install one of the free utilities BetterTouchTool or MagicPrefs, and configure it to add a middle click.

2.2 Running Remotely

To use AMSjobs with remote machines, you need to set up ssh first. You should take care to configure things such that you do not need to type a password when you access your remote machine. To do this you need to:

- create **keys**,
- run an **ssh agent**,
- add your public key to the **authorized_keys** file on the remote machine.
- set up a ssh config file, if needed. This allows you to automatically set options, like the user name to use on the remote machine.

Thus, users (and AMSjobs) should be able to use ssh to log in to the remote machine without ever needing to enter a password.

If you are using OpenSSH (typically on Linux machine or MacOSX) you can make the communication with the remote machine much more efficient by setting the `SCM_SSH_MULTIPLEXING` environment variable to yes (in the GUI Preferences module).

AMSjobs does not store passwords, it always uses the ssh command to communicate with remote systems.

For more information, consult your ssh documentation or one of the many guides on the internet.

2.3 OpenGL3.2+, fallback mode

The GUI needs OpenGL 3.2 or later. Using it results in a much better performance, especially for bigger systems. Also some new graphical features depend on it. That OpenGL version may not be available when you are running on old hardware or OS, or when you are running remotely via X11 with GLX or some remote desktop solutions.

When the GUI detects a problem with the OpenGL version it will automatically activate a fallback option. That will normally just run, but it will be significantly slower, and of course the features depending on OpenGL 3.2 will be missing. We advise that you try to update your system to solve this issue. For remote use of a linux machine a solution could be the use of VirtualGL.

For more details please see the [Installation manual](#).

2.4 Font issues

The GUI used unicode characters in many places. For example, the proper characters for Angstrom, degrees Celsius, and many more. You can see examples by starting AMSinput, selecting a Geometry Optimization, and then going to the details page (by clicking the ... button). There you should see a couple of Angstrom symbols if everything is working as intended.

On most machines this will just work out of the box. If not, the issue might be that the default font on the machine does not support such characters. Try installing a font line DejaVU Sans and Monospace DejaVu. On Centos you can do this (as root) with the command::

```
yum install dejavu-sans-fonts dejavu-sans-mono-fonts dejavu-serif-fonts
```

2.5 AMSjobs Queues

Defining proper queues in AMSjobs will making the GUI much easier. For example, you can run as easily on a remote compute cluster that you have access to as on your local desktop. AMSjobs will handle all the details like transferring input and output files, and you can even monitor the progress of running jobs as if they were running on your local machine.

So you should take some time to set up the queues correctly. It is possible that there are predefined queues on your remote machine. Then you can configure AMSjobs to automatically use those queues. You can find a description of the AMSjobs queues and how to set them up later in [this manual](#) (page 57).

2.6 Quantum ESPRESSO

Quantum ESPRESSO is an open source package that can perform plane wave SCF calculations, among other things. See www.quantum-espresso.org (<http://www.quantum-espresso.org>) for details.

Quantum ESPRESSO is *not* included in the main AMS distribution, but can easily be installed through the AMSpackages package manager.

The easiest way to install it is just to open AMSinput and go to the Quantum ESPRESSO section. This will use AMSpackages to check if QE has already been installed, and if not it will install it using AMSpackages. An environment variable `SCM_PKG_QE` will be set pointing to the install location.

To force installation of Quantum ESPRESSO (for example to re-install it to make sure you have a clean version), use the **SCM → Packages** command, and use the AMSpackages module to remove and reinstall Quantum ESPRESSO.

From the command line (e.g. on a remote machine), you can install the Quantum ESPRESSO package using the following command:

```
"$AMSBIN"/amspackages install qe
```

SCM MENU

All GUI modules have an SCM menu. The SCM menu is the leftmost menu in all modules, and is represented by either the SCM-logo or the text SCM.

In general, when you select a command from the SCM menu you will switch to the selected module, showing the same job as you have open in the current module. If such a module is already running with that particular job open, it will be brought to the front. If you use the SCM menu to open the same module as is currently active, it will start another blank copy of this module. Thus the SCM menu is very convenient to quickly switch between the already open windows belonging to the same job.

As a shortcut you can also right-click on a job name in AMSjobs, and the SCM menu will appear with the job you clicked on selected.

In AMSjobs the currently selected job will be used with the SCM menu. Thus, if you have one job selected, using the **SCM → View** command will start or switch to AMSview, showing the results of the selected job. If you have no job selected, or more than one, the selected module will start blank.

One important exception within AMSjobs: the Input module will always start a new blank version. If you wish to show the input GUI for the selected job you need to click in the icon in front of it (reading 'ADF', 'BAND', or some other job type).

The SCM menu also allows you access to preferences for that are used by all GUI modules: **SCM → Preferences**

In some cases a GUI module is able to open several file types. For example, AMSmovie can open the logfile or the .t21 file of a calculation. By selecting a file in AMSjobs (after opening the job by clicking on the triangle) you can force AMSmovie to open the .t21 or the .logfile. In a similar way, you can force AMSview to open a .t21 or a .t41 file.

The **SCM → Quit** and **SCM → Quit All** menu commands will close windows: the Quit command will close all windows belonging to the current job, the Quit All command will close all open GUI windows, as well as any invisible background processes belonging to the GUI (scmd and scmjobd).

BACKGROUND PROCESSES WITH NO INTERFACE

4.1 scmd: communication between GUI modules

The scmd process should always be running when any of the GUI modules is active. You should have one scmd process (per user, if you are working with multiple users). The scmd process has no window, it should normally be invisible to the user.

The purpose of the scmd process is to handle all communication between GUI modules. For example, if you use the SCM menu to start a GUI module, that command is actually handled by the scmd process. It will figure out if a new module needs to be started, or if an already open module with that particular job should be activated.

Starting jobs, and handling the local Sequential queue is also done by scmd.

Communication between the GUI modules and scmd is done using TCP/IP within the local machine. If such communications are blocked, for example by a firewall, this will make the GUI non-functional! So in case of trouble, make sure you are not blocking TCP/IP traffic within the local machine. The GUI will never accept any TCP/IP connection from outside, only from within the local machine.

4.2 scmjobd: monitor running jobs

The scmjobd processes are also faceless background processes that belong to the GUI. They are started and terminated as needed, normally one per queued or running job. Their task is to monitor the progress of jobs, updating the status if it changes, getting logfile results for remote jobs, and killing jobs (if you use the Kill command in AMSjobs). Even if all windows of the GUI are closed they may still continue to be present if you have queued or running jobs.

The scmjobd processes communicate with the other GUI components using TCP/IP, via the scmd process. To monitor remote jobs they will use ssh to connect to the remote machines.

VIEWING YOUR MOLECULE

5.1 Molecule style: Balls and Sticks, Sticks, or Wireframe

The molecule style can be selected via the View menu (**View → Molecule → ...**). You can select Balls and Sticks, Sticks, Wireframe or Hidden. This choice will affect the whole molecule. Note that if you use anything else then Balls and Sticks it will become much more difficult to select an atom (as there is no ball representing it).

You can also use different styles at the same time, for different regions. To do that, go to the Regions panel, make regions as you like, and set the molecule style from the pop-up menu per region in the Regions panel (the downward pointing triangle at the end of each region line).

5.2 Camera position: view direction

You can use the mouse to view your molecule from a different position.

In some situations you may want to view your molecule from some very specific angle. There are a couple of ways to do this:

- **View → View Directions → Along X-axes**
- **View → View Directions → Along Y-axes**
- **View → View Directions → Along Z-axes**

These commands need no further explanation.

- **View → Align Screen**

Using this command, you can align the screen parallel to some plane in your molecule. You need to select three atoms first which will define the plane to align to.

- **View → Camera → Save... / Load...**

First view your molecule from your favorite position (or use one of the just mentioned commands). Next, you can store the camera position by selecting one of the Save 1-5 commands. That will store the current view in the selected slot. To switch to one of the stored views, use one of the Load 1-5 commands.

The camera positions are saved for all GUI modules to use. Thus, if you start AMSinput (or any other GUI module with a View menu) the stored camera positions are available everywhere. This makes it possible to force the use of the same camera position in different GUI modules.

The Camera menu can be torn off, which is very convenient if you have several views that you wish to switch between frequently. To tear off the menu, select the dashed line in the Camerae menu.

5.3 3D

The GUI modules do have support for real 3D viewing. Use the **View → 3D** menu to select what kind of stereoscopic display you have. The simplest version is having none, but using colored glasses (like the Red-Cyan). The other options are for special hardware. Just try out what works for you. This is a rather new and not completely stable feature, so it may not work for you.

One known but at the moment this is written unsolved problem is that the 3D view does not work on Mac OS X using Lion (it does work with Snow Leopard).

5.4 Fly to selection

When you have a big system, it often make sense to zoom in to some particular region. To do this, use the **View → Fly To Selection** menu command.

Not only will this zoom in to the selected atoms, but the center of the selected atoms will also become the new center of rotation when you use the mouse to get a different view.

5.5 Reset View

You can reset the view using the **View → Reset View** menu command. This is especially useful after flying to the selection. Or when the view gets distorted for some reason.

MOUSE INTERACTION

6.1 Rotate, Translate and Zoom

In the modules that provide a 3D view of your molecule (like AMSinput, AMSview and AMSmovie) you can rotate, translate and zoom using the mouse.

Drag with the mouse: press a mouse button, and move it while holding it down. A one-button mouse button is the same as a Left mouse button. Which mouse button, and which modifier key you press at the same time, determines what will happen:

Rotate		Left mouse button
Rotate	in-	Ctrl with Left mouse button
plane		
Translate		Right mouse button
Zoom		Use the scroll wheel on your mouse, or Middle mouse button, or (not on windows) Alt with Left mouse button

The rotate, translate and zoom operations change how you look at the molecule. They do not change the coordinates.

In AMSinput operating with the mouse on the selection will move the selection only. In that case the geometry of your molecule (and thus the coordinates) will change. Zooming the selection will move it perpendicular to the screen, unless you are using the mouse-wheel. You operate on the selection by starting the drag operation with the mouse above a selected object.

In the View menu you can select either 'Mouse as trackball' or 'Mouse as joystick'. If 'Mouse as trackball' is selected, you need to drag with the mouse (move the mouse with a button pressed down). If 'Mouse as joystick' is selected you just need to press and keep the button pressed down. The direction of movement etc will depend on the position of the mouse with respect to the center of the 3D view area.

The mouse handling has changed in the 2013 release: the middle and right buttons have been switched. The right mouse button was used for zooming, now by default it is for translating (and zooming is much more convenient using the scroll wheel). Via the **SCM → Preferences** you can switch the mouse handling back to the previous style (use the "Right mouse button" option).

6.2 Selecting

In the modules that provide a 3D view of your molecule (like AMSinput, AMSview and AMSmovie) you can make selections using the mouse.

Click on an **object**: make a new selection **with** it

Click **in** space: clear selection

Shift-Click on **object**: add **or** remove it **from the** selection

Shift-Drag **in** space: add **all** objects within the rectangle to the selection

In some modules there are additional ways to select objects using menu commands. Furthermore, in AMSinput one can select atoms from the list in the coordinates window.

KEYBOARD SHORTCUTS

Many tools and menu commands have keyboard shortcuts associated. For menu command shortcuts, the shortcut is listed in the menu.

In many modules the shortcut to open or add a result file can be combined with the shift key. That allows selection of any file (otherwise a filter will be applied).

On most UNIX systems (including Linux) and Windows you need to use the control key together with some letter. On a Macintosh (running locally) you need to use the ⌘ key (command key) together with a letter.

If you are using a Mac, the GUI uses the X11 program to run. The X11 program itself may intercept the menu-shortcuts like ⌘-C, ⌘-H, and so on. You can change this using the Preferences command from the X11 menu (Enable key equivalents under X11 should NOT be checked).

The following table lists all keyboard shortcuts, per module. If no ⌘ or ctrl is specified Just press the indicated key without any modifier keys. On a mac, ⌘ should be used as shown, on other systems use the control key (ctrl).

7.1 AMSinput

Key	Function
arrow-up/down in search results	highlight the previous or next search result
backspace	Delete selection
delete	Delete selection
Return in search results	open the currently highlighted search result
Esc	Select-tool (end other tool), clear search, active Main page
1	Single bond
2	Double bond
3	Triple bond
4	Aromatic bond
⌘-0	Layout → Molecule Window
⌘-1	View Direction → Along X-axis
⌘-2	View Direction → Along Y-axis
⌘-3	View Direction → Along Z-axis
⌘-7	Layout → Use Default
⌘-8	Layout → Input Options
⌘-9	Layout → Molecule Window And Input Options
⌘-a	Select → Select All
⌘-b	Edit → Builder...
⌘-c	Edit → Copy
⌘-d	Select → Select Connected

continues on next page

Table 7.1 – continued from previous page

⌘-e	Atoms → Add Hydrogens
⌘-f	Search
⌘-g	Atoms → New Region From Selected Atoms
⌘-i	View → Properties (Estimated)
⌘-j	Transparency Periodic Cells → Increase Transparency Periodic Cells
⌘-k	Transparency Periodic Cells → Decrease Transparency Periodic Cells
⌘-l	Bonds → Add Bond
⌘-m	Select → Select Molecule
⌘-n	File → New
⌘-o	File → Open...
⌘-q	Quit
⌘-r	File → Run
⌘-s	File → Save
⌘-t	File → Add To Params
⌘-v	Edit → Paste
⌘-w	File → Close
⌘-x	Edit → Cut
⌘-z	Edit → Undo
⇧-⌘-b	Generate Super Cell...
⇧-⌘-e	Atoms → Remove Hydrogens
⇧-⌘-g	Atoms → Remove Selected Atoms From Regions
⇧-⌘-i	View → Geometry Info
⇧-⌘-i	Edit → New Molecule
⇧-⌘-r	Show RunScript panel
⇧-⌘-s	File → Save As...
⇧-⌘-z	Edit → Redo
Space	Structure Tool
c	Carbon
f	Fluorine
h	Hydrogen
n	Nitrogen
o	Oxygen
p	Phosphorus
s	Sulfur
x	Element (type to select element and then return)
⌘ -	Molecule Smaller
⌘ =	Molecule Bigger

7.2 AMSview

⌘-1	View Direction → Along X-axis
⌘-2	View Direction → Along Y-axis
⌘-3	View Direction → Along Z-axis
⌘-a	Select → Select All
⌘-b	Add → Isosurface: With Phase
⌘-d	Select → Select Connected
⌘-f	Fields → Calculated
⌘-i	Add → Isosurface: Colored
⌘-j	Transparency Periodic Cells → Increase Transparency Periodic Cells
⌘-k	Transparency Periodic Cells → Decrease Transparency Periodic Cells
⌘-l	Add → Add Bond
⌘-m	Select → Select Molecule
⌘-n	File → New
⌘-o	File → Open...
⌘-p	Add → Cut Plane: Colored
⌘-q	Quit
⌘-s	File → Save Picture...
⌘-w	File → Close
⇧-⌘-o	File → Open... (no filter)
⌘ -	Molecule Smaller
⌘ =	Molecule Bigger

7.3 AMSmovie

⌘-1	View Direction → Along X-axis
⌘-2	View Direction → Along Y-axis
⌘-3	View Direction → Along Z-axis
⌘-a	Select → Select All
⌘-j	Transparency Periodic Cells → Increase Transparency Periodic Cells
⌘-n	Transparency Periodic Cells → Decrease Transparency Periodic Cells
⌘-k	Scale Displacements → Smaller
⌘-l	Scale Displacements → Bigger
⌘-o	File → Open...
⌘-q	Quit
⌘-s	File → Save Geometry...
⌘-t	File → Add Coordinates → To Params
⌘-u	File → Update Geometry In Input
⌘-w	File → Close
⇧-⌘-o	File → Open Any... (no filter)
⌘ -	Molecule Smaller
⌘ =	Molecule Bigger
backspace	Delete active graph
delete	Delete active graph

7.4 AMSlevels

⌘-	Interactions → Decrease # Significant
⌘=	Interactions → Increase # Significant
⌘-a	View → Hide/Show All Fragments
⌘-i	File → Add...
⌘-o	File → Open...
⌘-p	File → Save Postscript...
⌘-q	Quit
⌘-w	File → Close
⇧-⌘-i	File → Add Any... (no filter)
⇧-⌘-o	File → Open Any... (no filter)

7.5 AMStail (Logfile)

⌘-c	Edit → Copy
⌘-o	File → Open...
⌘-q	Quit
⌘-w	File → Close

7.6 AMSoutput

⌘-c	Edit → Copy
⌘-o	File → Open...
⌘-p	File → Print...
⌘-q	Quit
⌘-w	File → Close
⌘-n	Show Next
⇧-⌘-n	Show Previous

7.7 AMSspectra

⌘-i	File → Add...
⌘-o	File → Open...
⌘-p	File → Save Postscript...
⌘-q	Quit
⌘-w	File → Close
⇧-⌘-i	File → Add Any... (no filter)
⇧-⌘-o	File → Open Any... (no filter)

7.8 AMSbandstructure

⌘-1	View Direction → Along X-axis
⌘-2	View Direction → Along Y-axis
⌘-3	View Direction → Along Z-axis
⌘-j	Transparency Periodic Cells → Increase Transparency Periodic Cells
⌘-k	Transparency Periodic Cells → Decrease Transparency Periodic Cells
⌘-o	File → Open...
⌘-p	File → Save Postscript...
⌘-q	Quit
⌘-w	File → Close
⇧-⌘-o	File → Open Any... (no filter)
⌘ -	Molecule Smaller
⌘ =	Molecule Bigger

7.9 AMSdos

⌘-1	View Direction → Along X-axis
⌘-2	View Direction → Along Y-axis
⌘-3	View Direction → Along Z-axis
⌘-a	Edit → Select All
⌘-d	Edit → Select Connected
⌘-j	Transparency Periodic Cells → Increase Transparency Periodic Cells
⌘-k	Transparency Periodic Cells → Decrease Transparency Periodic Cells
⌘-m	Edit → Select Molecule
⌘-o	File → Open...
⌘-q	Quit
⌘-w	File → Close
⇧-⌘-o	File → Open... (no filter)
⌘ -	Molecule Smaller
⌘ =	Molecule Bigger

7.10 AMSjobs

⌘-a	Tools → Add To SDF...
⌘-b	Edit → Clear
⌘-c	Edit → Copy
⌘-n	SCM → New Input
⌘-q	Quit
⌘-r	Job → Run
⌘-s	Queue → Status
⌘-t	Job → Add To Params
⌘-v	Edit → Paste
⌘-w	File → Close
⌘-x	Edit → Cut

SELECTING

In many modules there are menu commands to select atoms:

- **Select All:** select all atoms in your system.
- **Invert Selection:** all atoms in your system currently not selected will be the new selection.
- **Select Molecule:** extent the current selection so that all atoms connected (by bonds as visible) to it are included in the selection.
- **Select Connected:** extent the current selection so that all atoms directly connected (by bonds as visible) to it are included in the selection. Thus repeated use of this command will select all atoms in your molecule (same as Select Molecule).
- **Select Atom Close To Origin:** select one atom close to the origin. Often useful to select one atom in a crystal, and next selecting atoms in a sphere around it (with the next command).
- **Select Within Radius:** set the selection to all atoms in a sphere with specified radius around the center of the current selection.
- **Make Selection Cappable:** extent the current selection such that it is acceptable as a region for the Hybrid engine, were all bonds to other parts will be capped. A cappable region should not have bonds with an order > 1.25 to atoms not in the region, and atoms outside the region should have at most one bond to atoms inside the region. This command will extent the selection so that these requirements are handled.
- **Select Atoms Of Same Type / PDBName:** extent the selection with atoms with the same Type (element) or PDB-Name.

AMSPREFERENCES

The GUI preferences module of the GUI (started via the Preferences command in the SCM or Application menu) allows you to set many preferences.

9.1 General

In the preferences module you can change things as you would like them to be. To test them use the Preview button. This will instruct all currently open modules to use the selected preferences. If you then quit the preferences module your changes (to the preferences) will not be saved, and all open modules will revert to the previous preferences. Click the Save button to save the changes you made.

The preferences are save (by default) in a file called `.scm_guirc` in your home folder. Typically this is a hidden file, you need to use the terminal to see it. If you remove (or move) this file and start a GUI module a new clean `.scm_guirc` file will be made.

9.2 Restore defaults

In the file menu of the Preferences application you will find commands to restore the preferences to their default values. This is equivalent to deleting the `.scm_guirc` file.

You can also only reset the units as used by AMSinput to the default units.

9.3 Panels

On the top you will see tabs: Options, Colors, Module and Environment.

Click on any of those tab titles to get a menu that allows you to go to a panel belonging to that category.

9.3.1 General

The General panel allow you to set some general options, like mouse handling and font size.

9.3.2 Atoms and Bonds

This panel (click on the Options title of the Options tab to get to it) allows you to set some preferences for the visualization of atoms and bonds.

9.3.3 Colors

In the colors panel you can set details of the background color (for all modules), and the default colors for fields visualized by AMSview.

9.3.4 Atom Colors

This panel allows you to set the color range when coloring atoms by scalars. It also allows you to change the default color per element (click the + button to select for which element you want to change the color).

9.3.5 Bond Colors

Allows you to color the bonds by atom types or not.

9.3.6 AMSview

Set AMSview specific preferences (see also the Colors panel).

9.3.7 AMSjobs

Set AMSjobs specific preferences.

9.3.8 AMSlevels

Set AMSlevels specific preferences (which lines to show for contributions etc, you can change for what contribution/interaction range to show them, the width and the color).

9.3.9 Environment

Allows you to set environment variables to be used by GUI modules. They are saved in the GUI preferences, and have no effect on environment variables set by default in your shell.

AMSINPUT

This version of the GUI has unified the input modules for ADF, BAND, ReaxFF and other packages. The new input module, typically called AMSinput, can now prepare the input for all of the old methods, as well as for new methods like DFTB, Mopac, UFF and so on.

After starting AMSinput, you will notice a bright colored menu ‘ADF’ on the left of the panel bar. Depending on your license, it might read ADF, or BAND, or one of the other methods. If you click there, you will get a pull-down menu that allows you to switch the method. Thus effectively you can turn AMSinput into BANDinput, or ReaxFFinput, and so on.

In the SCM menu you will a New Input command. This command will start AMSinput. Next you can switch to BAND mode, etc, as just described.

The unified interface makes it possible to do things like reading a crystal structure file (which will cause AMSinput to switch one of the methods supporting periodicity automatically). Next you can build super cells, cut out clusters, and switch back to ADF without periodicity. No need for extra licenses, and no need to copy/paste between GUI modules.

The Main panel for some methods (like DFTB and UFF) contain a ‘Run’ button. If you press that, the current method will be started interactively, with the geometry of your system updating on the fly. The main use for this is as a pre-optimizer, no files will be saved and only the geometry will be updated.

