

# Export Controlled Information

**UCRL-SM-231803**

## **CRETIN User's Manual**

### **Version 2**

Howard A. Scott  
Hai P. Le

Lawrence Livermore National Laboratory

#### **EXPORT CONTROLLED INFORMATION**

Contains technical data whose export is restricted by statute. Violations may result in administrative, civil, or criminal penalties. Limit dissemination to U.S. Department of Energy and major U.S. DOE contractors. The cognizant program manager must approve other dissemination, in accordance with applicable regulations. This notice shall not be separated from the attached document.

**Reviewer signature/date:** Rodney M. Grayson 6/15/2007

Further dissemination authorized to U.S. citizens in DOE or DOE contractors only; other requests shall be approved by the LLNL Innovative Business and Information Services Department or the Author.

## Export Controlled Information

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

# Export Controlled Information

Introduction . . . . .	5
Acknowledgments . . . . .	6
Compiling CRETIN . . . . .	7
Parallel Processing . . . . .	11
Distributed Memory . . . . .	11
Shared Memory . . . . .	11
Runtime Files . . . . .	13
Input . . . . .	13
Output . . . . .	13
Execute Line . . . . .	15
Spectral Postprocessing . . . . .	17
Runtime Control . . . . .	19
Generator Commands . . . . .	21
Materials: . . . . .	21
Mesh: . . . . .	30
Continuum radiation: . . . . .	32
Spectral radiation: . . . . .	34
Lineshapes: . . . . .	36
Line radiation: . . . . .	36
Electron distribution: . . . . .	38
Laser Raytrace: . . . . .	38
Edits: . . . . .	39
Miscellaneous: . . . . .	42
Macros: . . . . .	48
Code Options . . . . .	51
Atomic Kinetics: . . . . .	51
Atomic Models: . . . . .	63
Radiation Transfer: . . . . .	68
Continuum Radiation: . . . . .	69
Spectral Radiation: . . . . .	72
Lineshapes: . . . . .	75
Line Radiation: . . . . .	78
Escape Factors: . . . . .	79
Electron Distribution: . . . . .	81
Temperature Calculations: . . . . .	82
Radiation transfer / Temperature evolution: . . . . .	86
Hydrodynamics: . . . . .	90
Laser Raytrace: . . . . .	91
Timesteps and convergence: . . . . .	93
Output: . . . . .	94
Neutral Diffusion: . . . . .	97
Summary of Switches . . . . .	101
Summary of Parameters . . . . .	112
Atomic Physics Input . . . . .	115
File organization: . . . . .	115
Model specification: . . . . .	115
Electron collisional excitation/ionization: . . . . .	117
Photoexcitation: . . . . .	119
Photoionization: . . . . .	119
Auger Excitation: . . . . .	120
Auger Ionization: . . . . .	121

# Export Controlled Information

Ion collisional excitation/ionization:	122
Sublevel transitions:	123
Xfile Input	124
Edits	128
Time and timesteps:	128
Mesh properties:	130
Plasma properties:	133
Electron Distribution:	140
Populations:	141
Rates:	148
Heating rates:	158
Continuum radiation:	163
Spectral radiation:	173
Line radiation:	178
Hydrodynamics:	183
Laser Raytrace:	185
Convergence:	185
Sizes, memory usage, cpu time:	186
Sample Generator Files	189
0-d kinetics	189
1-d radiation transfer	191
2-d line transfer	194
3-d radiation transfer	198
1-d Marshak wave	204
1-d laser-driven foam	207
Xfile reader	210
Index of Commands and Edits	211
Appendix: ULTRA and PDBVIEW basics	220

# Export Controlled Information

## Introduction

CRETIN is a 1-, 2-, and 3-dimensional non-LTE atomic kinetics / radiation transfer code. It self-consistently follows the time evolution of atomic populations and photon distributions as radiation interacts with a low-density plasma. Each element in the plasma is modeled with numerous atomic states, corresponding to the distribution of electrons in various atomic levels. Transitions between levels are caused by interactions between electrons, ions, and photons. Atomic level structure and transition rate information for each element must be supplied by the user, as described in the section on atomic physics input, or can be generated by the code using simple approximations.