10.1 Input: options

10.1.1 Normal Options

On the right side of the AMSinput window you have the panels with input options.

Most of the options presented in the different panels are generated automatically from configuration files that define the input for ADF, BAND etc. This ensures that all options are up-to-date, and that the descriptions in the help balloons (when your mouse pointer is over an option) are also up-to-date (and identical to the information in the user manuals).

The options are logically grouped together, the most important are on the Main panel.

Options regarding the physical Model are in the Model section, options that determine how and what properties are calculated are in the Properties section, and many other options that are details relating to the calculation are in the Details section.

If you cannot locate the option you are looking for, use the search field at the top right of the AMSinput window.

10.1.2 Expert Options

Some options are either hardly ever used and have not been put explicitly on some panel. And similar it might happen that the compute codes have some new options that have not yet been explicitly added to AMSinput. All such options are added to the Expert Options panel in the Details section (again automatically generated).

Most users will not need to use these, but all options are available in this way.

The Expert Options panel is available for all SCM software, except for ADF (this will be fixed in a future release).

Again, if you cannot locate the option you are looking for, use the search field at the top right of the AMSinput window.

10.1.3 Units

Often options have some unit associated with them. The units are clearly presented in the panels.

A hidden feature is that in most cases in AMSinput you can change the units. Just click on the unit with your mouse, and a menu will pop up that you can use to select the units that you like.

The units that are selected without moving the mouse are the default units.

When you make a change, that change is saved when you quit AMSinput (in the \$HOME/.scm_guirc file). The next time you start AMSinput, that particular field will use the units you selected. To revert everything to the default units, use the Preferences module, and select the 'Reset AMSinput units' command from the File menu.

10.2 Input: buttons

10.2.1 Searching

At the top right in the panel bar you will find a Search tool. You can activate it by clicking on it, or by using the ctrl/cmd-F shortcut.

After activating it, a small field will pop-up that allows you to enter your search text.

When you enter some text in the search field, the search will start immediately and the results will be presented below the search text. The results are divided in several sections:

Panels: a list of all panels that match the search text. All texts on a panel will be searched: titles, menus, help text and associated keys. When you click on one of the panel search results that panel will be activated, and the items matching your search query will be marked. This is a very convenient way to find a particular input option. The search is restricted to panels belonging to the currently active method (thus, while in ADF input mode you will not find BAND input panels).

Documentation: a list of matching documentation pages. If you click on it, that page will be shown in your browser. Note that this uses the local documentation, it does not contact the SCM web site. Again, the search is restricted to documentation that belongs to the currently active method.

Molecules: a list of matching molecules. The match may for example be in the name or in the molecule formula. If you click on a match, that molecule will be imported. The list of molecules is a mix of some ADF optimized molecules, and of results taken from the NCI-Open database (see <http://cactus.nci.nih.gov>). Again, all information is contained in the ADF distribution, the search will not contact external sources. In total about 35000 molecules are included.

Crystals: a list of matching crystals. The match may be in the name or formula. Most of the crystals are simple common crystals are found in many textbooks. On the MOPAC web site there is a list of crystals that have been optimized using MOPAC. Those are also included. Finally, the zeolite frameworks from the [IZA-SC database](http://www.iza-structure.org/databases/) (<http://www.iza-structure.org/databases/>) are available.

When starting, at most five matches will be shown in each section. At the right side of the section headers (Panels, Documentation, Molecules) you will spot a triangle pointing to the right. If you click on it, you will open that particular section and many more results will be shown, if available.

Pressing the “?” (to the right of the search text field) will bring up a detailed description of the search facility. It will explain how to do more complex searches.

If you move your mouse above one of the search results, without clicking on it, a balloon will pop up showing more details about that particular match. For the documentation matches it will tell you from what manual the result is, and from which chapter. For molecules it will give some more information about that molecule, like names, sometimes a CAS number, and what the source of the data is (for example, optimized using ADF, or from the NCI-Open database).

After typing a search text, you can also use the up and down arrow keys to select a particular search result, and then press the Return key to open that result. When starting with ctrl/cmd-F this means you can perform the search and accept result without using the mouse.

10.2.2 The “>” details buttons

In many panels in AMSinput you will find buttons with an arrow (or a greater-than sign). These buttons will activate a panel showing more details for some particular feature.

For example, the “>” button to the right of the Task option in AMSinput will activate a panel that allows you to set details for the current task. Thus details for a geometry optimization, or details for a transition state search.

It depends on the context and sometimes on the active options to which panel the buttons link. In all situations it should be intuitive, just try it out.

10.2.3 Using the “?” info buttons

Some panels in AMSinput have an “?” button (just as the search pop-up). When you press the “?” button, a new page will open with much more background information on some feature.

For example, the “?” button in the search pop-up will show a description of the search syntax. And the “?” button next to the basis set (in the Basis detail panel) will show more information about the basis sets.

10.3 Searching

In AMSinput there is a search icon in the top-right corner. Click on it to bring up a search box. Searching will be performed automatically when you enter text in the search box.

You will search multiple things at the same time:

- matching panels with input options in AMSinput (from the panel bar)
- documentation
- molecules

The search will be restricted to panels and documentation that belong to the currently selected program (ADF, BAND, DFTB, ...). In these panels, it will find any panel that contains your search string, including the matching program key.

Prefix: key

If you start your search with “key “ (without quotes), you will restrict your search to fields related to matching input keys only. This will allow you to locate an input option very quickly.

Prefix: all

If you start your search with “all “ (without quotes), the search will be global for both the panel search and documentation search. Thus it will not be restricted to the currently selected program.

The molecule search will search a (local) database with some common molecular structures. Currently it consists of a roughly 1800 molecules optimized with ADF for use with COSMO-RS, and of the molecules found in the NCI database (NCI-Open_09_03.sdf, see <http://cactus.nci.nih.gov>). The molecules in the NCI database have been filtered to include only molecules with names and xyz coordinates, and most CAS numbers have been removed for legal reasons. Also periodic structures are available: zeolites (from the [Database of Zeolite Structures](http://www.iza-structure.org/Databases/) (<http://www.iza-structure.org/Databases/>)), many simple crystal structures, and a set of complex crystals included with the MOPAC distribution.

As the structures come from an external source we can not guarantee that the found structures will be correct. So please check them!

In total roughly 35000 molecules are available.

To clear the search results at any time, also when the search box is no longer visible, use the Escape key.

Only the locally included documentation and molecule structures will be searched. No internet search will be performed.

10.3.1 Panel search details

All text belonging to the panels will be searched: panel titles, labels for input options, headers, text in pull-down menus, and text in help balloons.

You can use the * and ? wildcards: the * matches anything, the ? matches any single character. The search string will be broken into words, and only results containing all words will be found. Use double quotes (") around text to search for exactly that phrase. The words to search for are implicitly surrounded by * wildcards, so text matching will happen even in the middle of a word.

The names of the found panels will be listed. When you click on one of them, it will be activated. Options not matching the search string will be dimmed. If an option is not dimmed, it matches either directly, by its help text, or by the page title.

Use the arrows next to the search loupe to quickly loop through the found panels.

10.3.2 Key search details

To search for input fields that have to do with a particular input key, you can use the key search. To do this, start your search string with “key “ (without the quotes). This will restrict the panel search to find only items that set or change the matching key.

You can use the * and ? wildcards: the * matches anything, the ? matches any single character. The search string will be broken into words, and only results containing all words will be found. Use double quotes (") around text to search for exactly that phrase. The words to search for are implicitly surrounded by * wildcards, so matching will happen even in the middle of a word.

For example, searching for

key integration

will show you what input options in the GUI will set the INTEGRATION key.

The names of the found panels will be listed. When you click on one of them, it will be activated. Options not matching the search string will be dimmed. If an option is not dimmed, it matches either directly, by its help text, or by the page title.

Use the arrows next to the search loupe to quickly loop through the found panels.

10.3.3 Documentation search details

The included documentation (in html format) will be searched using the Swish-e search engine. You will be searching for words. If you specify more you will get matches that contain both words, in any order.

The (shortened) titles of the found documentation pages will be listed. If you point at an item with your mouse, it will show the full title. Click on it to open that item with your default browser. Note that you will be viewing local documentation, not the SCM web site.

When staring, only the first 5 documentation matches will be shown. To view all results, click on the arrow on the right side of the 'Documentation' header in the search results. Click on the same arrow again to show only the first 5 results again.

You can use the * wildcard to match anything, but only at the end of an word. You can use the ? wildcard to match a single character, but NOT at the beginning of an word. You can also perform complex searches using logical operators (and, or, not) and parenthesis. If no operator is specified between words, the and operator is used (both words need to be found) Finally, you can search for an exact phrase using quotes.

For example: integration and not accuracy

integration or accuracy integration key "integration key"

The search will be restricted to the parts of the documentation relevant to the module you are using. Thus, when searching from AMSinput with ADF active, the BAND documentation will not be searched.

If you prefix your search with "all " (without the quotes), all documentation will be searched.

For more information on Swish-e: <http://swish-e.org/>

10.3.4 Molecule search details

The included database of molecules will be searched. You can search for the IUPAC name, the molecule formula, the CAS number, and sometimes alternative names. Note that the CAS number search will often fail as only a few CAS numbers are included.

You can use the * and ? wildcards: the * matches anything, the ? matches any single character. The search string will be broken into words, and only results containing all words will be found. Use double quotes (") around text to search for exactly that phrase. The words to search for are implicitly surrounded by * wildcards, so text matching will happen even in the middle of a word.

When you start your search string with "=" (an equal sign, no quotes), no wild cards will be added automatically.

When you move the mouse over one of the found molecules, you will get full details including the origin of the entry (optimized by ADF, or from the NCI database).

When you click on the desired search result, that particular molecule will be imported.

10.4 Crystals and Slabs

10.4.1 Periodicity

Many GUI modules support periodicity. Some menu commands (like the **Edit → Crystal** command in AMSinput) will only be available when you have a periodic system.

To set up periodicity you first need to switch to a method that supports periodicity. For example BAND, DFTB, Mopac, or UFF. Some of these, like UFF, are currently available without extra license.

On the Main panel of a method that supports periodicity, you will find a 'Periodicity' pull-down menu. Use it to use periodicity in 1 (Chain), 2 (Slab) or 3 (Bulk) dimensions. Once periodicity is enabled, menus like the Crystal menu will also be enabled.

The tool bar below the molecule editor also changes when you have a periodic system. The structures tool gets replaced by the crystal tool (represented by a snowflake), followed by the Slice tool (which invokes the Generate Slab command). At the right side of the tool bar you find buttons to orient the camera, toggle perspective and toggle the display of periodic images.

To set up the lattice vectors, press the "... details button next to periodicity, or go to the Lattice panel (in the Model section).

When editing the lattice vectors: changing the 1,1 element also adjusts the 2,2 and 3,3 element. Changing the 2,2 element also adjusts the 3,3 element (but leaves the 1,1 as it is).

10.4.2 Building crystals

After setting the periodicity and lattice vectors, you can build your own crystal by adding atoms at the proper position. For many crystals this is a lot of work.

You can also build crystals using the **Edit → Crystal** command. The first list of options (Cubic, Hexagonal and so on) have sub-menus for some common crystals. If your crystal is not in there, you can use the "From Space Group..." command in the same menu. This brings up a dialog that will help you build any crystal.

The crystal tools are not only available from the Edit menu, you can also press the 'Snowflake' button in the molecule editor tool bar (next to the element button, if periodicity is enabled). It has the same options as the **Edit → Crystal** menu.

10.4.3 Build slabs

Building a slab is done most conveniently using the Generate Slab tool: first build a crystal, next use the "Generate Slab..." command from the Crystal menu to make a slab. Next you will need to enter the Miller indices of the plane that will define your slab.

The Generate Slab tool is available from the **Edit → Crystal** menu, or in the periodic toolbar (the slice icon next to the snowflake).

10.4.4 Crystal menu

Generate Super Cell...

In the dialog that appears when using this command you determine how the super cell should be created.

When editing the setup: changing the 1,1 element also adjusts the 2,2 and 3,3 element. Changing the 2,2 element also adjusts the 3,3 element (but leaves the 1,1 as it is).

Map Atoms To (-0.5..0.5)

Edit → Crystal → Map Atoms To (-0.5..0.5)

Using the translational symmetry, the atoms are mapped to the cell centered around the origin. Thus with fractional coordinates between -0.5 and 0.5 for each lattice direction.

Map Atoms To (0..1)

Edit → Crystal → Map Atoms To (0..1)

Using the translational symmetry, the atoms are mapped to the cell in the positive quadrant. Thus with fractional coordinates between 0 and 1 for each lattice direction.

Set (0.5, 0.5, 0.5)

Edit → Crystal → Set (0.5, 0.5, 0.5)

The selected atoms define some point. This point will be translated to the point with fractional coordinates 0.5, 0.5, 0.5. All atoms will be adjusted according to this translation.

Related, to shift the point defined by the selected atoms to the origin use the Set Origin command.

10.5 Builder (packmol)

To build systems consisting of many different molecules, randomly put together, the GUI includes the Builder. This is a graphical interface to the packmol program, using only a few possibilities of it. The most common use will be to set up a big system for Molecular Dynamics with Force Field, ReaxFF, ML Potential, or DFTB.

See also:



- [Tutorial](#) on the most important Builder features
- [PLAMS Python interface](#) to Packmol (maintained by SCM)
- Packmol [website](https://m3g.github.io/packmol/) (<https://m3g.github.io/packmol/>)
- Packmol citation: L. Martínez, R. Andrade, E. G. Birgin, J. M. Martínez. *Packmol: A package for building initial configurations for molecular dynamics simulations*. *Journal of Computational Chemistry*, 30(13):2157-2164, 2009 (<https://doi.org/10.1002/jcc.21224>)

10.5.1 The five starting modes of the Builder

The Builder can be started in five modes, depending on the current system when the Builder is launched:

1. *Packing in an empty box*: The Builder is started for an **empty system**. The Builder will then pack the molecules **and** modify the lattice vectors of the system.
2. *Packing in an empty box, replacing current system*: The Builder is started for a **non-empty non-periodic system**. This is identical to the previous mode. The current molecule is discarded. To pack the current molecule, first export it to a .in or .xyz file, and import the file from the Builder.
3. *Packing inside the crystal*: The Builder is started for a **non-empty 3D-periodic system**. The Builder will then pack molecules **around** the existing system, into a space defined by the lattice vectors. The lattice vectors will **not** change.

4. *Packing around the slab*: The Builder is started for a **non-empty 2D-periodic system**. This is similar to the previous mode, but you must specify the box bound along *z*.
5. *Packing around the chain*: The Builder is started for a **non-empty 1D-periodic system**. This is similar to the previous mode, but you must specify the box bounds along *y* and *z*.

Tip: Activate the engine that you want to use for your calculation *before* starting the Builder, for example by switching from  → .

Why? The Builder is most commonly used to build a 3D-periodic system. However, the ADF engine (default in AMSinput) cannot handle 3D-periodic systems. If the ADF engine is active when you start the builder, then the lattice vectors will be removed when you close the builder, since ADF cannot handle them.

Note on non-orthorhombic systems


The Builder for periodic systems is most reliable if the lattice vectors are oriented along *x*, *y*, and *z* with 90 degree angles. For non-orthorhombic systems, the Builder removes molecules that fall outside the unit cell. This usually means that you pack fewer molecules than you specify.


For non-orthorhombic systems, always **carefully** check the resulting structure to see if it matches your expectations.

10.5.2 The Builder has flexible input

The Builder has flexible input. This means that you specify only some of the input options, and appropriate or approximate values for the remaining input options will be automatically determined.

- **Density**: The mass density of the molecules to be packed.
- **Approximate number of atoms**: The approximate number of atoms to pack. Cannot be combined with *Number of molecules* (see below).
- **Box Dimensions (X, Y, Z)**: Pack the molecules into a box defined by these dimensions
- **Tolerance**: Approximate interatomic distance between molecules. To allow for more tight packing, decrease this number.
- **Use number of molecules**: Explicitly specify the number of molecules to pack in the *N mols* column.
- **Use mole fractions**: Specify relative mole fractions of different components in the *Mole fraction* column. The actual number of molecules to pack is determined by other input options.

You can choose how many types of molecules to pack. Click the  to add more molecules. For each type of molecule, you may specify:

- **SMILES or file**: (i) SMILES code for a molecule: enter CCO for ethanol, *or* (ii) a file from the SCM library: enter `ethanol` and click on the match in the drop-down, *or* any file on disk: click the  icon
- **N mols**: The number of molecules to pack for this type
- **Mole fraction**: Relative mole fraction for this molecule
- **Region**: Assign a region name for this molecule. By default, all packed molecules are placed into a region called `Builder`.

Finally, there is an option to perform a *Dry run*. This will generate a Python script that you can use as a starting point for running the Builder from Python (for example, to automate the generation of different box sizes, mole fractions, ...).

10.6 Building molecules

Building molecules with AMSinput is very simple, just draw them with the appropriate tools. Please check the tutorials for extensive description of this.

Here we will just note some details that may not be obvious.

10.6.1 Bonding mode

After creating an atom, it will be selected (the 'bond atom'). When you still have an atom tool and one atom is selected, you are in bonding mode. This is indicated by a line from the bond atom to the mouse cursor. If you next click to create a new atom, it will automatically be bonded to the bond atom.

To stop bonding mode, click once on the bond atom. This will switch to the select tool, and as a result you are no longer in bonding mode.

10.6.2 Atom Details: Connectors (valency) and Lone Pairs, Atom mass

Each atom you make has the possibility to connect to other atoms. The number of connections depends on the atom type. Some of the 'connectors' may be used by lone pairs and not be available for bonding to other atoms.

For example, carbon has 4 connectors and no lone pairs, oxygen has 4 connectors and 2 lone pairs.

When you use the Add Hydrogen menu command, all unsaturated connectors will be saturated with hydrogen atoms. The number of connectors will determine the geometry (2 linear, 3 flat, 4 tetrahedral). So if you plan to use this command it is essential that the number of connectors and the number of lone pairs is correct.

You can view and change these via the **Atoms → Details (Color, Radius, Mass...)** menu command. It will activate the Atom Details panel, in which you can change the number of connectors and lone pairs for each atom. Note that only selected atoms will be visible in the Atom Details panel.

You can also see and change other atom properties like the type, screen radius, nuclear charge, mass, color, Amber and Tripos atom types and more.

Right click on one of the fields to propagate that value to all selected atoms (for example to change the color of all selected atoms).

10.6.3 Pre-optimization

After drawing a molecule you will typically run a pre-optimization.

On the right end of the molecule editor tools you will see a cog wheel. If you click it your system will be optimized using the UFF method.

If you right-click the cog wheel you will get a pop-up menu that allows you to specify which method to use as a pre-optimizer: UFF, Mopac, DFTB or SimpleMM. UFF, Mopac and DFTB are identical to these methods within AMSinput, SimpleMM is a very primitive force field like method that used to be the only available pre-optimizer. It is mainly included for compatibility.

Alternatively you can just switch AMSinput into UFF, Mopac or DFTB mode (by using the orange method menu in the panel bar). For each of these you will find a 'Run' button on the Main panel. If you press it, it will run the current set up interactively, updating the coordinates and possibly the bonds on screen automatically. This is very similar to clicking the cog wheel, but it allows you to set details of the calculation. For example, for DFTB you can select what parameter set to use.

10.6.4 Geometry adjustments

Sliders

Select two, three, four or five atoms.

Below the molecule, a horizontal slider will appear, just above the toolbar.

You can drag the thumb inside the slider to change the distance, angle, dihedral, or plane angles.

The order in which you select the atoms determines what atoms will move. The selection order is shown in the molecule editing window. The last atoms selected will move (together with things connected to those atoms). However, if you press the control key and then move the slider, the smallest group will move. You can force this behavior (so you do not need to press the control key) by setting the `SCM_GEOMODSBYSIZE` environment variable (via the Preferences or via the command line). Note in the 2017 version the default was the reverse (smallest group moving, by order with control key).

You can use regions to group atoms. Atom in a region will move together when using a slider to change distance or angle.

Edit box

Select two, three, four or five atoms.

Above the slider on the right side you will get an text entry box next to the atom labels. For example, after selecting two atoms it might look like H(8)-C91) (pm): 149. The 149 in this example is editable. Instead of using the slider, you can type some specific number in this box instead.

Centroids

A centroid in AMSinput is defined as a dummy which is (and will stay) in the geometrical center of the atoms to which it is bonded. They are useful to control the geometry for molecules like ferrocene, as you can easily set the distance between the central atom and the center of a ring.

To make a centroid, select the atoms that should define the center, and use the **Edit → Add Centroid** menu command.

As an example, to change the distance between a ring and the Fe atom in Ferrocene:

1. Use the structure tool to load Ferrocene.
2. Select the carbons in one ring.
3. Add a centroid.
4. Delete the bonds between Fe and the carbons belonging to the centroid (otherwise you will not be able to change the distance)
5. Select the centroid and the Fe
6. Use the slider to set whatever distance you want (or set it in the input field).

Thus the centroid functions as a fixed manipulation point.

Another example, to change the angle ring - Fe - ring in Ferrocene, adjusting the planes of the rings as well:

7. Make a similar centroid in the second ring (also with Fe - C bonds removed).
8. select centroid - Fe - centroid.
9. Change the angle using the slider.

Another way to manipulate your geometry is selecting the centroid (as the only selected item). Dragging the centroid will drag everything to which it is bonded.

Move an atom (possibly perpendicular to the screen)

First select the atom that you want to move.

Next, translate (middle mouse button, or alt left mouse button), but start with the mouse on the atom that you want to move.

If you wish to move the atom perpendicular to the screen: use the right mouse button (or command left mouse button) and move the mouse up or down). This is equivalent to zooming.

Rotate or translate the selection

First make your selection.

Next rotate, translate or zoom as usual, but start with the mouse in the selection. So if you click and drag the selection, ONLY the selection will be rotated or translated. If you click and drag anywhere else the whole molecule will be rotated or translated (actually, only your viewpoint). Rotating a selection that has bonds across a periodic boundary will fall back to rotating the viewpoint.

If you 'zoom' the selection you are really moving the selected objects perpendicular to the screen (in or out the screen).

Coordinates

In some situations you want to set the coordinates of the atoms exactly to some value. You can do this via the Coordinates panel (in the Model part of the panel bar).

If you first select the atoms of interest, they will be highlighted in the Coordinates panel. The coordinates in that panel can be edited, and the molecule on the left side will immediately reflect the changes.

Z-Matrix

In some situations you want to set the coordinates of the atoms via a Z-Matrix. You can do this via the Z-matrix panel (in the Details part of the panel bar).

If you first select the atoms of interest, they will be highlighted in the Z-Matrix panel. The values for the distances and angles in that panel can be edited, and the molecule on the left side will immediately reflect the changes.