CRETIN also includes Lagrangian hydrodynamics and laser ray tracing and absorption, for 1-dimensional geometries only, with equation of state (EOS) information generated from the atomic kinetics.

This document does not describe the physics modeled by the code or the numerical algorithms used in the code. Much of this information is contained in the paper: “GLF - A Simulation Code for X-Ray Lasers” by H.A. Scott and R.W. Mayle, Applied Physics B, Vol. 58, pp. 35-43, 1994. The description of the treatment of atomic kinetics and radiation transport (for continuum and lines) applies to CRETIN as well as to GLF. A more recent overview of the code is contained in “Cretin – A radiative transfer capability for laboratory plasma” by H.A. Scott, Journal of Quantitative Spectroscopy and Radiative Transfer, Vol. 71, pp. 689-701, 2001. The screened-hydrogenic atomic models generated by CRETIN are described in “Advances in NLTE modeling for integrated simulations” by H.A. Scott and S.B. Hansen, High Energy Density Physics, Vol. 6, pp. 39-47, 2010.

One of CRETIN's main virtues is that it is written in (near-) standard Fortran to be as portable as possible. A few routines are written in C to provide capabilities not available from the Fortran compiler on some systems. CRETIN requires either Cray-style pointers or Fortran 90 pointers for memory management.

CRETIN was originally written as an astrophysics research tool to model accretion disks. (The name CRETIN arose from a contraction of “accretion”.) The emphasis of the code for the last decade has been the simulation of laboratory plasmas and experiment. The specialized routines for accretion disks are no longer attached to the code.

Bug reports, comments and requests concerning CRETIN should be sent to:

[hascott@llnl.gov](mailto:hascott@llnl.gov)      or      [hle@llnl.gov](mailto:hle@llnl.gov)

# Export Controlled Information

## Acknowledgments

Like most simulation codes, CRETIN owes much to people whose names do not appear on the user's manual title page (and who do not wish to receive bug reports) and to experiences (both positive and negative) with other codes. The treatment of atomic kinetics borrows heavily from that used by Peter Hagelstein and Sam Dalhed in the code XRASER, as does the general approach of evolving temperatures rather than energy densities. Sam Dalhed, Stewart Brown and Ron Mayle have influenced CRETIN's development over an extended period of time. Valuable suggestions, criticisms and/or coding were provided by George Zimmerman, George Maenchen, David Eder, Pieter Dykema and numerous others. The atomic data generated by CRETIN has benefited greatly from contributions of Hyun-Kyung Chung and Stephanie Hansen. For most codes, it is the users who find many of the bugs and drive much of the development, and Alan Wan and Steve Langer have certainly done their part.

# Export Controlled Information

## Compiling CRETIN

The standard distribution of CRETIN includes a set of source files, sample configuration files and test files. Source files with the extension *.cmn* are used as include files during the compilation. There are two sets of these include files. The files in directory *src/include/cft* are appropriate for compilers which support Cray-style pointers. If you wish to use Fortran 90 pointers, then use the include files in directory *src/include/f90*. (Most Fortran 90 compilers on UNIX workstations also support Cray-style pointers). All but one of the remaining source files are Fortran (fixed-format) source files and should have the extensions *.f*, or *.F*. The files with a *.F* extension contain system-dependent coding and must be preprocessed with a C preprocessor before compilation. The file *utilityc.c* is a C source file which is needed for a few systems or when using XGRAFIX for runtime graphics.

On most systems, a makefile is available or can easily be constructed to control compilation. For UNIX systems, the simplest option is to use the **crcconfig** utility, as described below. It will construct an upper level makefile, plus a makefile for each *src* directory. This can also be done by hand by concatenating the three files: *makehead*, *config*, and *maketail*. System-specific information such as compiler flags, libraries and paths resides in *config*. Various *config* files for different systems and scripts to manage the configuration and make process are included with the distribution in directory *config*. For WINDOWS systems, additional compilation information is given below.

NOTE: Check the config files for more specific information about individual systems.