The connectivity can not be edited, but you can effectively do this by reordering the atoms. That way you normally can create the Z-Matrix coordinate that you are after.

Set Center Of Mass

The **Edit → Set Center Of Mass** command translates all atoms so that the mass-weighted center of the selection will be the new origin. If nothing is selected, the center of mass of the whole molecule will be the new origin.

The atoms are really translated, thus the coordinates are changed. It is not a change of the camera.

When symmetry is used, the origin will also be the origin of symmetry.

Set Origin

The **Edit → Set Origin** command translates all atoms so that the center of the selection will be the new origin. If nothing is selected, the center of all atoms will be the new origin.

The atoms are really translated, thus the coordinates are changed. It is not a change of the camera.

When symmetry is used, the origin will also be the origin of symmetry.

Symmetry

To symmetrize your molecule, click on the **star tool** in the tool bar below the molecule editor. This is equivalent of the **Edit → Symmetry → Symmetrize Using Symmol** menu command. In most cases this just works, and is the preferred method of symmetrizing your molecule.

The **Edit → Symmetry** menu gives access to a less automatic but more power-full symmetry code. Using it is somewhat complicated, and sometimes it does not work as expected.

To use it, follow this procedure:

- Select the symmetry group
- Set the (symmetry) origin of your molecule via the SetOrigin menu command
- Define one or two operators for the selected group. In the symmetry menu you can see what operators you need to define.
- Use one of the symmetry commands:

Rotate 90

The **Edit → Rotate 90** commands (select an axes via the submenu) will rotate your system around some axes.

Not only the coordinates of the atoms, but also the lattice vectors (if any) will be rotated.

Thus, for example, if you get some .cif file with a nanotube structure and want to treat it with BAND as a one-dimensional periodic structure (a chain), you must ensure it is oriented along the x axes. If the structure in the .cif file is oriented along another main axes, these commands make it very easy to fix that.

Align

The **Edit → Align** command contains commands to align your molecule to the indicated axes or plane.

If you are aligning to an axes, you need to select which atoms to align. If you select one atom, the molecule is rotated around the origin such that the selected atom is on the specified axes. If you select two atoms, first the molecule is translated such that one of the selected atoms is on the origin. Next, the molecule will be rotated such that the other selected atom is on the specified axes.

If you are aligning to a plane, you need to select a plane to align. If you select two atoms, these will define the normal of the plane to be aligned. If you select three atoms, these define the plane to be aligned.

This Align command does change the coordinates of your atoms, unlike the Align Screen command in the View menu.

Not only the coordinates of the atoms, but also the lattice vectors (if any) will be rotated.

Typically the Align command will change your origin. You can always set the origin as you like after-wards.

Mirror

The **Edit → Mirror** command contains commands to mirror the selected atoms (or all atoms if no atoms selected) in a plane parallel to the indicated plane, through the center of the selected atom(s), or through the center of the molecule if no atoms selected.

The Mirror command does change the coordinates of your atoms. Also the lattice vectors (if any) will be mirrored.

Update coordinates via AMSmovie

If you have performed a calculation that saves multiple geometries, for example a geometry optimization or a linear transit run, you can use AMSmovie to see how the geometry changes. You can go to a particular geometry, and then use the **File → Update Geometry in Input** menu command in the Movie window to adjust the geometry in the matching AMSinput window.

10.6.5 Bonds

Bonds are normally made while drawing the molecule. There are several other options to handle bonds:


- Make a bond via an atom tool: Take one of the atom tools. Next click once on the first atom you want to connect. You will enter the bonding mode (the line to the mouse position from the atom you just clicked on will be your visual cue for the bonding mode). Next click on the atom you want to make the bond to. The bond will be created, and you will revert to the normal select mode.
- Make a bond using the Add Bond command: Select the two atoms that you wish to be bonded together, and then use the **Bonds → Add Bond** command.
- Via Guess Bonds Use the **Bonds → Guess Bonds** menu command. This command uses an algorithm that looks at bond distances to guess the bonds and bond orders. The bond orders are adjusted based on the known number of connectors and lone pairs.

10.7 Tuning Geometry: translation, rotation, position with respect to plane

The **Edit → Tune Geometry...** menu command gives you a panel with some options to fine-tune the geometry of your system.

You can use it for example to:

- precisely translate or rotate part of your system
- scale your system
- orient part of your system, for example an adsorbate
- move adsorbates around on surfaces

Typically you will have to set up some details first, like which atoms to move, or in which direction. The  button will use the currently selected atoms to set the details in front of the button.

10.7.1 Atoms to move

To tune the geometry of your system some atoms will have to be moved. This may be a single atom or many atoms. If you do not use this option, the selected atoms will be adjusted.

Often it is convenient to fix the set of atoms that will be adjusted.

To do that, use the  button: it will set the atoms to be moved to the currently selected atoms.

That allows you to change the selected atoms (for example to set other options) without having to re-select the atoms to move.

Click the button that lists the atoms (or shows the text ‘Selected Atoms’) to show which atoms have been set (they will be selected).

10.7.2 Translate atoms, direction

This option will translate atoms in a specified direction.

1. First specify a direction, this is a vector in x, y, z coordinates.

You can use the  button to set this vector using the currently selected atoms:

- from the first selected atom towards the second selected atom,
- perpendicular to the plane of three selected atoms.

2. Next you need to set the step size, the default is 0.1 Angstrom.

This step is in the direction specified (the direction vector will be normalized).

3. Click the left  or right  arrow to translate the atoms.


The atoms will be translated in the specified direction, with the specified step size.

4. After the first click you may use the space bar to repeat the action.

10.7.3 Translate atoms, target

This option will translate atoms towards a specified point (target).

1. First specify a target, this is a point in x, y, z coordinates.

You can use the  button to set the target to the geometrical center of the selected atoms.

2. If you check the “As a group” option, all atoms move in the same direction (from the center of the atoms towards the specified point). Otherwise, each atom moves towards the specified point.

3. Next you need to set the step as a fraction of the distance to the target, the default is 0.01.

4. Click the left  or right  arrow to translate the atoms.

The atoms will be translated in the direction of the target, with the specified step.

5. After the first click you may use the space bar to repeat the action.

Note: as the step is a fraction of the distance of the atoms to the target this option can also be used to scale all coordinates (use a target at 0 0 0, and be sure not to move the atoms as a group).


10.7.4 Rotate atoms



This option will rotate atoms around some axes, around some point with a specified angle.

1. First specify an angle in degrees.
2. Next specify the direction of the axes around which will be rotated.

You can use the  button to set this vector using the currently selected atoms:

- from the first selected atom towards the second selected atom,
 - perpendicular to the plane of three selected atoms.
3. Specify the center around which will be rotated.


You can use the  button to set the target to the geometrical center of the selected atoms. Obviously the rotation axes will run through this center.

- If no atoms are selected, that button will clear the fields.
 - If no center is specified, the center of the rotation will be the geometrical center of the atoms that will be rotated.
4. Click the left  or right  arrow to rotate the atoms.
 5. After the first click you may use the space bar to repeat the action.

10.7.5 Position atoms with respect to a plane

This option will position atoms with respect to a plane.

1. First define the plane.

This is done by selecting three atoms, and using the  button.

The plane through the three selected atoms will be used. The default is the XY plane through the origin.

Click the button that shows the atoms defining the plane to show these atoms by selecting them.

2. Next, optionally, define 'Above atoms', which is a set of atoms above which your atoms should be located. The geometrical center of this set of atoms will be used as reference point.

Click the button that shows the 'Above atoms' to show these atoms by selecting them.

3. You need to specify the distance to the plane (default 2.5 Angstrom), and how this distance is measured:
 - to the geometrical center of the atoms that will be adjusted,
 - to the closest atom of the atoms that will be adjusted,
 - to the first atom selected that will be adjusted.
4. Click **Move** to move (translate) the atoms to be adjusted with respect to the plane and the 'Above atoms'.

The atoms will be translated such that:

- the distance of the reference position to the plane is the specified distance,
 - if 'Above atoms' have been set: the reference position will be above the 'Above atoms' when looking perpendicular to the plane,
 - if multiple atoms have the same distance to the plane, the geometrical average of these atoms will be used.
5. Click **Orient Perpendicular** to position the atoms perpendicular to the surface:

The atoms will be adjusted such that:

- the line through the first and second atom of the atoms will be perpendicular to the plane,
- the first atom will be kept in place,
- if already perpendicular, clicking this button will rotate the selection 180 degrees.

6. Click **Orient Parallel** to position the atoms parallel to the surface:

The atoms will be adjusted such that:

- the first atom of the set of atoms to move will be kept in place,
- the line through the first and second atom of the atoms will be parallel to the plane,
- it will also be parallel to the line through two 'Above atoms', or to the line through the first two atoms defining the plane, or the X-axes,
- if already parallel, clicking this button will rotate the selection 180 degrees.

10.8 Importing and exporting your molecule

10.8.1 Importing your molecule

Importing from file: .ams, .t21, .xyz, .mol, .mol2, .pdb, .runkf, .adf, .band, .cif, .dmol, .rkf, .rxi, .rxkf ...

You can also import your molecule from a file using the **File → Import Coordinates...** command.

Many formats are handled explicitly. They are typically recognized by the file extension:

.ams, .band, .rxi

These files have been saved by AMSinput, BANDinput or ReaxFFinput. Every detail belonging to atoms or bonds will be imported: not only the coordinates and bonds, but also things like atom properties (for example the color, the force field atom type, number of connectors, bond orders, and so on).

.t21, .runkf, .rxkf, .rkf

These are all results files produced by the ADF package. The coordinates and bond information will be read, if present. The files are binary, and will be recognized even if having a different extension.

.pdb, .mol2

Importing from a .pdb or .mol2 file will get the coordinates and the protein information (residues, PDB atom names and so on). For PDB files, only the ATOM information is read, thus no bond information. For MOL2 files, also the bond information is read.

.mol

Importing from a .mol file will get the coordinates and bond information.

.cif

Importing from a .cif file (Crystallographic Information File) will import the proper crystal structure. No bonds will be read.

.dmol

Files from the DMol program can also be read, and atom positions and periodic information is properly handled. No bond information will be read.

.xyz, any file type not recognized

Importing from an unknown text file, or a file with .xyz extension is rather flexible: AMSinput needs three real numbers next to each other. These will be interpreted as x, y and z coordinate. One additional integer or the abbreviation of an element is also needed to identify the kind of atom.

To be recognized as real, the real number must contain a '.' (dot), and at least one digit before or after the dot. Real numbers with exponents (E or D) are **not** recognized.

If an integer is used to specify the element (the nuclear charge), it may **not** contain a ‘.’ (dot).

No bond information is read.

If no bond information has been read, AMSinput tries to guess the bonds between the imported atoms. Guessed bonds might be completely wrong, or have the wrong bond order.

Z-matrix import (internal coordinates) is not available.

After using the ‘Import Coordinates...’ command the newly imported atoms are selected. This makes it easy to reposition them with respect to other atoms that may already be present, remove the automatically guessed bonds, or use other operations on the newly imported atoms and bonds.

If you are importing many atoms, the molecule visualization will switch to wire-frame for the newly imported atoms. By default this happens when importing 50 atoms or more. You can change this limit via the **SCM → Preferences**.

10.8.2 Copy/Paste between GUI modules, XYZ coordinates, SMILES or InChI strings

When Copying or Pasting with AMSinput, you need to make sure that the molecule editor has focus. Otherwise you will be copying or pasting one of the text input fields. To be sure, just click once in the drawing area before pasting.

You can copy an XYZ-formatted geometry (for example from an ADF output file), and use the **Edit → Paste** command to import coordinates. Pasted text will be handled as when it had been import as .xyz or .cif file via the Import Coordinates command.

Copy/Paste also works between the GUI modules, for example if you have several AMSinput windows. This will copy all information about your molecule: xyz, bonds, but also things like color of atoms or number of connectors. Only selected atoms will be copied.

If you have the SMILES or InChI string of your molecule on your clipboard (for example, from wikipedia) you can just paste it in AMSinput. It will automatically be converted to a 3D structure (via OpenBabel). It works often, not always, so be sure to check your structure.

10.8.3 Exporting your molecule

Exporting to file: .xyz, .mol, .cif or .bgf

Use the **File → Export Coordinates...** to export your current system.

In the file select box you will have the option to select the format. Currently supported are .xyz (default), .mol, .bgf and .cif (for periodic systems).

You can also Copy your system via the Copy menu. The contents of the clipboard will be identical to the .xyz file that would have been exported. For periodic systems, the lattice vectors are included as special atoms VEC1, VEC2 and VEC3.

10.9 Regions

A Region in AMSinput is defined as a collection of atoms. Atoms may be present in one or more region.

Within the GUI they are used for many purposes. For example to change the visualization style for part of the molecule, to set up fragment calculations, or to set up other multilevel calculations like QUILD, QMMM or FDE.

When using the slider to change distances or angles you can use regions to group atoms (so they move as one).

When you select the Regions panel (in the panel bar Model section) you will find that one region has already been defined: All. This region consists of all atoms of your molecule, and is always present.

To add a new region, click the + button. If any atoms are selected when you press this button, they will automatically be added to this new region. With this new region you can do a couple of things:

- Click the - button: delete the region. Note that only the region is deleted, not the atoms in it.
- The check box: this selects all atoms currently in the region. In your molecule display you can easily see what atoms are in it, and it will also tell you how many atoms are contained in this region.
- Click in the region name to change the name
- Click the check box (the V-like shape) to show the region graphically
- The + button (on the right hand side): add the currently selected atoms to this region
- The - button (on the right hand side): remove the currently selected atoms from this region
- Click the triangle to get access to a pop-up menu with a number of commands that apply to that region.

The pop-up menu per region has the following options:

- **Split By Molecule:** split the region in many regions, based on the connectivity. Thus parts that are not bonded together will all be put in a separate region. The typical use is for solvents: first add a solvent with explicit molecules. The solvent molecules will automatically be put in their own region. Next, use the Split By Molecule command to split this into many small regions. If all your solvent molecules are identical, this leads to a very big time saving with fragment or FDE calculations. As the typical use is for replicated fragments, the naming of the newly generated regions will use the /n name, with n a number. Currently AMSinput does not check if the fragments are really identical!
- **Invert Selection Within Region:** perform an invert selection command, but just within the atoms part of this region.
- **Region Color:** set the color used to visualize the region
- **Balls And Sticks:** show the atoms and bonds in this region using Balls and Sticks
- **Sticks:** show the atoms and bonds in this region using Sticks only (that is, atoms not visible)
- **Wireframe:** show the atoms and bonds in this region using a wire frame only (again, no atoms visible)
- **Hidden:** do not show the atoms and bonds in this region

Regions can also be used in the Numerical Quality and Basis panel: you can change the basis per atom type per region. For example, give all carbon atoms in an outer region a smaller basis. Or you can set the Quality per region (the Becke / Zlm fit options and/or Basis and Core).

A quick and easy way to make a new region containing the current selection is to use the **Atoms → New Region From Selected Atoms** menu command (cmd/ctrl-G).

10.10 Proteins (from PDB or mol2 files)

AMSinput can read a protein from a PDB file or from a mol2 file.

The atom information is read (element type, coordinates, PDB name, residue to which the atom belongs, and chain information).

The PDB atom names are also essential to properly add hydrogens, in case your PDB or mol2 file does not have any hydrogens.

In the **Regions** panel for each chain detected in the PDB file, there will be a new line. This gives you a couple of visualization options (Ribbon, or all C, CA, N or O atoms connected). If you click on the triangle to the right you get a pop-up menu with some additional commands:

- **New Region From Chain:** make a new region, and put all atoms belonging to this chain into the new region
- **Select Residue:** select all atoms in a given residue (which you select from the menu)
- **Residue Protonation:** change the protonation state of some residues
- **Add Hydrogen:** add hydrogen atoms to this chain
- **Add Selection To Chain:** the selected atoms are marked as belonging to this chain
- **Add Selection To Residue:** the selected atoms are added to the specified residue

Many of these features depend on the proper PDB names and residue information for the atoms. If they are not correct, or non-standard, AMSinput may for example add hydrogens to the wrong positions, or visualize things incorrectly.

For the protonation state of some residues it is essential that their proper names are used: **LYN/LYS, ASP/ASH, GLU/GLH, HID/HIE/HIP**.

You can use the **Atoms → Details (Color, Radius, Mass, ...)** command to check and change the PDB names of the atoms.

Having the proper PDB atom names and residue types (including the proper protonation state) is essential. The atom types generated for MM (Amber95) depend on them, as well as the algorithm that adds hydrogens (if needed).

The hydrogens that are added by AMSinput do not use a very advanced algorithm. For the protein itself it should be correct in most cases, assuming you have made sure that you have the proper protonation state of the residues. However, if you have solvent molecules in your PDB file the hydrogens will be added properly but the solvent molecules will be oriented randomly. So no hydrogen bond structure in water for example.

To avoid problems with the primitive algorithm AMSinput uses to add hydrogens you can obviously use some other tool to first make a proper PDB or mol2 file for your protein including hydrogens.

Once read in, you can use in principle all tools within the GUI as you like. For example, perform MM, QMMM or QUILD calculations for the protein structure.

10.11 Presets and Defaults

AMSinput presets define input values to be used for some input options, replacing the normal default values.

A preset normally sets just a few input options. After you have selected a preset, fields that are set by the selected preset will be shown with a green color.

Fields that are adjusted by the user will **NOT** be overwritten when using a preset, they will be marked with a red color if the preset has a different value.

A special preset **None** is included which sets no options, using it will revert all fields that are not adjusted by the user (all green fields) to their Default values.

Note: Details regarding preset matching to modules/tasks have been changed, and bugs fixed, in the 2023.102 release. If you used SinglePoint.tpl to define your defaults for everything, rename that file to *_*.tpl. SinglePoint is no longer a special task, and instead the * wildcard may be used for task or module.

10.11.1 When are presets used

AMSinput sets the defaults (for all fields) according to the Defaults preset by SCM or the user on startup. Next, other presets may be loaded:

1. Automatic preset loading

AMSinput tries to load a matching preset when the user switches from module (ADF, BAND, etc) or task. First the matching preset is determined. If it is different from the current preset in use (if any), it will be used. If no matching preset is found, and the current preset does not match the current module, the **None** preset is used.

2. User loading a preset

A preset is also used when explicitly selected by the user from the File -> Preset menu. Alternatively, right-click on a field to get a pop-up menu with options to reset that particular field, or to save a particular Preset.

3. User resetting preset values

Use the Reset option (Reset To AMS_Defaults for example) in the Preset menu to reset all values to the defaults, without applying any other preset. The **None** preset is similar, it resets all fields set by the current preset but leaves fields changed by the user as they are.

10.11.2 Saving presets

You can save a Preset using the File -> Preset -> Save... menu command.

Note: Many input options are shared between modules, like most AMS driver options. Adjusting them for one module will also adjust them for other modules.

If you make your preset module specific (for example by saving it with name ADF) it will only be used for that module. Switching to another module will reset to **None** (or some other matching preset).

In the help balloon you can find detailed information about the matching rules for presets, which determine the preset automatically used.

By default presets are saved in the \$HOME/.scmgui/Tpl directory. This can be changed by setting the SCM_TPLDIR environment variable.

If you save a preset with the same name as one of the default presets, it will effectively replace that preset. One exception: the SCM supplied Defaults are always read first, and thus cannot be replaced.

To delete your own preset(s), use the 'Delete' command from the Preset menu.

10.11.3 Matching presets with modules / tasks

When you switch the Task or Module, AMSinput will try to load a matching preset.

The Default preset is always loaded first, thus fields will be reset to their default value. Next the first matching preset will be loaded according to these matches:

- `Module_Task`: Module and Task must match (for example `ADF_SinglePoint`),
- `Module_*`: Module must match (for example `DFTB_*`),
- `Module`: (no underscore): Module must match (for example `BAND`),
- `*_Task`: Task must match (for example `*_GeometryOptimization`),
- `*_*`: always matches (for example `*_*`).

For presets to match, spaces and minus signs are ignored. The Module names should be as visible in AMSinput (in the orange menu).

10.11.4 Default values used

AMSinput always first uses the Defaults preset as included by SCM.

Next the Defaults preset defined by the user (if any).

Next default values defined in the code (via the `input_def` json files) will be used, but only if they have not yet been set.

Next a preset matching the current module and task will be loaded (either from SCM or defined by the user). Fields that are adjusted by the user will **NOT** be overwritten when using a preset, they will be marked with a red color if the preset has a different value.

Finally a user may force a preset to be used by using the File -> Preset menu. This will override the automatically loaded preset from the previous step. Again, fields that are adjusted by the user will **NOT** be overwritten when using a preset, they will be marked with a red color if the preset has a different value.

10.11.5 Color Code

The input fields use a color coding to warn you they have been modified:

- No special color: the field has its default value (from `Defaults.tpl` by SCM or by the user).
- Yellow: the field has been changed by the user (only).
- Green: the field has been changed by the preset (only).
- Red: the field has been changed by the preset, and next by the user.

The pull-down menus in the panel bar use a similar color-coding to point you to fields that have been changed:

- No special color: all fields in the panel have their original default value.
- Yellow: some fields have been changed by the user.
- Green: some fields have been changed by the preset.
- Red: some fields have been changed by the preset, and some (possibly also) by the user.

10.11.6 Input options remarks

Spin and Occupation

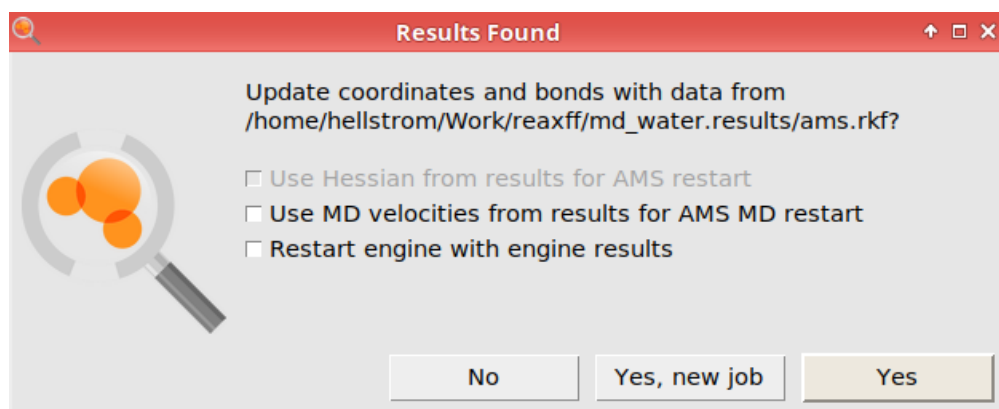
The spin and occupation panel allows you to specify the occupations of the orbitals per symmetry. In case of an unrestricted calculation you can also specify the occupations per spin type.

To show the available symmetries, AMSInput needs the result of an ADF calculation. If a previous calculation is available (without specifying the occupations), it will use the information from that calculation to generate the proper options in this panel. If such results are not available, AMSInput will suggest to run a short guess calculation: a preliminary run with an inaccurate grid, only a few SCF cycles and stopping immediately after the SCF. Hopefully this guess calculation will allow you to generate sensible occupation.

The energy levels of the guess calculation (or previous calculation if available) will be shown using AMSlevels. Be aware that it is the result of the guess calculation, and not your proper results!

10.12 Update geometry dialog

When opening AMSInput for a finished job, or if the AMSInput window is open when the job finishes, the following dialog pops up:



If you click

- **No:** The original input is preserved.
- **Yes, new job:** This creates a new job, so that you do not overwrite the original job. The geometry is updated from the results file, and new input options are set based on the checkbox selection (see below).
- **Yes:** This reuses the same input file. The geometry is updated from the results file. **You should not check any of the checkboxes**, since a job cannot be restarted from itself.

10.12.1 Meaning of restart options

- **Use Hessian from results for AMS restart:** If you calculated the **hessian, normal modes, or vibrational frequencies** in the original job, you can reuse the calculated hessian in a new job. This can help, for example, to converge a geometry optimization or transition state search faster. If you select the option, then on the **Details → Geometry Optimization** panel the **Initial Hessian** keyword is set to **From File**, and the **Initial Hessian from** is set to the engine.rkf file from the original job.
- **Use MD velocities from results for AMS MD restart:** If the original job is a molecular dynamics simulation, this option will load the velocities from the final frame from the original job, allowing you to continue an MD simulation. Note that if you use thermostats or barostats, the continuation trajectory will not perfectly match the trajectory which “would have” taken place if the original job was longer. If you select the option, then on the **Model → MD** panel, the **Initial velocities** keyword is set to **From File**, and the **File** is set to the ams.rkf file from the original job.
- **Restart engine with engine results:** If the original job is an ADF/BAND/DFTB job, this option lets you continue from the converged SCF solution in the original job. The other engines do not use engine restart information. If you select the option, then on the **Model → Restart** panel, the **Engine restart** keyword is set to the engine.rkf file from the original job.

AMSJOBS

11.1 Jobs

11.1.1 Job list: per folder or global

When AMSjobs starts, it will generate a list of jobs for you by scanning the local directory. All files that have the same file name, with only a different extension, will be considered to be one job. A job may also contain a directory, it should have the .results as extension.

AMSjobs can also show a global list of jobs. All recent jobs that it knows about (by previously scanning a directory, or by running them) are shown. This typically can be a huge list, and it is limited (currently) to 500 most recent jobs. The big advantage is that you do not need to remember where you saved a job, it will automatically show in the global list. To switch between the two modes (showing only jobs in the current directory, or showing all known jobs) use the toggle switch at the right bottom corner.

If you are viewing the global list of jobs, you can use the **View → Job Location** menu command to show where on your disk the jobs are located. For each job one extra line will be displayed with this information.

The job list refreshes automatically when needed. You can force a refresh via the **Job → Refresh List** menu command, or by pressing F5. For each job you will see:

The Filter command determines what kind of jobs will be shown. For example, you can choose not to show directories, or only ADF jobs.

At the bottom right you will find a toggle button. This button will toggle between the two modes of AMSjobs: show all jobs, or show the jobs per folder.

11.1.2 Job status (including WARNING and ERROR info)

The status icon is the icon on the right side of the job. It tells you if the job is a new, queued, running, terminated, ready, ready with a warning, or ready with an error condition.

The warning and error condition is determined from the logfile of the job. If it contains a WARNING, the icon will display a warning triangle. If the logfile contains ERROR, the status icon will change into a red stop sign. If the standard error output of a job contains unexpected information this will also be reported as an ERROR.

The details of the error or warning will be shown in a color just below the job. Obviously, when a job has ended with an error you will normally not have useful results. If a warning has been printed, you should make sure you understand the warning, and that the ADF did perform the calculation that you intended.

11.1.3 Selecting Jobs

You can select one or more jobs with:

- left click - select that job
- shift left click - select a range of jobs
- right click (or control left click) - toggle that job to be selected or not selected
- click on the search icon - clear the selection

To clear the selection, press the ESC key, use the **Job → Clear Selection** menu command, or click somewhere in white space on the bottom or in the selected job (thus not on the name, queue, or one of the icons).

11.1.4 Running a job: .run and .job

When saving your job, AMSinput saves a .ams file and a .run file. The .ams file contains all information for AMSinput, and if you wish to make changes to your job use AMSinput to make them. AMSinput will read the .ams file.

The .run file contains the basic commands and input to run your calculation. It is intentionally kept as simple as possible.

However, typically some more administrative things need to be done: make empty working directories, make some links to follow a running calculation, etc. This used to be done by the run script, but not any more. If you wish to use the run script yourself you are responsible of taking care of such details.

To run your calculation, use the Run command from the File menu. This will tell AMSjobs to run your job. Alternatively, you can switch to AMSjobs, select your job (that you should have saved from AMSinput), and select Run from AMSjobs Job menu.

AMSjobs will create the real job script (with .job extension). This is a the .run script as saved by AMSinput, with the administrative things included at the front and at the end.

As the .run script is simply included, you may edit it if you wish, and AMSjobs will automatically include your changed .run script.

If the environment variable SCM_RESULTDIR has been set, the job script will change into that directory. Next it will run from there, and all result files will be stored in that directory.

If the environment variable SCM_RESULTDIR has not been set, the job script will execute in the directory where it is located, and the result files will also be located in that place.

The exact command given to run the job depends on the queue configuration.

11.1.5 Job files

Local files

The local files section lists all files that AMSjobs found. They all have the same name (the name of the job), and different extensions. Here you can see what files belong to a job, including modification date and time, and the size of the files.

Double clicking on some of the extensions will open that file. For example, double clicking a .ams file will open it in AMSinput. Double clicking a KF results file will open it in the Kfbrowser.

One useful application is double click on the .run file. This will open the .run file in a text editor, depending on your operating system. In this editor you can actually make changes to the .run file. When you save it this modified run file will be used when you run the job. The .job file will be overwritten, thus you need to change the .run file if you wish to make manual changes.

You can also first select a local file. Next select one of the GUI modules from the SCM menu, and the selected file will be opened by that GUI module, if possible.

Within the local files you may also find a directory called 'results'. This will be created when result files are present other than the standard result files. For local jobs they will always be present, for remote jobs it will be created when using the Transfer from Remote command or when you click on one of the remote files in the remote .results folder, as displayed in AMSjobs.

You can select multiple files using click followed by shift-click (for a range), or using control-click to add the clicked file to the selection. Once multiple files are selected, you can open all of them at the same time via the SCM menu (either from the menu or via a right-click in the selection). Alternatively, you can use the Backspace or Job->Delete menu command to delete the selected files.

Remote files

This is a similar list of files as the local files list, but these files reside on a the remote machine as specified in the job details. If you are preparing a new calculation it will be empty. When a calculation is complete, it will show all the result files on the remote machine.

At the top of the list the name of the directory in which the files live on the remote machine is shown.

You can select a remote file by clicking on it. This will cause the file to be transferred to the local machine, replacing any local file with the same name. If you click on any of the remote files in the .results directory all remote files belonging to the job will be transferred (overwriting local files with the same names).

After selecting the remote file it will be used by GUI modules started from the SCM menu.

11.1.6 Clean Up

After a calculation many files may be saved. Some of them might not be of interest for you, or only temporarily. You can use the **Job → Clean Up** command to remove files no longer needed. It will present a dialog where you can select exactly what files to remove. The most important result files will by default be saved.

Note that you can also just select the files you want to delete and press backspace.

11.1.7 Archive jobs

You can use the **Job → Archive** command to collect most files belonging to a job and put them in a gzipped tar archive. This will significantly reduce the amount of space used by the job. The command will operate on all selected jobs.

If you use an archived job in AMSjobs (or any GUI module) will recognize it as archived, and automatically unarchive it before using. So in practice you can just archive many (or all) of your jobs and forget about it. This will save a lot of disk space.

11.1.8 Import / Export jobs

If you want to share jobs with other users (or SCM support) there is an easy way to do this. Just select your job in AMSjobs, and use the **Job → Export...** command. A dialog will appear that allows you to specify where to save the packaged job. The packaged job is in principle just a gzipped tar file containing all job files, with some extra data so that AMSjobs recognizes it as a packaged job.

Use the **Job → Import...** command to open such a packaged job and add it to your AMSjobs listing.

11.1.9 Find job files

After some time you may have a lot of result files collected in the same folder. The **Job → Show Files** command will show the files belonging to the selected job on your computer. How it does this exactly depends on your operating system. Typically a window will open showing the files in the directory, with the files of the job selected.

11.1.10 Test Job

When you have trouble running jobs, either locally or via some queue you have defined, it is sometimes not easy to figure out what the problem is. To help you (or us, when you ask for support) diagnose the problem AMSjobs can make a test job.

Use the **Job → Generate Test Job**, select it in your AMSjobs window, assign the queue you want to use (or skip this step if you want to test the default queue), and use the **Job → Run** command.

The logfile of that job will contain a lot of diagnostic information, like the environment, license details, adf version details and more. Some consistency checks are performed. This information might help you to solve the problem. If not, contact us, explain the problem and include the output of the test job.

11.2 Queues

A queue tells AMSjobs how to run the selected job: where (possibly on a remote system!), how and by whom.

In the Queue menu you see a list of queues. Select one of them to use that queue when running the selected jobs.

If you have configured queues for remote machines, you will be able to use those remote machines just as easily as your local machine. AMSjobs will take care of copying files to and from the remote machine. It will also start or submit your job, and inform you of the progress of your job.

If you have defined your own queue, for example to run on some remote cluster, you can make that the default queue (instead of the Sequential queue) by using the **Queue → Set Default** menu command.

For each job, you can specify some extra text in the options field next to the name of the queue (with the gray rectangle around it). How this text is used depends on how your queues are set up. For example, the Interactive queue uses it to specify the number of tasks to use in your job. For batch systems, it might be the number of nodes to use, or some time limit or batch queue name.

When starting AMSjobs the first time, you will see the Interactive and the Sequential queue. Both will run jobs on your local machine, using as many tasks as possible. You can enter a number in the options field of the job (with the gray rectangle) to set the number of tasks use explicitly.

Via the GUI Preferences, you can also configure AMSjobs to automatically pick up queues stored in a central location. They need to be defined once, and any AMSjobs user can import them. Such queues are called ‘Dynamic queues’.

Warning: The `Job status` command in default queues have been updated, to avoid excess load on the batch system.

For PBS the `qstat` command has been changed into `qstat $jid`. And a similar change for the other batch systems, the syntax depends on your batch system.

You may want to change your own queue in the same way.

11.2.1 Interactive Queue

When you run AMSjobs for the first time, it will make sure that an Interactive queue exists. If not, it will automatically create such a queue for you.

When you use the interactive queue to run a job, your job will run immediately on the local machine. Thus you can run many jobs at the same time.

To specify how many tasks to use, enter a number in the options field. If you leave it empty all cores will be used.

As you could be overloading your machine it may not be what you want, but it is great if you have some job running and want another small one to run at the same time. Another use would be to run several single-core jobs on a multi-core desktop machine at the same time.

11.2.2 Sequential Queue

When you run AMSjobs for the first time, it will also make sure that a Sequential queue exists. If not, it will automatically create such a queue for you.

When you use the Sequential queue to run a job, your job will run interactively on the local machine as soon as no other job is running. Thus you can give the run command in AMSjobs for many jobs at the same time, but they will actually run one after the other.

To specify how many tasks to use, enter a number in the options field. If you leave it empty all cores will be used.

Normally, this is the most convenient and efficient way to run jobs on your local machine. For that reason it is the default queue (unless you change that).

11.2.3 Setting up your own Queues

You can define a queue in several ways:

- **Queue → New...** based on one of the included example queues,
- **Queue → Edit...** change details of a queue (or make a new one if you change the queue name as well),
- **SCM → Preferences → AMSjobs → Dynamic Queues** import queues stored on some central system

When using the **Queue → New...** command, you can select what configuration to start with:

- **Interactive:** make a new Interactive queue. For example, to make a special version that will force your job to use only 1 task.
- **LSF, PBS, SGE:** make a new queue that will submit your job to the selected batch system. The configuration of these batch systems can be quite different. The included examples should serve as an example only, you will need to fix the details. In the included examples the options field is used to specify the number of nodes to be used.

When using the **Queue → Edit...** command, you select what queue configuration to edit.

Using either of these commands, a dialog will appear requesting you to set the details of the queue you are creating (or modifying).

Often it requires some experimentation to set up all the details correctly. You can use the **Help → Trace Commands To Remote** menu command in AMSjobs to trace the commands given to communicate with the remote host (typically the ssh or putty command with arguments), and to see the responses to these commands received from the remote host. This trace will be shown in via AMStail.

The following details may be set when configuring a queue:

Remote host

Name of the machine on which you wish to issue the command to run (or submit) your job. You should be able to connect to that machine using ssh, and the host name as specified here. If you wish to run on your local machine, leave this field empty or specify localhost.

Remote user

The username that you need to specify in the ssh command, if any. Typically, this is your username on the remote machine. If you have configured ssh to log in on your remote machine without specifying a user name, you can just leave this field empty. That is also the most convenient way to set up a queue that is to be used by other people as well (with their own accounts on the remote machine).

Remote job directory

On the remote machine, AMSjobs needs to set up your input files and run script, and needs to collect the results. For that purpose AMSjobs will make its own directories within the directory you specify. A typical value would be something like \$HOME/jobs.

Run command

The command **on the remote host** to be executed. AMSjobs will always first ssh into the Remote machine, and then issue the Run command. The run command should start the job interactively, or submit the job using some batch system. If the remote host field is empty or localhost, the run command will be executed on the local machine.

\$job will be replaced by the full path to your job script on the remote machine.

\$jobname will be replaced by a jobname based on the value of \$job, but truncated and with spaces removed.

\$options will be replaced by the contents of the Options field. The Options typically will be used to specify the number of tasks, a time limit, or a batch queue name.

If you use the run command to submit a job to some batch system, it should return a number. This number will be assigned to \$jid, and may be used by the kill and job status commands.

To run interactively, just enter “sh \$job”. To submit your job to a queue, specify the submit command (for example, qsub, or some other special submit tool). For example, check the pre-configured queues for Interactive and batch systems (via the **Queue → New...** menu command).

The job script that is automatically generated accepts an optional parameter. This parameter is ‘eval-ed’ at the start of the script. Thus, you can use it to set environment variables (like NSCM) or other things.

When the run command is issued, the working directory will be \$SCM_RESULTDIR or the location of the .job script if no SCM_RESULTDIR has been defined. The results (output file and so on) will also be saved in that location. Note that the computational engines (ADF, BAND, ...) will automatically change their working directory when started to \$SCM_TMPDIR. You should make sure that SCM_TMPDIR points to the proper location as you want to make sure you are writing to a fast local disk, and not to the (typically shared) disk where the results are stored.

Use Local Batch

If yes, jobs will be queued on the local machine. Only one job will be running at a time. This is set for the Sequential queue (which is default). Currently this value is ignored when the job will be queued or run on a remote system or using a cloud machine.

Kill command

The command to use to kill a queued or running job. In this command \$jid will be replaced by the job id (from the output of the run command), or by the process id. For interactive jobs, killall \$pid should work fine. This killall is actually replaced by a special script that takes care to kill adf and all child processes.

Job status command

This command will be used to determine if a job is still queued or running. If a job is no longer queued or running, it should return an empty string. Anything else will server as indication that the job is alive.

For interactive jobs ps -p \$pid | grep \$pid works fine.

For batch jobs the command typically is `qstat $jid | grep -w $pid` (note: the `$jid` after the `qstat` was added recently, and the exact syntax depends on your batch system).

The job status command is issued every 5 minutes by default, per job, while the job is running or queued. You can change that by setting the `SCM_STATUSINTERVAL` environment variable on the remote machine (where the batch commands are executed).

System status command

The command to use to determine the system status. This might be uptime, or some `qstat` command for batch systems.

Prolog command

The command to execute by the job script, just before the included `.run` script. This will be used to set up the environment properly. For example, you would source a script file to set all environment variables for ADF like `AMSHOME`, `AMSBIN` etc. This is especially important if you are working with different versions of ADF at the same time. Note that the job script is started using `/bin/sh`, so you should use sh-like syntax (and not csh-like). If your environment is already set up correctly for running ADF on the remote host, without issuing any commands, you can leave this field empty.

In your prolog command you can use environment variables that are defined by the job script. Most importantly, the environment variable `SCM_OUT` points to the output file of the job, `SCM_LOG` points to the logfile, and `SCM_ERR` points to the error output. This makes it easy to print some additional information to those files.

You can put multiple commands here separated by “;”. However, in most cases it is easier to source a file with commands. By sourcing it you make sure that changes in the environment defined in the sourced file will also be available by the job script.

Epilog command

This is the command to run by the job script, just after the included `.run` script. You can use it to copy save result files, or to perform some cleanup action. Again, use sh-style syntax, and the same environment variables are still defined..

Copy results back

If this field is set to ‘yes’, all result files from this job will be copied back to the local machine automatically when the job is ready (only for remote jobs).

Cloud init

When this field is not empty, it will be used to initialize a cloud machine, if needed. A typical value for AWS machines would be ‘`AWSInit adf2019.204 t2.micro laptop`’. Check the Cloud computing description for details.

Cloud ssh key

The path to the private ssh key for the cloud machine. If it is an absolute path it will be used as is, otherwise it will be relative to the home directory. The default is empty, in which case the default ssh key will be used (possibly defined in your `.ssh/config` file).

11.2.4 Cloud computing

You can use `AMSjobs` to run jobs on machines that are in the cloud. Currently the Amazon Elastic Cloud is supported (AWS EC2).

To use AWS:

1. Create an AWS account or ask the person responsible for managing your organization’s AWS account to create an IAM user for you. Log in to the AWS Console using the credentials and go through the following steps:
 - a. Generate an ssh key pair: in the left column AWS EC2 Dashboard you will find ‘Key Pairs’ in the ‘Network & Security’ section. Click on that, and next on the ‘Create Key Pair’ button on the top. Enter a name without spaces, for example `aws_ssh_key`, and the newly generated private key will be saved on your computer. Move it to a safe location (like your `.ssh` folder). Make sure others have no permissions to read/write/execute that file.

- b. Modify the default security group that will be used for each new instance, to allow inbound ssh connections from your IP address (or an address range). You can also restrict the outbound traffic but you need to make sure that the communication to SCM license servers (license.scm.com, license1.scm.com, license2.scm.com) on the https port (443) is not blocked. The security groups are found under 'Network & Security' in the EC2 Dashboard.
 - c. Create an AWS access key for the command-line interface: click 'My Security Credentials' in the account drop-down menu, press the 'Create access key' button under 'Access keys for CLI, SDK, & API access', store the key ID and the secret in a safe place. You can create extra keys at any time.
2. Install AWS CLI (the AWS command line interface).

For instructions check the [Amazon AWS CLI](https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-install.html) (https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-install.html) documentation. Make sure you can use the AWS CLI as a normal user (try `aws --version` in a terminal window).

For example, on a mac

```
sudo chmod -r ugo+rX /usr/local/aws/  
sudo chmod -r ugo+rX /usr/local/bin/aws
```

is needed at the time this documentation was written.

3. Configure the AWS CLI by running `aws configure`. Enter the access key ID, the corresponding secret and the AWS region when prompted. The default output format does not matter so you can leave it at the default.
4. Have a look at AMIs (Amazon Machine Images) available to you, by name or by ID. You need to know which AMI to use for your cloud instance. Pre-existing AMIs can be found under 'Images' on the EC2 Dashboard. You can, for example, search by name.

Optionally, you can also create your own AMI containing ADF from one of the standard Linux images provided by Amazon:

- Start a Linux instance,
- Copy an ADF archive and unpack it in the home directory of the default user,
- Edit `~/.bashrc` and add the following command at the top of the file, for example for ADF installed in `~/adf2019.301`:

```
source $HOME/adf2019.301/amsbashrc.sh
```

- Install the task spooler `tsp`. This is used by the default AWS queue in AMSjobs to get jobs to run in sequential mode (one after another):

```
sudo apt install task-spooler
```

5. In AMSjobs on your local computer make a new AWS queue.

The easy way to set up an AWS queue is to base it on the AWS example queue included in AMSjobs (use the **New...** command from the **Queue** menu). Make sure to change the `ssh_key_name` and the SCM cloud credentials.

In general the following queue options are relevant for AWS queues:

Queue name

'AWS' (or something like that, it's up to you)

Cloud init

Specify how to initialize cloud instances. Currently only AWS is supported, for which you need to use `AWSInit`.

For example `AWSInit adf2019.204 t2.micro aws_ssh_key`

AWSInit image-name instance-type key-name [--size size]

`AWSInit` manages AWS cloud instances. It starts new cloud instances when needed, or reuses previously started cloud instances if all arguments are identical. It will take care to pass the IP and username that should

be used with the new cloud instance to AMSjobs. The hostname field of the queue is ignored, the username field of the queue is used and can be used to override the username returned by ASWInit. Currently ASWInit always returns ubuntu as username.

image-name

The name of an AMI image (optionally using wildcards), or the id of an AMI image. If multiple images match the one with the latest creation date will be used. The search is for AMI images owned by you (self) or by SCM. You can use wildcards to get the latest adf versions: `adf*` to get the latest, or `adf2019.2*` to get the latest adf2019.2xx

instance-type

The instance-type to use (see the AWS documentation. Note you may use \$options in the AWSinit line to specify the actual instance-type

key-name

The name of your key as known by AWS (should have been created in step 1a)

size (optional)

the size of the disk space in GB

Copy results back

yes

Be sure to set this to yes, otherwise your results are gone once the cloud instance is terminated

Path to private ssh key

Set it to the path of the private key you saved in step 1a.

AMSjobs will automatically start an AWS cloud instance if needed. If a cloud machine using the same AWSInit command (with \$options replaced by the actual value) is already running it will be reused. Once it is ready the job will be started on that cloud machine, and from this point it is just like any other remote machine. If you do not want to reuse an already running cloud instance you need to make a new queue, with a different Cloud init command. For example, use a different image-name (which actually may be the same image), or add a -size option.

Starting an AWS cloud instance may take a while. AMSjobs will be waiting for it, and show this using a progress dialog. You can click the **Dismiss** button which will stop the submission of the job. It will NOT cancel the starting of the cloud instance. So you can use the Run command again at a later time and AMSjobs will automatically pick up this same instance (either use it or continue to wait for it).

Once you are done with this remote machine do not forget to terminate the cloud instance. You can do this via the AWS web interface. Alternatively, when you Quit AMSjobs it will propose to terminate the running cloud instances, if it can find any.