When preprocessing, you must define (using -D) the system that you are using. The following table lists some systems on which CRETIN has been run and appropriate define flags.

<u>System (OS)</u>	<u>-D flags</u>
IBM (AIX)	AIX, [F90]
HP (HPUX)	HPUX
Cray (Unicos)	UNICOS, [F90]
Cray T3D	T3D
Linux w/ PGI or Intel Fortran	UNIX, [F90]
Linux w/ Absoft Pro Fortran	UNIX, SIGNALC, STATIC_CHAR
Linux w/ gfortran	LINUX, [F90]
Windows w/ Visual Fortran (Intel or Compaq)	DF, [F90]
Windows w/ Cygwin + gfortran	LINUX
Macintosh w/ PGI, Intel Fortran or gfortran	UNIX, [F90]
Macintosh w/ AIX compiler (OSX)	AIX, [F90]
Generic	[F90]

# Export Controlled Information

For compiling CRETIN using the configuration scripts on a UNIX system, you must set the environment variable CRDIR to point to the top of the CRETIN directory structure (i.e. the directory which contains *src*, *config*, and *test*). The main configuration tool is *crconfig*. “**crconfig -help**” will explain the usage of this script, which is also summarized here. If desired, run “**crconfig -v**” to update the version file with the current date and time. Run “**crconfig xxx**” to configure directories and construct the makefiles for version xxx, using config file *config.xxx*. This will leave *Makefile.xxx* in the top directory for compiling the main code and each of the included packages, as well as a *Makefile* in each package directory. This configuration system constructs a separate directory tree for the binary files, with the top level called xxx. A single config file *config.yyy* can be used for multiple code versions by using the syntax “**crconfig xxx : yyy**”. This will configure directories and construct the makefiles for version xxx, using config file *config.yyy*. A third argument, e.g. “**crconfig xxx : yyy : zzz**”, specifies that the script *runtest.zzz* has the correct syntax for running short test cases on the system, if a special syntax is required for parallel execution or for setting environment variables. Additional tools, *crtest* and *certify*, are useful for running the included tests with the compiled executable and comparing to stored results.

### Examples of using `crconfig`:

To compile an optimized version using the Portland Group compiler on a Linux system -

**crconfig pgi : linux**

To compile a debuggable version using gfortran with F90 pointers on an OSX system -

**crconfig gfortran.f90.debug : osx**

To compile a parallel (MPI+OpenMP) version on a 64-bit AIX system which uses srun for execution

**crconfig** aix.mpi.openmp.64 : aix : srun.openmp

The *config* files included with the distribution are designed to work for multiple code versions (e.g. optimized, debuggable, message passing, threaded), specified by the name of the version. The mechanism for doing this uses conditionals (*ifeq*, *ifneq*) which are supported by *gmake*. The conditionals key off the appearance of a particular string (e.g. “debug”, “f90”, “mpi”, “openmp”) anywhere in the version name, and multiple such strings may be combined in the version name. If your make utility is not compatible with these conditionals, construct a separate *config* file without conditionals for each version. The resulting makefile should work with almost any make utility. A sample file without conditionals, *config.generic*, echoes the defaults which are set in *makehead* and can be used as a template.

The default configuration used by most of the config files assumes that the PACT libraries are available on your system and that XGRAFIX will be compiled for runtime graphics – see the description of these later in this section. These can be omitted by the use of the *crconfig* options “-nopdb” and “-noxgr”.

For compiling CRETIN using Visual Studio on WINDOWS systems, follow these steps:

1. Create a Project of type QuickWin or Console Application. QuickWin will allow the use of multiple windows, dialog boxes and the SciGraph display library – define WINDOWS in this case.
  2. Add all the ".f" and ".F" files in .../src/main and .../src/xxx (for each desired directory) to the project except message.f and parallel.F (and divertor.F, if not desired).
  3. Change Project->Settings as follows:

Under Fortran:

Compatibility:

add “filenames from command line”

processor:

# Export Controlled Information

add “Preprocessor symbol to FPP only”  
Predefined Symbols: e.g. DF WINDOWS ...  
INCLUDE paths: add .../src/include  
and .../src/include/cft or .../src/include/f90  
(if using TOTAL: add .../src/main/total)

(/nodefine)  
(/define: “DF WINDOWS ...”)  
(/include:”.../src/include”)  
(/include:”.../src/include/xxx”)  
(/include:”.../src/total”)

If using CFT:

Under Fortran:  
Optimization:  
“None” for generate.F

If using F90:

Compilation Diagnostics:  
remove “data alignment”

(/warn:noalignments)

If using PDB:

External Procedures:  
String Length Argument Passing: After All Args

Under Link->Input:  
Object modules: pdb.lib, pml.lib, score.lib  
Ignore libraries: add libcd.lib  
Additional library path: path to above libraries

(/iface:nomixed\_str\_len\_arg)

(/nodefaultlib:"libcd.lib")  
(/libpath:" ... ")

If using SCIGRAPH:

Under Link->Input:  
Object modules: scigraph.lib  
Additional library path: path to above library

(/libpath:" ... ")

# Export Controlled Information

The conditional compilation accounts for differences between systems such as timing calls, date and time routines, memory management and handling of execute lines and interrupts. The F90 flag denotes the use of Fortran 90 pointers, while the lack of the F90 flag requires a compiler that supports Cray-style pointers. On Cray systems, the compiler flag which restricts double precision variables to 64 bits (-d p) should be used.

Depending on your system, you may either compile or link to the linear algebra routines used by CRETIN. The libraries to link to are *libslatec.a*, *liblapack.a*, and *libblas.a*, or equivalents. If any one of these is unavailable, you may compile the appropriate file (*dslatec.f*, *dlapack.f*, or *dblas.a*).

NOTE: If you do compile *dslatec.f*, you may have to modify the two subroutines with system-dependent data statements (*DIMACH* and *IIMACH*) to match your system. It is highly unlikely that this will be necessary.

The neutral diffusion package is included with the standard distribution in the *main* directory but is considered a separate package. To include this package, define DIVERTOR while compiling the code.

Runtime graphics are currently available on UNIX systems using XGRAFIX for which a modified version 1.94 is included as a package in the Cretin source. In this case, define XGRAFIX and link to *libX11.a*. You must also include *utilityc.c* when compiling. This version of XGRAFIX was obtained from the Plasma Theory and Simulation Group of U.C. Berkeley (<http://langmuir.eecs.berkeley.edu/pub/codes/xgrafix>).

Runtime graphics are also available on WINDOWS/DOS systems. On WINDOWS 95/98/NT/2000/XP/... systems with Visual Fortran, the SCIGRAPH library (included under SAMPLES), can be used – bug fixes or a precompiled library are available upon request. In this case, define SCIGRAPH when compiling. Under WINDOWS or DOS with the Lahey compilers (F77L3, or LF90) use the utility library (UTIL[3,90]) and the Graphoria library (GRAPH[3,90]). In this case, define LAHEY when compiling.

Graphical postprocessing can be accomplished in a number of ways. If PDBLIB is available on your system, define PDB and link with *libpdb.a*, *libpml.a*, and *libscore.a*. CRETIN will then produce PDB files directly readable by ULTRA and PDBVIEW. PDBLIB, ULTRA and PDBVIEW are part of the Portable Application Code Toolkit (PACT) which is available at <http://pact.llnl.gov>. Portions of the ascii files produced by CRETIN can also be used as input to most graphics programs. PDBLIB is also necessary for producing restart dumps and other binary dump files. PDBLIB is available for WINDOWS and Mac pre-OS X systems, but ULTRA and PDBVIEW are not.

Another postprocessing option is YORICK, an interpreted language specialized for numerical analysis. It can read PDB files and includes graphics capability. It is available for UNIX, WINDOWS and Macintosh systems at <http://yorick.sourceforge.net/>. The include file *ultra.i* provides the capability to read data from the PDB graphics files produced by CRETIN. YORICK can directly read data from restart dumps and other binary dump files – include *basfix.i* to make this easier. Binary dump files will be structured as YORICK time history files if the **dump history** command is used. YORICK scripts for working with CRETIN files are available by request.