Obviously terminating cloud instances while they are being used is not a good idea. The GUI will run into lots of TIME-OUTS as it cannot reach the cloud instance any more, while it expects it to be alive. So before terminating a cloud instance make sure you have no jobs running that are using it.

11.2.5 Dynamic queues

Dynamic queues are updated automatically when AMSjobs starts.

AMSjobs will check with the hosts that you have specified in the Preferences:

```
Open GUI preferences: SCM → Preferences
Panel bar Module → AMSjobs
Click the + button in front of Dynamic Queues
Enter the host name of the host on which to look for dynamic queues
Optionally: enter the username of the remote machine on which to look for dynamic_
```

(continues on next page)

(continued from previous page)

→ queues
Click the 'Save' button at the bottom of the panel, and restart AMSjobs

If the username is left empty (or to the value (username)), no username will be used when connecting to the remote machine. Then your ssh configuration determines what username to use.

On the remote hosts listed, AMSjobs first checks the \$SCM_QUEUES environment variable. If it is set, it will import the queues defined in the \$SCM_QUEUES directory. If it is not defined it will try to import queues from \$HOME/.scm_gui. This is the location where AMSjobs stores the queue information.

To define the dynamic queues, first figure out what queue settings you (and/or others) want to use. You can do this by configuring a normal queue with AMSjobs as described. As dynamic queues typically will be used by many users, you should not specify a username (unless you want all users to use the same account on some system). Make sure it works properly.

Next make a directory on the remote system where you want to store the dynamic queue definitions.

Set the SCM_QUEUES environment variable system wide on remote system for all users. If SCM_QUEUES is not defined, AMSjobs will search the \$HOME/.scmgui directory on the remote platform. This way users can set up their own dynamic queues.

Locate the files that define your queue: you can find these files in the \$HOME/.scm_gui directory. They have the name of the queue, with a .tid extension, and are plain text files. Next, copy these files into the \$SCM_QUEUES directory on the remote system. Make sure all users have permissions to read the \$SCM_QUEUES directory and the files in it.

Note: the server that stores the dynamic queues need not be the same machine on which the jobs will run.

Note: the AMSjobs user needs to have access to the server via ssh

11.2.6 Example Queue configurations for SGE and PBS

Here you will find example .tid files for SGE and PBS. They will not work without change, you need to set at least the proper hostname and runcmd, and most likely the prolog needs to be changed (or just made empty).

The contents of the .tid file for a SGE queue might look something like this:

```
# hostname: machine.domain
# username:
# jobscript:
# prolog: source $HOME/setup/adf2017
# epilog:
# jobdir: $HOME/jobs
# runcmd: qsub -pe s3_mpich $options -q short3.q "$job"
# batch: no
# options: 2-2
# killcmd: qdel $jid
# jobstatuscmd: qstat | grep " $jid "
# sysstatuscmd: qstat
# label: My SGE queue
# logfile: logfile
```

Similar, for a PBS queue it might look something like this:

```
# hostname: machine.domain
# username:
# jobscript:
# prolog: source $HOME/setup/adf2017
```

(continues on next page)

(continued from previous page)

```
# epilog:
# jobdir: $HOME/jobs
# runcmd: qsub -lnodes=2:ppn=2:infiniband -lwalltime=$options "$job"
# batch: no
# options: 0:15:00
# killcmd: qdel $jid
# jobstatuscmd: qstat | grep $jid
# sysstatuscmd: qstat -q
# label: My PBS queue
# logfile: logfile
```

11.3 Tools: set up many jobs and collect data from many jobs

You can use the Prepare tool to set up batches of jobs. For example, first set up an ADF calculation with your preferred basis set, XC potential and so on using AMSinput. Next, use the Prepare tool to generate a batch of similar jobs, but for different molecules (taken from .xyz files for example). Or you could set up a calculation for your molecule, and generate a set of jobs with different XC potentials and / or integration accuracies.

The Reporting tool is to generate a report of one or more calculations. This report will contain the information that you select when you define a 'report template'. Most of the properties that have been saved to .t21 will be available. And you can generate images as well (like HOMO or SCF density). These results will typically be collected in an HTML table: one row for each molecule, and one column for each property.

A report template defines what information to put into the report.

11.3.1 Prepare: set up many jobs at once

You activate the Prepare tool via the **Tools → Prepare...** menu command.

A window will appear that you can use to specify how to generate a set of jobs.

Three main list fields are presented: the Run field, the coordinates field, and the input options field. In each of these lists you can specify multiple options. When pressing 'OK' AMSjobs will generate the jobs by combining the options in all possible ways.

Run list Select one or more .ams files to run. A .ams file is just a calculation that has been set up using AMSinput before. Alternatively, one may use one of the predefined .ams files as present in the pull-down menu when you press the '+' button. To add a .ams file, use the pull-down menu, or specify a file name in the text field and press return. You may use wild card in the text field, so the default value (*.ams) will expand to all .ams files in the current directory. To remove something from the list, select it and press the '-' button.

Coordinates list When this list is empty, the molecule as found in the .ams files will be used. When one or more sets of coordinates is present in this list, the molecule in the .ams file(s) will be replaced by the molecules as defined in the coordinate files. You may use .ams files, .xyz files, .t21 results files, .mol files, .pdb files or whatever other format AMSinput can use with the 'Import Coordinates' function. By listing multiple files here your .ams files (that you listed in the Run: list box) will all be executed with each of the molecules in turn. Thus, if you specified two run files (for example a Single Point calculation, and a Geometry Optimization), and three molecules, you will end up with 6 jobs.

Input options list In this list you may define alternative input options. The corresponding input options in the .ams file will be replaced by the values that you specify here. So if you specify two different basis sets, each job will be replaced by two new jobs, one with each basis set that you specify. You may also specify other things, like integration accuracy and so on. If you specify only one value, that value will be used in all jobs. If you specify multiple values (by repeatedly adding the option) you will generate multiple jobs.

The text field may be used to add additional keywords, or replace existing ones, with the value specified. These options will be added to the list of options by pressing return in the text field. The values will be used as the '-k' argument to the amsprep command. For detailed information about this please check the amsprep documentation.

Produce jobs options The final fields will tell the prepare tool where to generate the jobs (the directory is relative to the current directory, and will be created if it does not yet exists). Also one big job will be created that is just a concatenation of all the individual jobs. When running interactively it might be more convenient to run this job instead of all the individual jobs. The results should be identical, the big job will produce files that look as if they have been produced by the individual jobs.

11.3.2 Report: collect data from many jobs

To make a report of calculation results, you first have to set up a 'Report Template' that defines what information should be collected for the report. You manage these report templates via the New, Edit and Delete Report Template menu commands in the Tools menu.

When editing a report template, a dialog box will appear with many options. Just check the options for the information that you wish to collect in your report. Note that you can also include images (of orbitals, and so on) in the report.

If you wish to include something that is not present in the dialog, you can use the last field: Extra AMSreport command line options. Whatever you specify here will be passed to AMSreport to generate the report. This allows you to get any information that is available on a .t21 result file into your report. Check the AMSreport documentation for syntax details.

Once you have your Report template set up, you can use the Build command to actually generate a report. The data for the report will be taken from the currently selected jobs. You can update the report with new data from other jobs by using the Update Last Report command, with one or more different jobs selected. These jobs may even be located in a different directory. This allows you to collect information from jobs a few at a time.

11.4 Series of jobs depending on each other

In AMSinput you can specify (in the Coordinates panel) that it should use the molecule from some other file or job. Then a job is created that depends on another job or on some other file (for example a .sdf file).

AMSjobs shows such dependencies when you select a job.

To run such a series of jobs that depend upon each other, you can just run the last job in the series. AMSjobs will determine what other jobs need to be run as well, and automatically start them (using the same Queue as the job you started yourself).

In the Tutorials you will find several examples of such jobs.

AMSVIEW

The AMSview module of the GUI is used mainly to visualize field data like orbitals, densities and potentials. Additionally, it can also visualize some scalar atomic data, tensor data, and Bader (AIM) results.

12.1 Visualization methods

Via the Add menu you can select a visualization tool. This tool (an isosurface, or a cut plane for example) can be used with any kind of field. After selecting the command to add the visualization tool you will get a new extra control bar at the bottom of the window.

The following visualization tools are available:

- Isosurface: a simple isosurface through a scalar field
- Colored Isosurface: an isosurface through a scalar field colored by a second field
- Isosurface with Phase: an isosurfaces through a scalar field, colored by the complex phase (or sign for a real field). Typically used for orbitals.
- Multi Isosurface: multiple isosurfaces colored by isovalue
- Colored cut plane: a plane through some field, colored by the values of the field
- Contours cut plane: a plane through some field, with contour lines on the plane to show the values of the field
- Contours(+/-) cut plane: a plane through some field, with contour lines on the plane to show the values of the field, with negative contours using dashed lines
- Vector Field: many vectors on a grid, pointing as directed by some vector field. You can get a vector field by calculating the gradient of some other field for example. You can do this via the Calculated Fields menu command.
- StreamLines: an alternative to visualize a vector field.

In the control bar you select what field to visualize. The fields will be calculated on the fly. The leftmost field of the control bar (containing the name of the visualization method) also is a pull-down menu that you can use to access details, or to delete that particular visualization tool.

12.1.1 Spinor: spin magnetization density

Spinors are a result of spin-orbit coupled calculations. Visualization of spinors is more difficult than visualization of orbitals. A spinor Ψ is a two-component complex wave function, which can be described with four real functions ϕ : real part $\alpha \phi_\alpha^R$, imaginary part $\alpha \phi_\alpha^I$, real part $\beta \phi_\beta^R$, imaginary part $\beta \phi_\beta^I$:

$$\Psi = \begin{pmatrix} \phi_\alpha^R + i\phi_\alpha^I \\ \phi_\beta^R + i\phi_\beta^I \end{pmatrix}$$

The density ρ is:

$$\rho = \Psi^\dagger \Psi$$

The spin magnetization density m is:

$$m = \Psi^\dagger \sigma \Psi$$

where σ is the vector of the Pauli spin matrices σ_x , σ_y , and σ_z . A spinor is fully determined by the spin magnetization density and a phase factor $e^{i\theta}$, which both are functions of spatial coordinates.

The (square root of the) density and spin magnetization density are visualized as a double isosurface and a vector field respectively. The phase factor $e^{i\theta}$, reduced to a plus or minus sign, is visualized with the double isosurface and with the color of the vector field.

The main control bar is identical to the control bar of a normal isosurface. If you show the details, you will find that in addition to the controls available for a normal isosurface, you can also specify coloring information. In this case, the two numbers for the HSV colors define the colors of the minus and plus sign.

12.2 Fields

Fields can be things like orbitals or densities. You select them from the field pull-down menu in the control bars. In the Fields menu the Grid option determines on what kind of grid the field will be calculated.

As a convenient short-cut, you can switch fields using the up/down arrow keys

For many fields, like orbitals, transition densities etc selecting the option from the field pull-down menu brings up a window where you can select the field of interest. As an example, selecting 'Occupied Orbitals' will bring up a window that lists all occupied orbitals.

In these field select windows you can sort the fields by clicking on a column header. Clicking again on the same column header will reverse the sort order. When you use another command from the field pull-down menu the select window will close, unless you check the 'Keep Open' check box. That allows you, for example, to show all orbitals (both occupied and virtual) at the same time.

You can also create new fields out of these basic fields by combining them (calculated fields, for example the difference between two fields), or by interpolating them to get a finer or less dense field.

12.2.1 Steric Interaction

The Steric Interaction field is generated using Van der Waals radii to visualize steric bulk. The field is the minimum distance to the Van der Waals surface of the selected atoms. The radii are taken from the MM3 method by Allinger (N.L. Allinger, X. Zhou, J. Bergsma, *Molecular mechanics parameters*, *Journal of Molecular Structure: THEOCHEM* 312, 69 (1994) ([https://doi.org/10.1016/S0166-1280\(09\)80008-0](https://doi.org/10.1016/S0166-1280(09)80008-0))). An isovalue of 0 corresponds to the Van der Waals surface.

12.3 Properties

In the Properties menu you find commands to visualize scalar atomic info. Either by displaying the numbers, scaling the atom radii by these numbers, or by using the scalar values to color the atoms.

You also find some short-cuts to generate orbital plots: HOMO-1, HOMO, LUMO, LUMO+1, Density and Spin=Density. These commands will generate a double isosurface, and select the appropriate field for it.

Next are commands to visualize some tensor data, depending on what tensor data is available. The tensor is visualized as a sphere scaled in the directions of the eigen-vectors of the tensors by the eigenvalues of the tensors. For each visualized tensor type you will get a control bar that allows you to tune the visualization.

Finally, the properties menu allows you to visualize some Bader results. The Bader sampling shows you the integration grid, but with all the grid points colored that are together in the same Bader basins. The AIM (Bader) command visualized the critical points and critical paths.

12.4 Comparing data from several molecules

AMSview can handle more than one molecule at the same time. You can show fields for different molecules in the same window, you can create calculated fields to see differences, and so on.

The different molecules may come from different files, or from one result file containing multiple geometries. An example of the first situation would be two different calculations, with different XC potential, resulting in two different .t21 files. An example of the second would be the .t21 file from a NEB calculation. That file contains the information for all images, so you can see how (for example) the HOMO changes from image to image.

To add a new molecule from a different file, just open an additional file using the Open menu command from the File menu.

AMSview has a 'current' geometry. The molecule shown will be the one for the current geometry only.

To change the 'current' geometry, use the horizontal slider below the molecule window.

The visualization items (surfaces) might be filtered in such a way that only items related to the current geometry are shown. This is the default when visualizing NEB results: you want to see how the density or an orbital changes going from one image to the next (using the slider). If you open different files the default is to show visualization items for all geometries at once. Thus you might compare orbitals from one fragment with those from another. You can switch this behavior using the 'Show All Geometries' menu command from the View menu.

You can easily compare calculations on the same molecule that differ in something else than geometry. Just open both result files (.t21). Next, you can calculate differences between similar things. If you add a calculated field, you will find that the first command in the field select menu is used to select the geometry from which to take the data. Thus, you can select the same property from different files and compute the difference.

Warning: The current implementation has no possibility to adjust the orientation or the grid. In practice this means that you need to take care that the fields that you compare actually make sense to compare. This is only the case if the geometry of the molecules is identical and thus the grid is identical. Though this is very restrictive, you can make interesting comparisons for a given molecule: change due to different XC, basis sets or integration accuracy for example.

12.5 Temporary Files

AMView normally will run DENSF or BAND in the background. This means that it needs scratch space to store inputs and result files to be visualized. After normal termination of AMView (using Quit) all scratch files will be removed.

The scratch files will be created in the following location:

- If \$SCM_TMPDIR is defined : use the \$SCM_TMPDIR directory
- else, if \$HOME is defined: use the \$HOME directory
- else, if \$TMPDIR is defined: use the \$TMPDIR directory
- else try to use the current working directory.

12.6 Calculating Fields

AMView will show you a progress dialog with some options when calculating fields.

In front of the progress bar you will notice a right-pointing arrow. If you click on it the progress dialog window will enlarge, and you can follow the progress of the field calculation (by DENSF) in detail. If you scroll back in that window you can also examine the input given to DENSF. Obviously this output is also very useful when there is some problem calculating the field. Click the Close button at the right of the progress bar to close the progress dialog.

You will also see a Run button, and before it a pull-down menu that you can use to select a queue. When the calculation of a field is slow (for a fine grid or big molecule) you can press Run, and the calculation will be saved as a job in AMSjobs, and will then automatically be started to run in the specified queue. After the calculation has started you may quit AMView if you wish, the results should automatically be picked up when you start AMView again (and when the fields are available). As you can select the queue to use, you can also calculate the fields on some remote machine if you wish. See the AMSjobs documentation about the use of queues.

While a field is being calculated via AMSjobs, it will be visible in AMSjobs as a new job. The results will be stored with that job. When you start AMView for your original molecule it will detect the newly available fields automatically. It will move the newly calculated fields to it self, and delete the job that was used to calculate the fields.

When you close AMView after calculating some fields, AMView may ask you to save the field. If you save the field it will automatically be available the next time you start AMView (for this job). Be warned that saving many fields may take a lot of disk space. You can use the Clean Up command in AMSjobs to delete them.

AMSMOVIE

The AMSmovie module of the GUI is used mainly to visualize molecular geometries, and graphs related to it. For example, follow the progress of a geometry optimization, the trajectory of an MD run, or visualize the Potential Energy Surface.

Additionally scalar atomic data (like names, charges, atomic temperature, etc) and vector atomic data (forces, gradients, velocities) may be visualized when available.

13.1 Geometries

When opening a results file, AMSmovie will show the current geometry of the system. If multiple geometries are available, there will be a slider and buttons that allow you to play a movie, and to move to a specific frame.

If the calculation is a PES (Linear Transit) calculation, there may be many unconverged geometries as well as converged geometries. In the View menu you can toggle if you want to see all geometries, or just the converged geometries.

13.2 Scalar Atomic Properties

If atomic properties are available for a geometry, you can show them using the View menu:

- Atom Info: show the selected property as a text label
- Color Atoms By: use the selected property to color the atoms
- Atom Radius From: use the selected property to scale the atoms

Properties may be things like Name, AtomType, Charge (actually many different kinds of charges), ...

13.3 Atomic Vectors

If atomic vectors are available for a geometry, you can show them using the View menu: use the Vectors command in the View menu to select which vectors to show.

Examples of atomic vectors: gradient, force, velocity

When doing a geometry optimization via AMS, or an MD calculation via AMS or ReaxFF these may be present.

With the Scale Vectors menu command in the View menu you can make the vectors bigger or smaller (often needed to see very small vectors at all).

13.4 Graphs

AMSmovie can also make graphs of many properties (like energy, temperature, selected distances, angles etc) that change with the geometry. Check the tutorials for many examples.

Note that you can have multiple curves in one graph, and you can have multiple graphs.

You can save the graph in many different formats, and you can export the graph data using commands in the Graph menu. To delete a graph make it the active graph by clicking in it, and next use the Backspace key or the Delete Graph menu command.

In some special cases like NEB and PES you can also visualize a function of two variables in an XYZ style plot. Thus for a PES you can actually see the landscape as a function of two scan coordinates.

13.5 Temperature Profile

Temperature profiles, if calculated in an AMS MD run, can be visualized either as XY plot for a specific frame “Temperature Profile At Current Frame”, or as 2D plot “Temperature Profile At All Frames. Both commands live in the Graph menu.

You will have to select which temperature profile to visualize (along which cell axes). And you will need to specify over how many frames to average.

The average is a simple average over a block of frames of the required size. For frame i , averaging over n frames, the frames $\max(i-n+1, 0) \dots i$ will be averaged.

For example, if averaging over 5 frames, the value for frame 10 will be the average over frames 6, 7, 8, 9 and 10. For the first few frames, the blocks will be smaller. In the same example, the value for frame 2 will be the average over frames 0, 1, and 2.

The visualization at the current frame will show on the X axes the points along the selected cell axes, and on the Y axes the temperature. When moving through the trajectory the full graph will refresh to show the data for the current frame.

13.6 Trajectory Analysis

AMSmovie also provides an interface to some tools to analyze MD trajectories, apart from the graphs. They are available in the MD Properties menu.

- Autocorrelation functions (ACF, including diffusion coefficient from velocity ACF, and including the Fourier transform)
- Histograms
- Radial Distribution Function (RDF)
- Molecule Fractions: make graphs of specific molecules detected
- ChemTraYzer: use the Reaction Event Detection command in the Properties menu to analyze reactions happening

See also:

The [Trajectory Analysis](#) AMS utility.

13.7 Export trajectories to other formats

In the File menu of AMSmovie there are also commands to convert trajectories to other formats (**File → Export Trajectory As**).

If you have an AMS results file (with extension .rkf) corresponding to an MD trajectory you can convert it to:

- DCD file (.dcd, with corresponding .psf file)
- BGF (.bgf)
- Xmol (.xmol)
- XYZ (.xyz)

You can also use the **Scripting** menu in **AMSjobs** to perform such conversion.

KFBROWSER

With the Kfbrowser you can inspect all KF files produced by the ADF package. It has two modes: the default non-expert mode, and the expert mode.

The SCM menu **SCM → KF Browser** command will start or switch to Kfbrowser. Kfbrowser can also be started using `$AMSBIN/kfbrowser path/to/ams.rkf` on the command line.

14.1 Non-expert mode

You will see a sub-set of the data contained on the KF file. It may be empty, depending on the KF file. For KF files that the Kfbrowser knows about (t21, runkf, rkf and rxkf) it will show the results that may be of use for users.

If you select values you can use double click on the selection or use the **Selection As Graph** menu command to see the values in a graph.

You can use Copy/Paste to copy the selected data and paste it in a different program, for example Excel. The data will be copied as tab-separated so it will be pasted nicely formatted in your editor or spreadsheet.

14.2 Expert mode

In expert mode you will see all data on the KF file. This is mainly intended for developers. You should not assume any meaning to anything shown here unless you are familiar with the code that produced the KF file.

SPREADSHEETS (.XLSX)

15.1 Export AMS results to spreadsheet format

Starting from AMS2021.1, you can export the results of an AMS calculation to spreadsheet (.xlsx) format.

You can

- convert a single AMS results folder to spreadsheet format
- compare multiple AMS results in a single table

15.1.1 Create a .xlsx file for a single AMS result

In AMSjobs, select a finished job and go to **Tools → Build spreadsheet**. This will create a file called `jobname.xlsx` in the `jobname.results` folder, and open it in your default spreadsheet viewer.

You can also double-click on the file in AMSjobs to open it.

Note: Your spreadsheet viewer **must be able to evaluate formulas**. We recommend that you use **Microsoft Excel** (for Windows, Mac) or **LibreOffice Calc** (for Windows, Mac, Linux). Other spreadsheet programs (such as Apple's Numbers, Google Docs, and Gnumeric) may not be able to calculate all formulas, or may lack some features like conditional formatting.

You can also choose to generate the spreadsheet when setting up the job in AMSinput. Select **Details → Summary as Spreadsheet (.xlsx)**, and set the output and unit options to your preferences.

To generate a spreadsheet file from the **command line**, run

```
"$AMSBIN/amspython" "$AMSHOME/scripting/scm/xlsx/amsresults2xlsx.py" /path/to/jobname.  
↪ results
```

You can also pass a `--help` flag to the above script to see some customization options.

15.1.2 Generate spreadsheet by default for all jobs

To automatically generate a spreadsheet at the end of every job that you set up via AMSinput,

- set the **Output** option on the **Details → Summary as Spreadsheet (.xlsx)** panel to either **job.results/job.xlsx** or **job.xlsx**,
- Follow the instructions for *Presets and Defaults* (page 51)

15.1.3 Compare multiple AMS calculations in a spreadsheet

Graphical user interface: Select multiple jobs in AMSjobs, by holding Shift while clicking on the job names. Select **Tools → Build spreadsheet**.