HDF5 may now be used in place of PDBLIB for restart dumps and most other binary dump files, but not for ULTRA files. To use HDF5, define HDF and link with *libhdf5.a* and *libhdf5\_fortran.a*.

NOTE: The *crconfig* script can configure makefiles for using PDBLIB, HDF5 and/or XGRAFIX. Run “**crconfig -help**” for instructions.

# Export Controlled Information

## Parallel Processing

CRETIN supports two models of parallel processing - a shared memory model (using threads for parallelized loops), and a distributed memory message-passing model (using MPI). Both models may be used simultaneously.

The hydrodynamics, laser ray trace and neutral diffusion packages are not currently parallelized with either model.

### Distributed Memory

To compile the message-passing version, define MP and link to the appropriate libraries for your system. The files *parallel.F* and *message.f* must be included. Message-passing libraries other than MPI may also be used by replacing the MPI calls in *message.f*.

The atomic kinetics is parallelized by zone. Each processor receives a comparable number of zones, regardless of the distribution of elements within the zones. A crude attempt is made at static load balancing at problem startup, using the relative complexities of the atomic models. This will be replaced in the near future with dynamic load balancing using the actual computational expense of each zone.

There are two options for parallelizing the radiation transfer. The default option replicates the spatial mesh on each processor. With this option, both the continuum transfer and spectral calculations are parallelized over frequencies, with each processor receiving an equal number of frequencies. Since the computational expense is identical for each frequency, no further load balancing is necessary. The line transfer is parallelized by line groups, i.e. groups of lines which are transferred simultaneously due to overlaps or other interactions. Radiation transfer with the rad package is not parallelized and is incompatible with the use of distributed memory.

The second option decomposes the mesh into domains, with each processor receiving (at most) one domain. Each processor then handles all frequencies, angles and line groups, while exchanging boundary information with other processors. This option is usually less efficient, but scales up to much larger mesh sizes. This option is chosen by the execute line argument “**domains=k,l,m**”.

The restriction on the physics performed by the parallel code is that the rad package may not be used with distributed memory. The hydrodynamics will run on a single processor, but is compatible with the use of distributed memory. Also, with mesh replication (i.e. the default option), the continuum quantities associated with a line are assumed to have constant values, corresponding to line center, over the entire line profile (equivalent to setting switch(38) ≠ 0).

### Shared Memory

Compiling the shared-memory version requires use of an OpenMP-compliant compiler or preprocessor (see <http://www.openmp.org>). Specify the OMP\_NUM\_THREADS environment variable at run time to set the maximum number of threads. Note that if the environment variable is not set, the maximum number of threads will be set to 1.

The atomic kinetics is parallelized by zone. Reasonable load balancing is achieved by giving each processor the most expensive zone remaining to be calculated as the processor finishes the previous zone.

Both the continuum transfer and spectral calculations are parallelized over frequencies, with each processor receiving an equal number of frequencies. Since the computational expense is identical for each frequency, no further load balancing is necessary. Radiation transfer with the rad package is parallelized over frequencies.

## Export Controlled Information

In 1-d, the line transfer is parallelized by line groups, i.e. groups of lines which are transferred simultaneously due to overlaps or other interactions. In 2-d and 3-d, the line transfer is parallelized over directions.

The hydrodynamics and laser raytrace are not parallelized.

# Export Controlled Information

## Runtime Files

### Input

CRETIN requires an ascii generator file to specify the problem to be run and the edits to be produced. Generator commands are discussed in their own section of this manual.

CRETIN requires an ascii atomic datafile for each atomic model specified in the generator, with the exception of screened-hydrogenic atomic models generated internally. The format is discussed later in this manual. Data types supported include: *coes*, *coll excite*, *coll ionize*, *phxs*, *phis*, *augxs*, *augis* (including *rec*, *rad*, *aug* and *cex*), and *cois*. For users of HULLAC or FAC, CRETIN also supports data types *colex*, *colon*, *rad\_rate*, *phot\_cs*, and *aut\_rate*.

CRETIN requires a supplementary data file (xfile) for each **xfile** command in the generator. These files can be either ascii or binary. If you specify an ascii xfile, CRETIN will make a binary or PDB/HDF5 version (a file with the same name and extension *.xdf*) which you can use in subsequent runs. The format for the xfile is discussed later in this manual. The xfile can be used either for specifying initial or time-dependent conditions.

When running from a restart dump, the only files necessary are the restart dump and the binary xfiles.

The locations of the atomic datafiles, xfiles, and redistribution files can be specified by including the path (relative or absolute) with the filename in the **atoms**, **xfile** or **r2file** commands. The file is looked for first in the specified directory (or current directory, if no path is included) and then in the same directory as the generator file. If it is not found in either directory, CRETIN will search in the directory specified by the environment variables AFILEDIR (for the atomic datafiles), XFILEDIR (for the xfiles), or CRDATA (for the redistribution files).

Other input files are assumed to be in the same directory as the generator file. Paths are interpreted relative to the current directory. On WINDOWS systems, the current directory is set to the directory containing the generator file.

### Output

CRETIN can produce a number of different output files. The output files will have the same base name as the generator (unless otherwise requested) with different extensions:

<u>extension</u>	<u>file</u>
.tbl	ascii output, tables
.plt or .pxx	ascii plots
.ult or .uxx	binary plots (PDB)
.rxx	restart dump (PDB/HDF5)
.sxx	spectral dump (PDB/HDF5)
.dxx	dumpfile (PDB/HDF5)
.qxx	sensitivity dump (PDB/HDF5)
.opc	opacity dump (PDB/HDF5)
.lrm	linear response dump (PDB)
.drt	Drat spectral output (PDB/HDF5)

where *xx* indicates one of a sequence of two-digit numbers: 00, 01, 02, .... The plot files will have numbered extensions if switch(12) is set.

## Export Controlled Information

The ascii output (*.tbl*) file will contain a copy of the generator, warning or error messages from CRETIN, and requested tables. If PDB was defined, the requested plots will by default be contained in a binary (*.ult*) file which can be read directly by ULTRA and PDBVIEW. If PDB was not defined, the plots will be contained in an ascii (*.plt*) file in a format suitable for use with a number of graphics packages. Under DOS or WINDOWS, a postprocessor is available to automate producing plots with EASYPLOT. A file format other than (or in addition to) the default can be chosen by setting switch(11) to the appropriate value.

Restart dumps, spectral dumps, dumpfiles, and sensitivity dumps will be produced upon request, if PDB or HDF5 is available. Each restart dump contains sufficient information from a single time to continue executing the problem. Spectral dumps (sdumps) will contain sufficient information to produce detailed spectra, for one or more times. Dumpfiles contain the information requested by **dump** commands, for one or more times. Edits may also be included in dumpfiles. Sensitivity dumps contain information sufficient to perform a sensitivity analysis for the zones requested.

**WARNING: CRETIN does not check for a previous version before creating most files and will overwrite any such files!**

Note that if switch(12) is set to a non-zero value, the *.tbl*, *.plt* and *.ult* files will have a numbered suffix. In this case, restarting a problem will not overwrite the previously made files. If switch(12) is set to a negative value, the first such suffix used when starting or restarting will be incremented so as to avoid overwriting any existing file.

The location and base name of all output files may be set with the **out=name** (or **-o name**) execute line option. If *name* includes a path component (relative or absolute), all output files will be placed in the specified directory (which must already exist). If *name* includes a filename component, all output files will have the same base as the specified filename.