This creates a spreadsheet containing a table that you can filter or sort based on a number of properties, for example

- Job name
- Engine, task
- Elapsed time, termination status
- Energy, zero-point energy
- HOMO, LUMO, band gap
- Dipole moments
- Excitation energies, oscillator strengths
- Lattice vectors and parameters
- Reaction energies and barriers

Command-line:

```
"$AMSBIN/amspython" "$AMSHOME/scripting/scm/xlsx/amsresults2xlsx.py" --multi \  
  jobname1.results jobname2.results [jobname3.results...]
```

Pass the `--help` flag to the script to see more options.

15.1.4 Units

The generated spreadsheet contains a worksheet called **Units**. Under the heading **Current units**, you can see and change the units used throughout the spreadsheet. By clicking on one of the yellow cells, a drop-down list appears with the available options.

Unit	Comment
Angles	degrees; cannot be changed
Dipole moment	
Electronic energy	Orbital energies, excitation energies, band gaps, ...
Energy	Potential energy, kinetic energy, ...
Entropy	expressed in energy_unit/K
Frequency	cm ⁻¹ ; cannot be changed
Gradients (forces)	
Heat capacity	expressed in energy_unit/K
Hessian, 2D stress	
Length	
Mass	
Pressure, 3D stress	
Quadrupole moment	atomic units; cannot be changed
Temperature	Kelvin; cannot be changed
Time	ps; cannot be changed

15.1.5 Distributions and spectra

A spectrum (e.g. the IR spectrum) is calculated in the spreadsheet using **Gaussian broadening**. If you prefer a different type of broadening (e.g. Lorentzian), you can do this in the GUI module AMSspectra.

15.1.6 Evaluate formulas with LibreOffice

The LibreOffice Calc spreadsheet program does not by default recalculate formulas in .xlsx files. If you use LibreOffice Calc, we recommend that you (in LibreOffice Calc) go to **Tools -> Options -> LibreOffice calc -> Formulas** and set **Excel 2007 and newer** to **Always recalculate**.

15.2 Results available in spreadsheet format

The Amsterdam Modeling Suite can calculate thousands of different types of results. Only the most common ones are exported to spreadsheet format (see the below tables).

If you cannot find a result in the spreadsheet file, open the output file or use KFBrowser to inspect the binary .rkf results files.

15.2.1 Engines

ADF	✓
BAND	✓
COSMO-RS	✗
DFTB	✓
ForceField	✓
ReaxFF	✓
ML Potential	✓
Hybrid	✓
Quantum ESPRESSO	✗
VASP via AMS	✓ (partial)

15.2.2 Tasks

COSMO-RS Compound	✗
Conformers	✗
GCMC	✗
Geometry optimization	✓
IRC	✓
Molecular Dynamics	✓
NEGF	✗
Nudged elastic band (NEB)	✓
PES Exploration	✓
PES Scan	✓
Single point	✓
Transition state search	✓
Vibrational Analysis	✗

15.2.3 Single-point properties

These properties are also given for the final frame of a geometry optimization or molecular dynamics simulation, or for the highest-energy image in an NEB calculation.

Property	
Atomic charges	✓
Bader analysis, QTAIM	✗
Band gap	✓
Band structure	✗
Bonds and bond orders	✗
Bulk modulus	✓
Density of states (DOS)	✓
Diffusion coefficient	✗
Dipole moment	✓
Distance matrix	✓ (color-coded)
Eigenvalues and occupations per k-point	✗
Elastic tensor	✓

continues on next page

Table 15.1 – continued from previous page

Property	
Electron density at nuclei	✓ (ADF only)
Electrostatic potential at nuclei	✓ (ADF only)
Energy	✓
Enthalpy	✓
Entropy	✓
Excitations (UV/VIS)	✓ (types, energies, spectrum)
Fermi energy	✓
Gibbs free energy	✓
Heat capacity	✓
Hessian	✗
HOMO	✓
HOMO-LUMO gap	✓
Internal energy	✓
k-point coordinates	✓
LUMO	✓
MDC-d charges and spins	✓
MDC-m charges and spins	✓
MDC-q charges and spins	✓
Molecules	✗
Mulliken charges and spins	✓
NMR	✗
Normal modes (IR spectrum)	✓ (symmetries, frequencies, spectrum)
Nuclear gradients (forces)	✓
Orbitals	✓ (symmetries, energies, occupations)
Partial energies (Coulomb, ...)	✓
PES point character	✓
Partial DOS (PDOS)	✗
Phonon DOS	✓
Pressure	✓
Quadrupole moment	✓
Radial distribution function (RDF)	✗
Raman	✗
Shear modulus	✓
Statistical thermal analysis (thermodynamics)	✓
Stress tensor	✓
Velocity autocorrelation function	✗
Young's modulus	✓
Zero point energy (ZPE)	✓

15.2.4 Geometry optimization properties

Geometry optimization summary

Convergence	✓ (see Termination status)
Final maxGrad	✓
Final maxStressEnergyPerAtom	✓
Final rmsGrad	✓
Number of iterations	✓

Per-frame information: These are plotted vs. the frame number. For some of the properties, the plots only show data near the end of the optimization.

Energy	✓
Frame number	✓
Lattice parameters (a, b, c)	✓
maxGrad	✓
maxStressEnergyPerAtom	✓
Relative energy	✓
rmsGrad	✓

15.2.5 NEB properties

Left barrier	✓
Number of images	✓
Number of iterations	✓
Plot of energy vs. image	✓
Right barrier	✓

Per-frame information: See geometry optimization per-frame information.

15.2.6 MD properties

Timestep	✓
Number of steps	✓
Simulation time	✓
Start and end step	✓
Start and end time	✓

Per-frame information: These (except for Step) are plotted vs. time.

If the trajectory file contains more than 2000 structures, two worksheets are created:

- The first contains data for the first 2000 frames
- The second gives 2000 evenly spaced data points spanning the entire trajectory. Example: The trajectory contains 10000 frames, data is then given for frames 1, 6, 11, ..., 9991, 9996.

This is done to limit the size of the .xlsx file if the trajectory is very long.

Cell volume	✓
Conserved energy	✓
Kinetic energy	✓
Lattice parameters (a, b, c)	✓
Number of atoms	✓
Potential energy	✓
Pressure	✓
Step	✓
Temperature	✓
Time	✓
Total energy	✓

Block averages: The total trajectory is divided into 5 equally sized blocks, and for each block the mean and standard deviation of all per-frame quantities are reported. Currently you cannot change the number of blocks.

15.2.7 System

Atomic masses	✓
Atomic positions (xyz coordinates)	✓
Cell volume	✓
Charge	✓
Chemical formula	✓
Density	✓
Job name	✓
Lattice parameters (a, b, c, α , β , γ)	✓
Lattice vectors	✓
Net spin	✗
Number of atoms	✓
Mass	✓
Periodicity	✓
Picture	✓ ¹
Reciprocal lattice vectors	✗
Symmetry	✓ (only for ADF)
Velocities	✓

¹ A picture is generated if it is possible to start the AMS GUI.

15.2.8 General

AMS version	✓
CPU time	✓
Elapsed time	✓
Errors	✓
Names of compute nodes	✗
Number of compute nodes	✗
Start and end time	✗
Text input	✓
Termination status	✓
Warnings	✓

CONFORMERS

The GUI has an option to handle multiple conformers of molecules.

Conformers may be created using RDKit (see www.rdkit.org (<http://www.rdkit.org>)) via AMSinput. There is now a Conformers job method (just like ADF, BAND, DFTB etc) that will show a panel with options to generate the conformers. The algorithm used by RDKit for conformer generation will only work properly for single molecules, and the proper connectivity needs to be present.

After setting the proper options, just run the job like any other job. So it may also be running in the background.

The conformer geometries are stored in an .sdf file. This is in principle a series of xyz structures, with bond information and some other data. The energy is stored in the comment line, and in the extra fields there is also information like what program calculated the energy, and sometimes a pointer to the matching input file (.ams file) and results file (.t21 file).

You can use AMSmovie to open the .sdf file and see the molecules in there. Open the Energy Graph in AMSmovie to get a graph of the corresponding energies.

With AMSinput you can set up your calculation as usual (single point, geometry optimization, or anything else). In the Coordinates panel (in the Model section) you can specify that you want to use the molecules from the .sdf file (Use Selected File, and specify the .sdf file generated by the Conformers job). You can select what entries to use from the .sdf file using the SDF Filter panel (click on the ... button after Use Selected File, or go to the SDF Filters panel in the Details section directly). You can filter based on count or energy. In this panel you can also select what post-processing options will be applied to new sdf files generated by a calculation.

When you save your setup in AMSinput, a run script is created that will execute your job for all the conformers. Results of the individual calculations will be saved in the .results directory, and the original .sdf file will also be put there (with a time stamp). After running a new .sdf file will be created that has the results of the calculations, with pointers to the result files. Technically this is done using amsprep and amsreport in the run script.

Based on this new sdf file you can now do other things, if you want. Thus you can chain a series of calculations, for example first optimizing with DFTB or MOPAC, next optimizing with ADF, and finally calculating some property with ADF.

If you have calculated some spectrum (IR, NMR, UV/Vis) for all conformers in this way, you can visualize the averaged spectrum via AMSspectra. If it finds an .sdf file with the job, it will use it and the pointers in it to calculate averaged spectra. In AMSspectra there will be extra controls on how this averaging is done, most importantly the Temperature. In the View menu in AMSspectra you can also show all individual spectra, put together in one plot.

As the individual result files are present in the results directory, you can also visualize results from any specific conformer by selecting that t21 result file in AMSjobs, and then using the tools in the SCM menu as usual.

FUKUI FUNCTION

The Fukui function describes the electron density after adding or removing some amount of electrons. It can predict where the most electrophilic and nucleophilic sites on a molecule are.

Fukui functions can be easily calculated using the GUI, as shown in the [Fukui tutorial](#).

The Fukui function is implemented as a finite charge change. By default a whole electron is added or removed, however the calculation is not restricted to this amount. Fractional amounts of electrons are allowed.

The Fukui equation for the electrophilic attack, f^- , is given for when 1 electron is removed.

$$f^- = \rho(N) - \rho(N - 1)$$

Equally, the Fukui function for the nucleophilic attack, f^+ , is given for when 1 electron is added.

$$f^+ = \rho(N + 1) - \rho(N)$$

Do note that in all cases the geometry of the neutral state is used to get the electron density.

The Dual Descriptor is a way to combine the two Fukui functions. It has a positive value where it is electrophilic and negative where it is nucleophilic. It is implemented as the difference between the Fukui plus and Fukui minus functions.

$$f(r) = f^+ - f^-$$

The Fukui function can be used to describe local chemical reactivity. This can even be done per atom by using the condensed Fukui function.

$$f_k^- = q_k(N) - q_k(N - 1)$$

$$f_k^+ = q_k(N + 1) - q_k(N)$$

Atomic charges are used to get the condensed Fukui function. There are several ways to partition the atomic charges. The Fukui calculation prints the condensed Fukui functions for Hirshfeld, Mulliken, Voronoi, and if calculated, Bader charges.

The condensed Fukui function is normalized, so that atomic Fukui values sum to one.

$$\sum_{k=1}^{N_{atoms}} f_k(\mathbf{r}) = 1$$

The local softness can be calculated as the product of the condensed Fukui function and the global softness. An approximation for the global softness is given as the inverse of the HOMO-LUMO gap.

$$S = \frac{1}{E_{LUMO} - E_{HOMO}}$$

The local softness is then simply

$$s_k^-(\mathbf{r}) = S f_k^-(\mathbf{r})$$

$$s_k^+(\mathbf{r}) = S f_k^+(\mathbf{r})$$

MINIMUM ENERGY CROSSING POINT

The Minimum Energy Crossing Point (MECP) is the lowest point where two potential energy surfaces cross. These two potential energy surfaces can be any two orthogonal states. The situation for different spin states is implemented in the GUI, which takes the current state and the next spin state, e.g. singlet and triplet. This can be used to find the most likely location for a spin forbidden chemical reaction. SCM gratefully acknowledges Professor Jeremy Harvey for the initial MECP code.

The model menu in AMSinput has the option to calculate the MECP. The same convergence criteria as for ADF is used for the MECP optimization.

QUANTUM ESPRESSO

See the [Quantum ESPRESSO Engine manual](#) for AMS.

VASP VIA AMS

20.1 VASP support in the Amsterdam Modeling Suite

20.1.1 General

With AMS2019.3, the graphical user interface supports setting up, running and visualizing some results from calculations run using the [Vienna Ab initio Simulation Package, VASP](https://www.vasp.at) (<https://www.vasp.at>), version 5.X. VASP itself is *not* included in the Amsterdam Modeling Suite, but needs to be obtained and installed separately. As the VASP code has not been developed by SCM, we cannot give support for VASP issues other than use via our GUI or PLAMS.

20.1.2 VASP as an external engine

The GUI supports setting up calculations with VASP as an [external engine](#) to the [AMS driver](#), hence “VASP via AMS”. This means that the AMS driver handles all changes to the system’s geometry during for example a geometry optimization, NEB calculation, or molecular dynamics simulation. The energy and forces at each step are calculated by a single point VASP calculation. Efficient use of restart files minimize the resulting overhead.

VASP can also natively perform geometry optimizations, NEB calculations, and molecular dynamics simulations, but the VASP-via-AMS feature only supports those types of calculations via the AMS driver.

AMS driver tasks that change the number of atoms from one step to the next, for example Grand-Canonical Monte Carlo, Molecule Gun, or the numerical calculation of phonons (via a supercell), are currently not supported by the VASP-via-AMS interface.

20.1.3 Tutorials

There are [tutorials](#) that show how to use VASP via the GUI, and via the python scripting toolkit PLAMS.

20.2 Setting up a VASP calculation

20.2.1 VASP executable and number of processes (parallelization)

There needs to be a working VASP installation on the machine where the calculation is run. Specify the exact command you would like to execute to run VASP as the **Command to execute VASP** on the main VASP page in AMSinput. For example, set it to `mpirun -np 16 vasp`, if you use mpirun to launch VASP and would like to parallelize over 16 processes.

Note: The environment variable NSCM must be set to 1. This is automatically done by AMSinput.

20.2.2 Pseudopotentials and PAW potentials (POTCAR)

VASP requires that pseudopotentials or PAW potentials be used for each element. Such potentials are distributed with VASP in files called POTCAR or POTCAR.Z. For example, the following files might exist for PAW potentials intended for use with the PBE density functional:

```
/some/path/PBE_PAW/H/POTCAR  
/some/path/PBE_PAW/O/POTCAR  
/some/path/PBE_PAW/O_h/POTCAR  
/some/path/PBE_PAW/O_s/POTCAR
```

In this case, `/some/path/PBE_PAW` would be a **POTCAR Library**, that can be entered on the main VASP in AMSinput. That would by default use the `/some/path/PBE_PAW/H/POTCAR` and `/some/path/PBE_PAW/O/POTCAR` files for H and O, respectively. To specify other POTCAR files, for example the harder (O_h) or softer (O_s) PAW potentials for O, select those files on the **Details → Pseudopotentials** page.

If you set a POTCAR library, it will be remembered for the next time you launch AMSinput.

Note: The POTCAR library is a path on the machine running AMSinput, which need not be the same machine as where the calculation is eventually run.

Tip: If as an example you frequently use the `O_h/POTCAR` file for O atoms, consider creating a custom POTCAR library where this file is instead moved to `O/POTCAR`, so that it is automatically selected if you specify the custom POTCAR library.

20.2.3 k-point sampling (KPOINTS)

You can specify the grid sizes for Monkhorst-Pack or Gamma-centered k-point grids, together with fractional k-point coordinate displacements, on the main page in AMSinput. If you would like to use a custom KPOINTS file (to be read by VASP), select your file under **Details → Expert VASP → KPOINTS File**. That file will override any other specified k-point settings.

Currently, to specify a path through the Brillouin zone for band structure calculations, or to use some manually specified set of k-points, you must generate such a KPOINTS file yourself.

20.2.4 Other settings (INCAR)

Many calculation settings can be set using the GUI, for example,

- The planewave energy cutoff (**main panel**)
- The exchange-correlation functional (**main panel**)
- Fractional occupation scheme (smearing) (**main panel**)
- van der Waals corrections (**Model → Van der Waals correction**)
- Hubbard U parameters for different elements (**Model → Hubbard U**)
- WAVECAR or CHGCAR files for restarting or initializing a calculation (**Details → Expert VASP**)

Moreover, you can set arbitrary settings as “Additional INCAR options” on the **Details → Expert VASP** page. In case of any conflict between these and other INCAR-related settings set in the GUI, the “Additional INCAR options” take precedence.

You can also use your own INCAR file (**Details → Expert VASP**). That will ignore all INCAR-related settings in the GUI.

Note: the INCAR tags NSW, IBRION, ISIF, and LWAVE are automatically set by the AMS driver before VASP is executed.

20.3 Running VASP via AMS

20.3.1 Running VASP with AMSjobs

You can run a VASP-via-AMS job just like any other job using [AMSjobs](#). If a geometry optimization, molecular dynamics simulation, or NEB calculation was performed, you can visualize the results in AMSmovie.

Note that the AMS driver enforces that LWAVE be set to `.TRUE.`, thus leaving the WAVECAR file on disk. This file can become rather large, so if you do not need it for post-analysis, or for initializing another calculation, you could consider removing this file after the calculation has completely finished.

20.3.2 Preprocessing the input

You can also choose to only do the necessary preprocessing of a job before VASP is executed, by selecting “Only preprocess” on the **Details → Expert VASP** page. By running such a job, the INCAR, POSCAR, POTCAR, KPOINTS, and any other auxiliary input files will be left on disk in the folder `jobname.results/external`, but VASP will not be executed. It may be useful to manually inspect the contents of these files, to see exactly what the input to VASP is. For the preprocessing step, no working installation of VASP is needed.

20.3.3 Scripting support

Using VASP-via-AMS via the Python scripting framework PLAMS is fully supported, see the [tutorial](#).

20.4 VASP via AMS without the GUI

This section describes how to set up a VASP-via-AMS calculation without the GUI. Please note that the keyword names, and the syntax for setting up a VASP-via-AMS calculation, may change in future versions of the Amsterdam Modeling Suite. For this reason, we recommend setting up the calculation via the GUI.

Before `ams` is executed, the pertinent POTCAR (as read by VASP) must be created in `$AMS_RESULTSDIR`.

VASP runs as an external engine to the AMS driver. The engine block to the AMS input file should read:

```
Engine External
  InputDefinition $AMSBIN/input_def/vasp.json
  Execute "$AMSBIN"/amspython "$AMSHOME"/scripting/standalone/external_engines/vasp.
  ↳py
    Input
      ...
    EndInput
  EndEngine
```

where \$AMSHOME and \$AMSBIN are replaced by their respective values, and where the allowed settings in the Input block are the following:

CHGCARFile**Type**

String

GUI name

CHGCAR file

Description

Use a custom CHGCAR file for initializing the charge density. Note: You must also set ICHARG to an appropriate value for this CHGCAR file to be read by VASP.

EnergyChoice**Type**

Multiple Choice

Default value

FreeEnergy

Options

[FreeEnergy, EnergyWithoutEntropy, EnergySigmaToZero]

GUI name

Energy from OUTCAR

Description

The energy from the VASP output which is used with the AMS driver.

EnergyCutoff**Type**

Float

Default value

400.0

Unit

eV

GUI name

Planewave energy cutoff

Description

The energy cutoff for the planewave basis set.

VASP keyword: ENCUT

INCARFile**Type**

String

GUI name

INCAR file

Description

Use a custom INCAR file.

The keywords NSW, IBRION, ISIF, and LWAV must not be set in this INCAR file, since they are controlled by the AMS driver.

Initialize**Type**

String

GUI name

WAVECAR file

Description

Path to a WAVECAR (wavefunction) file from a previous calculation that is used to initialize the present calculation.

If specified, the file will be copied into the new calculation directory and the VASP keyword ISTART will be set to 1.

KPOINTSFile**Type**

String

GUI name

KPOINTS file

Description

Use a custom KPOINTS file. This will override any other settings related to the k-point sampling.

KPOINTSOrigin**Type**

Multiple Choice

Default value

Monkhorst-Pack

Options

[Gamma-centered, Monkhorst-Pack]

GUI name

Type of k-point grid

Description

Type of regular k-point grid. Gamma-centered k-point grids always include the Gamma point, whereas Monkhorst-Pack grids do not include the Gamma point if any of the k-point grid dimensions is even.

To use a custom set of k-points, or to specify paths in the Brillouin zone, prepare such a KPOINTS file manually and set the KPOINTSFile setting.

LDAU**Type**

Bool

Default value

No

GUI name

Use DFT+U

Description

Enables a DFT+U calculation. The options LMAXMIX, LDAUType, LDAUU, LDAUJ, and LDAUL must also be set.

LDAUJ

Type
Float List

Description

LDAUL

Type
String

Description

LDAUType

Type
Multiple Choice

Default value
2

Options
[1, 2, 4]

GUI name
Hubbard U-Type

Description
1: The rotationally invariant DFT+U introduced by Liechtenstein et al.
2: The simplified (rotationally invariant) approach to DFT+U, introduced by Dudarev et al.
4: Same as type 1, but without exchange splitting.
VASP keyword: LDAUTYPE.

LDAUU

Type
Float List

Description

LMAXMIX

Type
Integer

Default value
2

GUI name
LMAXMIX

Description
VASP keyword: LMAXMIX

Misc

Type
Non-standard block

GUI name
Additional INCAR options

Description
The contents of this block are copied verbatim into the VASP INCAR file.

Note: do not set NSW, IBRION, ISIF or LWAV in this block, as those keys are set by the AMS driver.

nk1

Type

Integer

Default value

1

Description

Number of k-points in the first dimension

nk2

Type

Integer

Default value

1

Description

Number of k-points in the second dimension

nk3

Type

Integer

Default value

1

Description

Number of k-points in the third dimension

Occupation

Type

Multiple Choice

Default value

Gaussian

Options

[Gaussian, Fermi, 1stOrderMethfesselPaxton, 2ndOrderMethfesselPaxton, TetrahedronMethod-WithoutBlochlCorrections, TetrahedronMethodWithBlochlCorrections]

GUI name

Type of smearing

Description

Type of electronic smearing (fractional occupation scheme).

VASP keyword: ISMEAR

OnlyPreprocessing

Type

Bool

Default value

No

GUI name

Only preprocess

Description

If Yes, VASP will not be executed. Instead, the first set of input files (e.g. INCAR, POSCAR, POTCAR, KPOINTS) that would have been read by VASP will be left on disk for inspection.

POTCARLibrary

Type

String

GUI name

Path to POTCAR library

Description

Path to the POTCAR (pseudopotential) library, as delivered with VASP. This library must be accessible on the machine currently running AMSinput.

For example, setting this to /library/PBE/PAW/ will use /library/PBE/PAW/Cu/POTCAR as the POTCAR file for Cu atoms by default. Click the arrow to see or change exactly which POTCAR files are used.

Precision

Type

Multiple Choice

Default value

Normal

Options

[Single, Normal, Accurate]

GUI name

Precision

Description

VASP keyword: PREC

sk1

Type

Float

Default value

0.0

Description

Fractional shift of the k-mesh in the first dimension

sk2

Type

Float

Default value

0.0

Description

Fractional shift of the k-mesh in the second dimension

sk3

Type

Float

Default value

0.0

Description

Fractional shift of the k-mesh in the third dimension

Smearing**Type**

Float

Default value

0.2

Unit

eV

GUI name

Smearing width

Description

VASP keyword: SIGMA

SpinPolarization**Type**

Bool

Default value

No

GUI name

Spin-polarization

Description

Determines if a spin-polarized calculation is performed. If set, the VASP keyword ISPIN is set to 2.

VASPExec**Type**

String

Default value**GUI name**

Command to execute VASP

Description

Command used to execute VASP.

For example: mpirun -np 4 vasp

vdW**Type**

Multiple Choice

Default value

Disable

Options

[Disable, DFT-D2, DFT-D3, DFT-D3(BJ), TS, TSWithIterativeHirshfeldPartitioning, Many-BodyDispersionEnergyMethod, IDVW=4]

GUI name

van der Waals method

Description

The van der Waals correction method.

VASP keyword: IVDW

Default: IVDW = 0 (Disable)

Verbosity

Type

Bool

Default value

No

GUI name

Verbosity

Description

If set, more detailed information is printed in the output file.

XC

Type

Multiple Choice

Default value

Auto

Options

[Auto, LDA, PBE, PW91, HF, PBE0, HSE06, B3LYP, 03, 05, 10, 20, 91, AM, B3, B5, BF, BO, CA, CO, HL, MK, OR, PE, PL, PS, PZ, RA, RE, RP, VW, WI]

GUI name

XC functional

Description

The exchange-correlation functional.

This setting **must** be consistent with POTCAR. If your desired functional is not present in the list, choose Auto and set the GGA tag manually under Additional INCAR options.

Note: not all functionals are implemented in all versions of VASP.

Auto: The XC functional is determined by the information in POTCAR. The GGA setting is not set.

LDA: the VASP GGA setting is set to an empty string.

PBE: sets GGA = PE.

PW91: sets GGA = 91.

HF: Hartree-Fock. Sets LHFCALC = .TRUE, AEXX = 1, ALDAC = 0, AGGAC = 0.

PBE0: sets GGA = PE, LHFCALC = .TRUE.

HSE06: sets GGA = PE, LHFCALC = .TRUE., HFSCREEN = 0.2.

B3LYP: sets LHFCALC = .TRUE., GGA = B3, AEXX = 0.2, AGGAX = 0.72, AGGAC = 0.81, ALDAC = 0.19.

All other options set the VASP GGA setting to the corresponding value. For more information, see the VASP manual.

PARAMS

21.1 General

21.1.1 Start ParAMS

- Open AMSjobs, and select **SCM → ParAMS**
- From the command-line, run `$AMSBIN/params -gui`. To open a file, run for example `$AMSBIN/params -gui job_collection.yaml`.

By default ParAMS opens a Lennard-Jones project. Switch to ReaxFF, GFN1-xTB, ASE and No Parameters in the **Parameters** menu.

21.1.2 Overview

The ParAMS GUI lets you set up training and validation sets, run ReaxFF and DFTB parametrizations, and view the results. It is a GUI for the [ParAMS Python library](#).

Get started with the ParAMS GUI: See the [tutorials](#).

The ParAMS GUI shows you the

- **Jobs**, containing structures and settings for the jobs to be run during the parametrization
- **Training set**, defining the *loss function* to be minimized
- **Validation set**, an optional data set that can be used to evaluate over/underfitting
- **Engines**, a collection of settings that can be used for **(a)** running reference jobs, **(b)** viewing reference job settings, or **(c)** setting the engine settings for the jobs during the parametrization (augments the AMS settings defined in the *Jobs*).
- **Parameters**, a table of parameters with allowed ranges for the parameters
- **Settings**, settings for the parametrization (e.g. which optimizer to use)
- **Info**, detailed information about *jobs* and *training set entries*. (read-only)
- **Graphs**, graphs with details about the parametrization results
- **Results**, text output files with details about the parametrization results (read-only)

When saving, you will enter the name of a directory. The directory name must end with `.params`, otherwise the extension will be added. In the directory, the saved files are called

- `job_collection.yaml`
- `training_set.yaml`

- validation_set.yaml
- engines.yaml (ParAMS also sometimes uses the name job_collection_engines.yaml)
- parameters.yaml (ParAMS also sometimes uses the name parameter_interface.yaml)

21.1.3 File handling

Open

Use the Open command in the File menu to open a specific file. Alternatively, one can start `$AMSBIN/params -gui` via the command line and specify the file to open as argument.

The current setup will be cleared (that is, the New command is automatically used).

ParAMS will try to determine what kind of file you have selected to open. It will also read all related files in the same directory having any of the file names listed above.

Merge

When using the Merge command in the File menu, the selected file will be added to the current ParAMS window. Only the selected file will be imported.

Job names and training/validation sets need to be unique. If you import jobs with already existing names, then the new jobs will be renamed by adding a suffix `_1`, `_2` etc.

Save

First it will be suggested to save with the same names as has been used while opening.

If you do not like that (probably you want to keep your original files), click 'No'. You will be asked to select a directory in which to save the files. All files will have their default names.

Revert

If you have unsaved changes, you can revert to the last save point using File -> Revert. There is no other Undo functionality.

Note that for running parametrizations, the table of parameters is updated on-the-fly.

21.1.4 Tables

The Jobs, Training Set, Validation Set, Engines, and Parameters are shown as tables.

Header menus: sort and filter

On top of the tables is an header. All the items in the header are pull-down menus, you can use them to sort the table (by the values in the corresponding column).

For some headers you can also filter by specific types. You can select which types of data to show, one or more types, or all.

Search

Both tables have a Search field on the bottom. Use cmd/ctrl-F to go there, or click in the field. While you type, the data shown in the corresponding table will automatically be filtered by whatever you type. The spaces in your search string will automatically be replaced by wild-cards.

Selection

To make selections in the tables use your mouse.

Left click: set the selection to the clicked line,

Shift-left click: extend the selection up to and including the clicked line,

Control-left or right click: toggle the clicked line in/out the selection.

The backspace key: delete the selected lines.

The Escape key: clear the selection.

All lines in the Training Set correspond to one or more systems (molecules). If only one system is selected (or multiple lines with the same system) it is displayed on the left. When a single line with an Energy term is selected, all systems in that energy expression are shown on the left. Use your arrow keys (left and right) to go through the different systems in the energy expression.

You can move the selected line in the tables using up and down arrow keys.

21.2 Import reference calculations

The easiest way to import reference results is to either

- Select the finished calculation in AMSjobs and choose **Job → Add to ParAMS**
- In the ParAMS GUI, choose **Jobs → Add Jobs** and browse to the **ams.rkf** file from an AMS calculation, the **OUTCAR** file from a VASP calculation, or the **.out** file from a Quantum ESPRESSO calculation

This will bring up a results importer dialog, where you can choose for example which frames to import and which properties to extract (energy, forces, charges, ...).

In this way, you will add entries to the Jobs, Training Set, and Engines.

For details, see the [ParAMS documentation](#).

You may also import structures from AMSinput or AMSmovie:

- AMSinput: use the 'Add To ParAMS' menu command (in the File menu). This also copies the calculation settings to the added Job (for example, maximum number of geometry optimization iterations).

- AMSmovie: use the 'Add To ParAMS' menu command (in the File menu) to add the current frame. You can do this repeatedly.

If you import from AMSinput or AMSmovie, you will manually need to add the training set entries and reference values.

21.2.1 Jobs

A job is a molecule or a crystal, together with details on how to run a calculation with it. So it might be a single point calculation, a geometry optimization, with extra details (like constraints, maximum number of geometry iterations, ...).

If you select a Job, and if it is the only selected system, the molecule will be shown on the left.

You can use the commands in the View menu to change details of what is shown on the left, just as in other AMS-GUI modules. Use the mouse to rotate, zoom, and select atoms.

In a Job line in the table you can see the details of the system:

- Job ID,
- Details
- Associated engines

If you double-click in the Details column, you will be able to modify the job details:

- Modify the AMS settings, or set them via AMSinput (click the AMSinput button)
- Modify the reference engine (expert option, use only if you want to use ParAMS to calculate new reference values)
- Modify the ParAMS engine
- Set the system charge
- Some metadata is shown (read-only)

When saving your setup, the jobs are saved to `job_collection.yaml`.

To make renaming your SystemIDs easier, there are menu commands:

- Jobs → Prefix JobIDs - apply a prefix to all selected job ids
- Jobs → Change JobIDs - search and replace in the selected job bids. You will be asked for a string to search for, and for a string to replace it with.

21.2.2 Training set

A training set also defines the objective function that will be minimized with the parameter optimization. The objective function consists of many terms, each line in the table is one term.

To add terms to your training set, use a Results Importer in most cases (energy, forces, charges, stress tensor). For geometric data (bond lengths, angles, cell parameters) use the Add menu. Typically the selected systems and the selected atoms are used when adding terms.

As an example, to add a distance terms to your training set:

1. select the system for which you want to add a distance term,
2. select two atoms in the system,
3. use the Bonds command from the Training Set → Add menu.

One distance term will be created in your training set. When you are going to use a distance term your system will need to be a geometry optimization. If it would be a single point calculation, the distances will always remain the same so this term would just be a constant.

The atom selection determines what distance terms are added:

- If no atoms are selected (thus step 2 is skipped), all bonds found in the system will be added.
- If two atoms are selected (as in the example): the distance between those atoms will be added (even if they are not bonded directly).
- If more than two atoms are selected: all selected bonds in the system will be added (thus existing bonds between selected atoms).

If more than one system is selected, the above is repeated for all selected systems. The atom selection criterion will still be used, note that this makes no sense if the systems do not have the same atoms in the same order. This way of handling the atom selection is used for all geometric tests.

In the distance term line you can see:

- the Type ('Geometry: distance'),
- the job id (some name, identical to the name of the selected job),
- the Weight (W) (to set the relative contribution of this term to the objective function),
- the Details (the atom numbers and atom types involved in this particular distance),
- the Value (the reference value, calculated using the current system),
- the Prediction (when opening a training set after optimizing, the predicted value of this term).
- the Loss Contribution (when opening a training set after optimizing, the actual contribution to the objective function of this term).

The Loss Contribution column will be filled with results from the training (read from a `data_set_predictions.yaml` file). It shows the contribution to the objective function for each term in the training set. It is especially convenient if you sort by this column.

Term types that can be added are evident from the Add menu:

- distances (Bonds)
- angles
- dihedrals (Torsions)
- several types of charges
- energy
- force
- cell parameters

When saving your setup, the training set is saved in a file called 'training_set.yaml'.

If you double-click in the Details for a training set entry, you can edit

- The expression (including tab completion for job entries)
- Sigma
- Weight
- The reference value

If the entry is of type **Energy**, you can also use the **Balance** function to automatically balance a reaction energy. See the [ParAMS tutorials](#) for details.

21.2.3 Validation Set

The Validation Set works just like the Training Set.

Training Set → Generate Validation Set lets you move a random subset of the selected training set entries to the validation set.

Training Set → Move To Validation Set moves the selected training set entries to the validation set.

Training Set → Move To Training Set moves the selected validation set entries to the training set.

21.2.4 Mouse interaction

In addition to the general mouse interactions for selection (already mentioned):

- Double click item to edit
- Right click to search for jobids
- Mouse over to see details, error info etc (you will get a popup if you do not move the mouse).

21.2.5 Editing

To edit something, double click on it (or single click on a selected item). Next you can edit it in place, or in a window that appears.

Depending on what you are editing, the changes you make will be applied to the full selection! So when selecting multiple terms and editing the Weight, that field will be set for all your selected items. Similar, when changing runtime options, like runtime or maximum number of iterations, it applies to all selected systems. Constraints are not propagated as they depend on a particular system.

When editing Job IDs, you can use the tab key to cycle through all Job IDs matching what you have been typing. Add a space to use the current suggestion, or type more letters to narrow the search down. This works when changing Job IDs as well as when editing energy expressions.

21.2.6 Warnings and errors

If you hover with your mouse over a red line you will get extra information on why it is red, what kind of error.

For example, checking a distance for a system which is not optimized is useless, or having systems that are not used in any term. For energy expressions there is a check that the stoichiometry is correct (thus the net amount of all atom types should be zero).

Duplicate lines will also be marked as warning/error.

21.2.7 Reference data

Running reference jobs

If some training set entries do not have reference data, the corresponding reference calculation will be run before the parametrization.

To run, use AMSjobs.

Importing results from reference jobs

Use the ‘Get Data From Ref Jobs’ menu command from the ‘Training Set’ menu to update all selected items in the training set with data from the reference jobs (if they are available). If you have no selection, all items in the training set will be updated.

21.3 Parameters

For a description of the parameters, please check the [parameter interface documentation](#).

21.3.1 ReaxFF parameters

IgDispersion, ACKS2 or eReaxFF

The ParAMS GUI currently does not support IgDispersion parametrization.

For ACKS2/eReaxFF parametrization, you **must** set [ereaxff acks2] as the beginning of the header (Parameters → Edit ReaxFF header). It is not enough to just activate the parameters.

Add a ReaxFF block of parameters

ReaxFF parameters come in **blocks**: GEN, ATM, BND, ANG, OFD, TOR, and HBD. There can be only 1 GEN block.

To add a block: **Parameters → Add ReaxFF block**, and type e.g. ANG : H . C . H to add a block of parameters describing the angular H-C-H interaction.

Initialize ReaxFF parameters

You can search through all force fields that come with AMS for a given block of parameters, and copy them into your current project.

- Select one or more parameters in the bottom table.
- Parameters → Initialize ReaxFF block...
- Select which blocks to search for

Edit ReaxFF header

The first line in the `ffield.ff` file is a header. You can change it with Parameters → Edit ReaxFF header.

Editing parameters and optimization details

Each parameter needs

- an (initial) value
- the minimum and maximum allowed values
- to be either **Active** (optimized) or **Not Active** (not optimized)

Select one or more parameters using Left Click and Shift Left Click. The Shift Left Click action will extend the set of selected parameters with all parameters inside a rectangle with respect to the first selected parameter. Click on a selected parameter to edit these values (thus if no selection double click on a parameter).

The parameters are saved to `parameters.yaml`.

21.3.2 ASE Parameters

The GUI cannot be used to create ASE parameters. You need to first create the `parameter_interface.yaml` file and then load that. The loaded ASE parameters can be modified as usual. It is easiest to create the file with a Python script. See the [ASE Calculator parametrization](#) tutorial.

21.3.3 No Parameters

There no parameters to modify in the GUI.

21.4 Graphs

On the Graphs panel, you can plot three different graphs simultaneously. Each graph has three drop-down menus:

- Left: The type of graph (running loss, stats, parameter values, PES scan, correlation plot)
- Middle: Options (e.g. switch between MAE and RMSE, or select Best Training/Best Validation etc.)
- Right: Data From. Choose to compare to the initial parameters

All graphs show the output from text files that already exist in the parametrization results directory. You can also save the xy-values using File -> Export Graph as XY, or save the graph image with File -> Save Graph as Image

21.5 Settings

On the Settings panel, you can add settings from the yellow drop-down. Click the minus button in front of a setting to remove it (to use the default value).

For details about what the various options can be set to, see the [ParAMS documentation](#).

21.6 Results

The Results tab lets you choose from various text output files. All the files exist on disk already.

21.7 Info

The Info panel shows details about a selected Job, Training Set/Validation Set entry, or Engine.

AMSCRS: COSMO-RS

AMScrs is a utility program (\$AMSBIN/amscrs), which enables AMS users to easily create COSMO-RS jobs. You can use AMScrs to add compounds, choose the desired property, and to set details of your COSMO-RS job using an easy-to-use graphical user interface. AMScrs will generate the complete job script for you. This script takes care of running COSMO-RS. You can also use AMScrs to run these script files and visualize the results.

The description of a compound that you want to use should be on a file, and should be a result file of quantum mechanical calculations using COSMO. In ADF such a COSMO result file is called a `adf.rkf` (.rkf) file, or a `COSKF` (.coskf) file. AMScrs might also be able to read a result file, for example a `.cosmo` file, from other programs.

If you have installed the AMS package correctly, the `amscrs` command is located in \$AMSBIN.

If \$AMSBIN is included in your PATH environment variable, you can start the AMScrs program with the following command:

```
amscrs [filename filename2 ...]
```

The file names are optional. AMSinput handles files that were created by AMSinput before (which have a `.crs` or `.crskf` extension), use only one file name in these cases. One can add multiple files, which correspond to different compounds, if these files contain results of quantum mechanical calculations including COSMO on these compounds. these result files should have `.rkf`, `.t21`, `.coskf`, or `.cosmo` extension. The file can also be a plain text file with a list of file names (`.compoundlist`), which contains on each line the filename of a `.rkf`, `.t21`, `.coskf`, or `.cosmo` file.

An alternative method to start AMScrs: select the COSMO-RS command from the SCM menu, or use AMSjobs to start AMScrs.

Under windows you can start AMScrs by double-clicking the icon on the desktop.

22.1 Menu Commands

22.1.1 File menu

New

Same as quitting AMScrs and starting again without specifying a file name.

Open . . .

Open an existing AMScrs file (with `.crs`) or an AMScrs result file (with `.crskf`).

Same as quitting AMScrs and starting again with specifying a file name. When you open a `.crs` file AMScrs also tries to add all compounds that are specified in this `.crs` file and tries to open all `.crskf` files that belong to this `.crs` file. If you open a `.crskf` file only this file is read.

Save

Save the current state of what is present in AMScrs. If you have not saved before, AMScrs will ask you to specify a file name.

A run file will not be saved.

Save As...

Save the current state of what is present in AMScrs in a file with a name of your choice.

A run file will not be saved.

Save Runfile

Not only the .crs file will be saved, but also a matching .run file which is a run script corresponding to your input (for the selected property).

Run

Start the COSMO-RS calculation as selected in all the input options.

You will first be asked to save the changes. The run script (with the .run extension) for the selected property is created. Next your job is run. When the run is finished the results will be visualized by AMScrs and the .run file will be deleted.

Kill

Kill the COSMO-RS calculation if one is running. The .run file will be deleted.

Close

Stop AMScrs, ask you to save changes if you made any.

22.1.2 Compounds menu

List of Added Compounds

Shows a list of all the added compounds, including a possibility to set some input parameters, specific for a certain compound.

On the left side of the window the list of the compounds is given with the full path name to the file from which the quantum mechanical results data are read. Part of this quantum mechanical data is given in the right window for the selected compound. In this right window one can also write some pure compound input data. For ring compounds it is important to write the number of ring atoms. For example, this number should be 6 for benzene.

Add Compound(s)

Use this menu command to add one or more compounds to the list of compounds. The selected files should be of type:

- .compoundlist: List of compounds
- .coskf: COSMO kf file
- .rkf .t21: ADF result file
- .compkf: COMPKF file
- .cosmo: COSMO file

The added file(s) should be either ADF result files (.rkf or in previous versions .t21), COSMO kf files (.coskf), COMPKF kf files (.compkf), or ASCII COSMO files (.cosmo). These files should be result files of a quantum mechanical calculation including COSMO, which contains COSMO segment data, or a result of the Fas Sigma prediction (.compkf). In case of ADF the ADF result file (.rkf, .t21 or .coskf) will contain COSMO data if the ADF calculation was done with COSMO. The added file can also be a plain text file with a list of file names (.compoundlist), which contains on each line the filename of a .rkf, .t21, .coskf, or .cosmo file.

Compound with multiple forms

Use this menu command to make a compound with multiple forms, like conformer(s), dimer(s), trimer(s), dissociated form(s), or associated form(s).

Install COSMO-RS database...

Download and install the COSMO-RS database ADFCRS-2018.

Select Compound(s) from COSMO-RS database

If the COSMO-RS database was downloaded, it is easy to select with this menu compounds from this database.

22.1.3 Method menu

The methods and its parameters are described in the [COSMO-RS manual](#).

COSMO-RS

Select this menu command to set the method that will be used in the calculation to COSMO-RS.

COSMO-SAC

Select this menu command to set the method that will be used in the calculation to COSMO-SAC.

UNIFAC

Select this menu command to set the method that will be used in the calculation to UNIFAC.

Parameters

Change the COSMO-RS or COSMO-SAC parameters and technical parameters that are used in the calculation. The COSMO-RS and COSMO-SAC model have general and element specific parameters. Technical parameters are, for example, convergence criteria and thresholds.

22.1.4 Properties menu

How the properties are calculated and definitions used can be found in the [COSMO-RS manual](#).

Vapor Pressure Pure Solvents Vapor Pressure Mixture

In case of a mixture, the mixture can be composed of up to five compounds, and the mole fraction of each compound of the solvent should be given, and it should add up to 1.

It is possible to calculate the vapor pressure for a temperature range, if the first temperature (from:) is different than the last temperature (to:).

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will be approximated using the COSMO-RS method.

Boiling Point Pure Solvents Boiling Point Mixture

In case of a mixture, the mixture can be composed of up to five compounds, and the mole fraction of each compound of the solvent should be given, and it should add up to 1.

It is possible to calculate the boiling point for a pressure range, if the first pressure (from:) is different than the last pressure (to:).

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will be approximated using the COSMO-RS method.

Solvent Flash Point

The mixture can be composed of up to five compounds. The mole fraction of each compound of the solvent should be given, and it should add up to 1.

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will

be approximated using the COSMO-RS method. However, for the compounds in the solvent the flash points of the pure compounds should be given as input, since COSMO-RS does not calculate these.

Activity coefficients

The solvent can be a mixture of up to five compounds. The mole fraction of each compound of the solvent should be given, and it should add up to 1. One can use an input value for the density of the solvent instead of a calculated value, which can influence the calculated Henry constants.

The activity coefficients will be calculated for the selected compounds in the listbox, on the bottom-left side of the window. Compounds not present in the solvent are considered to be infinitely dilute.

Log partition coefficients

The log of the partition coefficient (logP) of a solute in 2 phases of a solvent. Both phase 1 and phase 2 can be a mixture of up to three compounds. The mole fraction of each compound in phase 1 and phase 2 of the solvent should be given, and for both phases it should add up to 1. For example, if one wants to calculate partition coefficients for Benzene/Water, which are 2 immiscible liquids, phase 1 of the solvent will be purely Benzene, and phase 2 will be purely Water. Another example is the calculation of partition coefficients for 1-Octanol/Water, which are partly miscible liquids, phase 1 of the solvent will be a mixture of 1-Octanol and Water, and phase 2 will be purely Water (the solubility of 1-Octanol in Water is small).

One can use an input value for the quotient of the molar volumes of phase 1 and phase 2 instead of calculated values.

The logP will be calculated for the selected compounds in the listbox, on the bottom-left side of the window. Compounds not present in the solvent are considered to be infinitely dilute.

Solubility in Pure Solvents Solubility in Mixture

In case of pure solvents one can select any number of different pure solvents, but only 1 solute.

In case of a mixture, the mixture can be composed of up to five compounds, and the mole fraction of each compound of the solvent should be given, and it should add up to 1. In this case can select any number of different solutes, but they should not be present in the solvent. In case of a mixture one can also use an input value for density of the solvent.

It is possible to calculate the vapor pressure for a temperature range, if the first temperature (from:) is different than the last temperature (to:).

For the solubility of a gas in a solvent one has to set the vapor pressure of the gas.

For the solubility of a solid compound it is necessary to include the melting point, the enthalpy of fusion, and optionally, since it is often not so important, the Δ heat capacity of fusion of the pure compound (default Δ heat capacity of fusion is zero). These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window.

Binary mixture VLE/LLE

Exactly two compounds should be selected. One can select an isothermal, an isobaric, or a flash point calculation of the binary mixture. The binary mixture will be calculated for a list of molar fractions between zero and one.

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will be approximated using the COSMO-RS method.

Ternary mixture VLE/LLE

Exactly three compounds should be selected. One can select an isothermal, an isobaric, or a flash point calculation of the ternary mixture. The ternary mixture will be calculated for a list of compositions.

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will be approximated using the COSMO-RS method.

Solvents s1 - s2 Composition Line

Linear interpolation between the compositions of solvent 1 and solvent 2, which both could be mixtures. The solvents s1 and s2 can be a mixture of up to five compounds. The mole fraction of each compound of the solvent s1 and s2 should be given, and it should add up to 1. One can select an isothermal, an isobaric, or a flash point calculation.

One can use input values for the vapor pressure of the pure compounds at a given temperature. These input values can be set by selecting **Compounds → List of Added Compounds**, select the desired compound, and set input values for the pure compound in the right window. If these values are not specified (if they are zero) then they will be approximated using the COSMO-RS method.

22.1.5 Solvent Optimizations menu**Optimize Solubility**

Optimize the solvent (mixture) in order to maximize or minimize the mole fraction solubility of a solid solute in the liquid mixture.

Optimize Liquid-Liquid Extraction

Optimize a mixture of immiscible solvents in order to maximize or minimize the distribution ratio (D) of two solutes between the two liquid phases.

22.1.6 Analysis menu**Sigma Profile Pure Solvents Sigma Profile Mixture**

The σ -profile is calculated for COSMO charge densities between minus the maximum value for sigma and the maximum value for sigma. In case of pure solvents one can select any number of different pure solvents. In case of a mixture, the mixture can be composed of up to five compounds.

Sigma Potential Pure Solvents Sigma Potential Mixture

The σ -potential is calculated for COSMO charge densities between minus the maximum value for sigma and the maximum value for sigma. In case of a mixture, the mixture can be composed of up to five compounds.

22.1.7 Graph menu**Graph X Axes**

Use one of the submenu commands to change the X axes of your graph.

Graph Y Axes

Use one of the submenu commands to change the Y axes of your graph.

Graph Z Colormap

Use one of the sub-menu commands to change the colormap of your graph. Only available for **Properties → Ternary Mixture**.

Save As PostScript

If you have any graphs, you can select one of the graphs. It will be saved in a postscript file.

Save As XY

If you have any graphs, you can select one of the graphs. It will be saved in a text file as XY pairs. Next you can use most other plotting programs to make the graph just as you want it to be.

22.1.8 Help Menu

The help menu provides an easy way to get to information about the AMS-GUI. It will start a browser on your local machine, and opens a local copy of the documentation on the SCM web site.

AMSKINETICS (MKMCXX AND ZACROS)

The AMSkinetics GUI module provides an graphical interface to set up *microkinetics* (page 123) or *kinetic Monte Carlo* (page 123) (KMC) simulations.

23.1 MKMCXX Microkinetics

GUI Tutorial: *Microkinetics: CO oxidation*

Microkinetics simulations can be done via the **MKMCXX** (<https://www.mkmcxx.nl/>) program by **Filot, I.A.W., Zijlstra, B. and Hensen, E.J.M.** The AMSkinetics GUI module was initially developed by **Micha Davids**. For a description of the **MKMCXX** program and an overview of the microkinetics method, we refer the reader to the following pages:

- **MKMCXX website** (<https://www.mkmcxx.nl/>)
- **MKMCXX wiki** (https://wiki.mkmcxx.nl/index.php/Main_Page)
- **Introduction to microkinetic modeling** (<https://research.tue.nl/en/publications/introduction-to-microkinetic-modeling>) by I.A.W. Filot

When you publish results in the scientific literature which were obtained with the MKMCXX program you should include the following reference:

Ivo A. W. Filot, Prof. Dr. Rutger A. van Santen, Prof. Dr. Emiel J. M. Hensen, *The Optimally Performing Fischer–Tropsch Catalyst*, *Angew. Chem. Int. Ed.*, 53: 12746-12750 (2014) (<https://doi.org/10.1002/anie.201406521>)

23.2 Zacros Kinetic Monte Carlo

Note: To use Zacros, you need a valid Zacros license that you can obtain from SCM (it is separate from the general Amsterdam Modeling Suite license).

GUI tutorial: *Zacros Kinetic Monte Carlo*

The kinetic Monte Carlo tools in the Amsterdam Modeling Suite based on Zacros include

- The Zacros program itself
- **pyZacros** : a Python library for running and analyzing Zacros simulations
- **AMSkinetics** : a GUI for setting up Zacros calculations (jobs can be submitted via AMSjobs)
- **Zacros-post**: a GUI for viewing Zacros results. Installable via the package manager (**SCM → Packages**).

When you publish results in the scientific literature which were obtained with the Zacros program you should include the following reference:

- Stamatakis, M. and D. G. Vlachos (2011). “A Graph-Theoretical Kinetic Monte Carlo Framework for on-Lattice Chemical Kinetics.” *The Journal of Chemical Physics*, 134(21): 214115.
- Nielsen, J., M. d’Avezac, J. Hetherington and M. Stamatakis (2013). “Parallel Kinetic Monte Carlo Simulation Framework Incorporating Accurate Models of Adsorbate Lateral Interactions.” *The Journal of Chemical Physics*, 139(22): 224706.
- Ravipati, S., Nielsen, J., d’Avezac, M., Hetherington, J. and M. Stamatakis (2020). “A Caching Scheme to Accelerate Kinetic Monte Carlo Simulations of Catalytic Reactions”. *The Journal of Physical Chemistry A*, 124(35): 7140-7154. [please cite if you are using the caching scheme]
- Savva, G. D. and M. Stamatakis (2020). “Comparison of Queueing Data-Structures for Kinetic Monte Carlo Simulations of Heterogeneous Catalysts”. *The Journal of Physical Chemistry A*, 124(38): 7843-7856. [please cite if you are using the skip-list or the pairing heap queueing system]
- Ravipati, S., Savva, G. D., Christidi, I.-A., Guichard, R., Nielsen, J., Réocreux, R., and Stamatakis, M. (2022). “Coupling the Time-Warp algorithm with the Graph-Theoretical Kinetic Monte Carlo framework for distributed simulations of heterogeneous catalysts”. *Computer Physics Communications*, 270: 108148 [please cite if you are using the MPI Time- Warp algorithm implementation]

GUI ENVIRONMENT VARIABLES

As of the 2016.102 version, if you start the MacOSX ADF-GUI application the environment defined by your shell startup scripts is ignored. Instead the bundled ADF is used, and environment variables may be defined in a file `$HOME/.scmenv` and/or `/Library/Application Support/SCM/scmenv`.

In the following list the environment variables that are specific for the ADF-GUI are listed (the normal environment variables like `AMSHOME`, `AMSBIN` and `SCMLICENSE` are also used by the ADF-GUI):

BROWSER: Fallback for **SCM_BROWSER**: the HTML browser used for both local and remote documents (not for Windows or Mac). Default value: none (the `PATH` will be searched).

DISPLAY: X-window display to use, when using ssh normally you should NOT set it manually. It is required (for all X11 programs) except on Windows.

SCM_ADFCRSDIR: Path to the folder containing the ADFCRS-2018 database, if different from the default set by `AMSPackages`.

SCM_AMSREPORT_PLAIN: No labels or units in amsreport output. Not set by default.

SCM_BROWSER: The HTML browser used for both local and remote documents (not for Windows or Mac). Default value: none (the `PATH` will be searched).

SCM_DEBUG: Turn on many debug items in ADF and avoid clean-up by GUI. Not set by default.

SCM_DEBUG_SSH: Set to yes to debug communication issues: all non-multiplexed ssh commands will be printed to stderr. Set to all to see all SSH commands, even if multiplexed. Not set by default.

SCM_DEFAULT_DIALOG: Set to 1 to get the original file dialogs back on Linux. Not set by default.

SCM_ERROR_MAIL: E-mail address for error reports. Default value: `errors@scm.com`.

SCM_DOCINDEXDIR: Location of the directory in which the index for the documentation will be stored. Setting it makes it possible to share this index and save some space. Default value: `$HOME/.scm_gui/doc.index`.

SCM_FFMPEG: The ffmpeg program to use to generate movies. Default value: none (the `PATH` will be searched).

SCM_FFMPEG_ARGS: The arguments to pass to ffmpeg. Default value: none. The `-i` and `-t` (for looping movies) will always be added automatically.

SCM_GEOMODSBYSIZE: The geometry slider will always move the smallest group of atoms. Default value: none (the last selected atoms will move).

SCM_GUIRC: Location of the preferences file. Default value: `$HOME/.scm_gui.rc`.

SCM_GUIPREFSDIR: Location of the preferences directory. Default value: `$HOME/.scm_gui`.

SCM_SCMJOB Restart: Force a restart of the scmjobd job monitors every this many milliseconds. Not set by default (600000 which is every 10 minutes).

SCM_KEEJOB: Do not remove the .job files automatically

SCM_KFBROWSER_EXPERT: Define it to make the expert mode in the Kfbrowser the default. The non-expert mode will be the default mode.

SCM_LFDFT: Path to the folder containing the LFDFT database, if different from the default set by AMSpackages.

SCM_LOG: Set to the full path of a file to log some stuff the GUI does (mainly invocation of other modules/programs). Not set by default.

SCM_LOG_SCMJOB: Set to the full path of a file to log some what scmjobd is doing. Not set by default.

SCM_NO_RSYNC: do not use rsync to transfer files to/from remote machines, but use the previous implementation (with tar/gz). Not set by default.

SCM_NO_LOGIN_SHELL_FOR_MONITOR: the scripts to monitor a running job normally run in a new login shell. Setting this env var will not force a login shell. Note it must be set on the machine where the monitor script is running, and you need to make sure that the queue commands (as defined in the AMSjobs queues) will work properly.

SCM_NO_SSH_MULTIPLEXING: If set SSH multiplexing (bundle multiple ssh connections in one) will be avoided. Not set by default.

SCM_OFFSCREEN: Set it to some value to force the GUI modules to run offscreen. Used for testing.

SCM_OLD_TRANSFERS: If set files are transferred one by one instead of in a tar archive (turn on when you run into tar compatibility issues). Not set by default.

SCM_OPENGL_SOFTWARE: Force software rendering, this often prevents problems with remote GUI use or incompatible graphics cards. It is significantly slower. Not set by default.

SCM_PACKMOL: The packmol program to used by the Builder. Default value: none (the PATH will be searched).

SCM_PARAMS_RESULTS_ONLY_SIZE: when opening a job in the ParAMS GUI, and results are present, ask if the results should only be loaded. You might want this for speed (loading large input sets may be slow). This is only done when the size of the input files is larger than the value of this env var. A special value of 0 means the feature is turned off (same as not setting it).

SCM_RESULTDIR: Location of the results directory. Default value: none (directory where the .ams files etc live is used).

SCM_SCMD_NOPIPE: On Windows only. Setting this to any value will disable the use of named pipes for inter-GUI communication. Local tcp/ip sockets will be used instead. Use this if your security software or policies blocks the use of named pipes.

SCM_STRUCTURES: Location of the structures directory. Default value: none.

SCM_TITLEVERSION: Show SVN revision number in title. Not set by default (show only version number).

SCM_TMPDIR: Temporary files will be made in this directory (should exists if set). Default value, if not set \$TMPDIR \$TMP /tmp \$HOME or the current directory is used (in that order).

SCM_TPLDIR: Location of the presets directory. Default value: none.

SCM_STATUSINTERVAL: The time in seconds between the job status commands used to detect killed jobs (like qstat). Default value: 600.

SCM_QUEUES: Path to the dynamic queues directory. Default value: none (AMSjobs will search the remote \$HOME/.scmgui).

SCM_USE_LOCAL_QE: When set, instead using the AMSpackage version of Quantum ESPRESSO, use a local version of QE. The Quantum ESPRESSO binaries (like pw.x, projwfc.x, ...) of the local QE version should then be accessible on \$PATH.

SCM_QESTARTCMD: Command called to launch Quantum ESPRESSO binaries (like pw.x, projwfc.x, ...). If not set, srun will be used if running under SLURM (aka SLURM_JOB_ID is set), and exec will be used otherwise. Might need to be changed when running jobs with the Quantum ESPRESSO through a batch system.

SCM_QSTARTCMD_OPTIONS: Options, to be passed to the QE start command defined by `SCM_QSTARTCMD`.

SCM_QERC: Can be set to the (absolute) path to a shell script that will be source right before a Quantum ESPRESSO binary is executed. (Environment setup that might be needed to run QE binaries can then be put into that script.)

AMSPACKAGES

25.1 General

AMSPackages is a GUI module that can be used to install optional software components. Users of the graphical user interface will automatically be prompted to install an optional component when the option is selected in AMSInput. Alternatively they can use SCMMMenu → Packages to install packages at any time. The graphical interface can also be started using "`${AMSBIN}/AMSPackages`" gui on the command line.

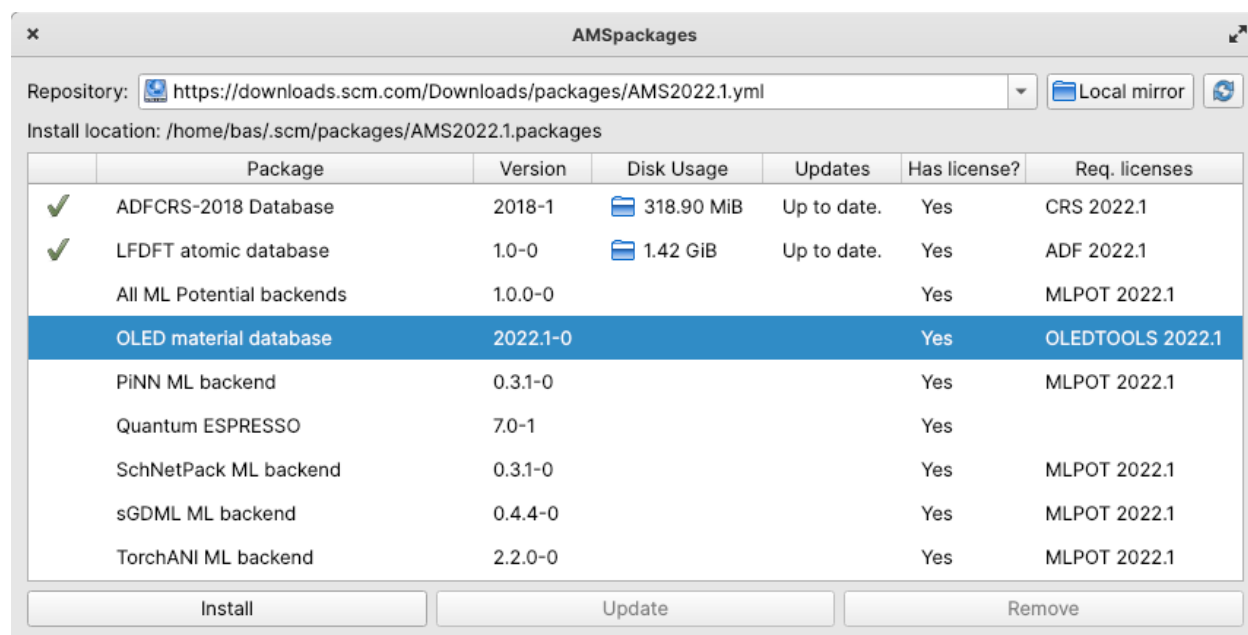


Fig. 25.1: The main window of the AMSPackages gui.

Command line features of AMSPackages are documented in the [installation](#) chapter of the manual.

25.1.1 Installing packages

Select the packages that you wish to install and click install. Some information will be printed to the screen regarding progress. If you wish to abort an installation that is in progress, you can click the abort button.

After the package is installed, it will automatically be detected by any programs that use it. To view the content of an installed package, you can double-click the folder icon in the disk usage column. This will open it inside your default file explorer.

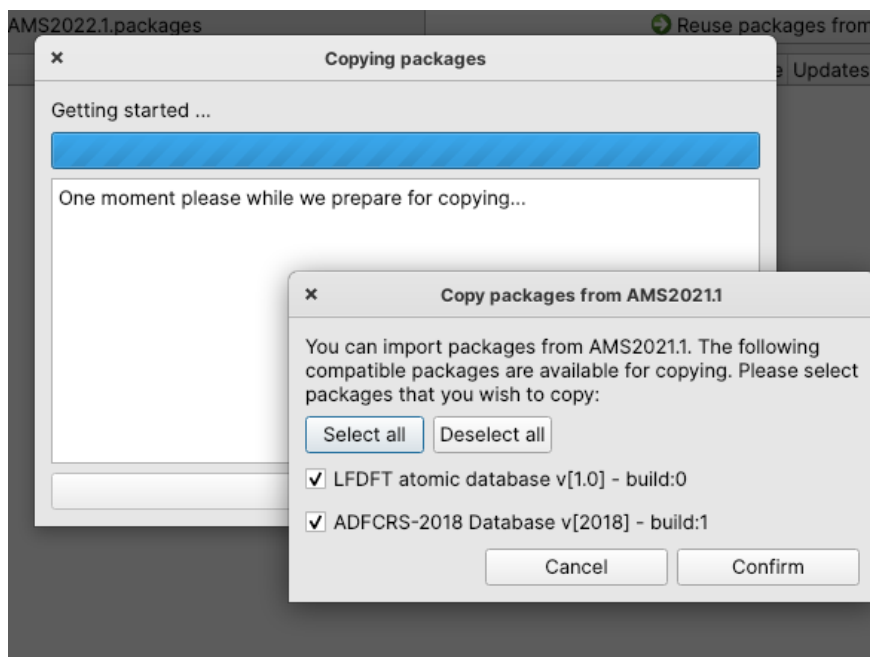
25.1.2 Removing packages

Select the packages that you wish to remove and click remove.

25.1.3 Reinstalling packages from a previous version of AMS

New in version AMS2022.101: Compatible components from a previous installation of AMS can be copied over, so redownloading and reinstalling is not necessary for some packages.

If compatible packages are detected, you will see the *reuse* button appear. Clicking it will provide you with a new dialog, where you can select your choice from the list of compatible packages.

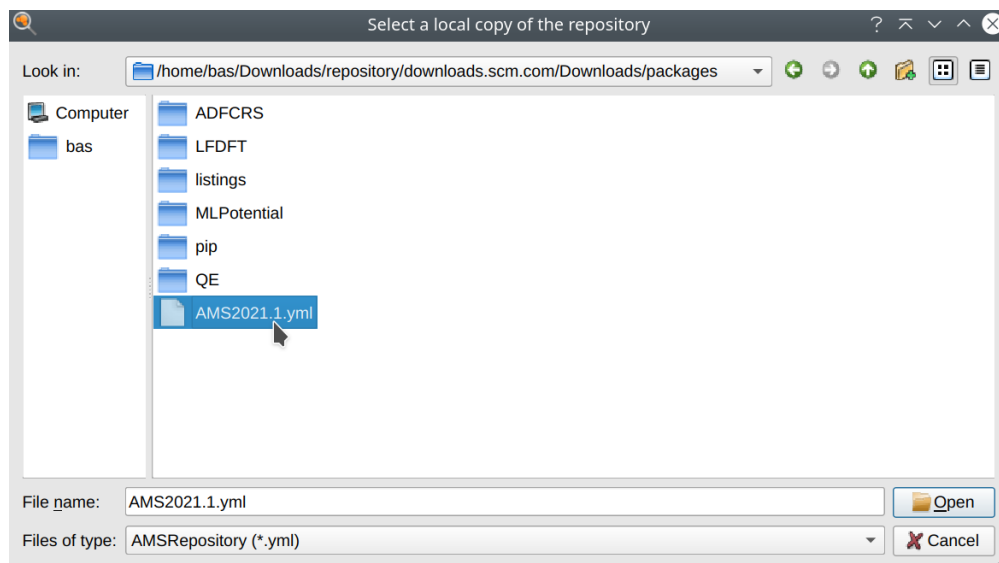


After clicking confirm, copying of selected packages will proceed.

25.1.4 Selecting an alternative package source

If you are on a computer with a slow connection or behind a firewall, you can select a local copy of our repository using the **local mirror** button. Navigate to your copy and select the `AMS2022.1.yml` file inside the `packages` directory.

Warning: Never install packages from an untrusted source. Only use a copy directly downloaded from our website.



Instructions on how to obtain a copy can be found in the [installation section](#).

REQUIRED CITATIONS

When you publish results in the scientific literature that were obtained with programs of the Amsterdam Modeling Suite, you are required to include references to the program package with the appropriate release number, and a few key publications.

26.1 Authors and General References

The current version of the GUI was developed by O. Visser and L. Groot.

Other contributors include: O. Carstensen, M. Luppi (artwork), L. Hong (NMR spin-spin coupling in AMSSpectra), M. Koenis (VCD tools in AMSSpectra), K. Salimans (GUI support for QE CPMD), Dennis Coppens (GUI support for VASP), Micha Davids (AMSKinetics, formerly Microkinetics: GUI support for MKMcxx), and the SCM developer team. Thanks to M. Mouthaan for testing and documentation.

If you used the GUI you may optionally include the reference:

AMS GUI 2025.1, SCM, Amsterdam, The Netherlands, <http://www.scm.com>

26.2 External programs and Libraries

Third party software used in the 2025.1 version of the Amsterdam Modeling Suite can be found in the file titled “third-party-software.txt” in the root of your AMS installation.

INDEX

B

.bgf, 74

D

.dcd, 74

P

.psf, 74

X

.xmol, 74

.xyz, 74