# Export Controlled Information

## Execute Line

The basic form of the execute line which will run the problem specified by generator (or restart) file *problem* is

**CRETIN** *problem* [*options*]

There is a limited capability to modify the problem defined by a restart dump with commands in an additional generator file. The generator can change switch and parameter settings, add new edits and problem specifications. The generator name must appear immediately after the problem name (and must not match any legal option):

**CRETIN** *problem* *generator* [*options*]

Execute line options:

<b>pause</b>	stop in interactive mode after initialization but before the initial timestep
<b>generate</b>	run only through the initial timestep (far enough to produce the initial restart dump)
<b>opacity</b>	run only through the initial timestep and produce an opacity dump
<b>zfiles</b>	use existing hydrogenic models (ascii or binary) instead of generating models
<b>out=name   -o name</b>	base all output filenames on <i>name</i> (rather than the generator or restart dump name)
<b>groups=ng</b>	divide the continuum radiation energies into <i>ng</i> groups for radiation transfer.
<b>domains=k,l,m</b>	spatially decompose mesh with <i>k</i> , <i>l</i> , and <i>m</i> domains in the k-, l-, and m-directions (missing arguments have value 1, e.g. <b>domains</b> =2 is equivalent to <b>domains</b> =2,1,1)
<b>define var</b>	assign value 1 to symbol <i>var</i> when interpreting generator commands
<b>undefine var</b>	assign value 0 to symbol <i>var</i> when interpreting generator commands
<b>alias var value</b>	assign <i>value</i> to symbol <i>var</i> when interpreting generator commands
<b>display off</b>	run without displaying runtime graphics
<b>nohold</b>	on Windows and Mac systems, do not keep a window open after execution
<b>trace</b>	provide notice when entering or leaving major packages
<b>tracemem</b>	allocating or deallocating memory (Cray pointers only)
<b>tracemsg</b>	sending messages between processors
<b>traceall</b>	any of the above conditions

Execute line options for a restart dump only:

<b>zone=n</b>	extract zone <i>n</i> from the restart dump and run 0d physics for this zone only.
<b>reset</b>	restart the problem from the initial time but the current conditions

Execute line options for spectral postprocessing:

<b>spectra</b>	see discussion of <a href="#">Spectral Postprocessing</a>
<b>drat</b>	

# Export Controlled Information

Other execute line forms which do not run a problem:

<b>version   -v</b>	identify the current version of RETIN
<b>help   -h</b>	show execute line options
<b>size</b>	give the best estimate of what the parameters in <i>param.cmn</i> , etc. should be, as well as the memory required to run the problem
<b>threads=<i>nthreads</i></b>	assume the problem will run on <i>nthreads</i> threads per processor (only valid with <b>size</b> )
<b>procs=<i>nprocs</i></b>	assume the problem will run on <i>nprocs</i> processors (only valid with <b>size</b> )
<b>xname source [binary history directoried]</b>	
	produce a binary   Yorick history   directoried PDB xfile from the ascii xfile <i>xname</i>
<b>klm=<i>k,l,m</i></b>	expand mesh dimensions by factors of <i>k,l,m</i> while producing binary xfile (only valid with <b>source</b> ) – this option can be used to change the dimensionality of the xfile
<b>klm=<i>k1:k2:k2,...</i></b>	extract specified zones while producing binary xfile (only valid with <b>source</b> ) – this option can be used to create a subset from the xfile, or decrease the dimensionality of the xfile
<b>xyz=<i>i,j,k</i></b>	permute coordinate axes from the ascii xfile to the order specified, where <i>i,j,k</i> denote the data in the xfile, with values of 1,2,3 denotes the x-, y-, and z-axes, respectively
<b>bname reconst</b>	produce an ascii xfile from the binary xfile <i>xname</i>
<b>afiles   bfiles</b>	construct hydrogenic models specified in generator in ascii   binary format and stop
<b>t0=<i>time0</i></b>	set initial xfile time <i>time0</i> to consider (used with <b>source</b> or <b>reconst</b> )
<b>t1=<i>time1</i></b>	set final xfile time <i>time1</i> to consider (used with <b>source</b> or <b>reconst</b> )

Atomic model execute line options:

When using the first of these options, RETIN will construct a screened-hydrogenic atomic model and write it to the text file *problem.z01*. The elemental symbol must follow immediately after **hydrogenic**. The **hydrogenic** option may have the same extensions described under the **atoms** command. Options are available to transform between ascii and binary atomic models, and to produce a sorted transition list which will appear in the *aname.tbl* output file.

<b>hydrogenic</b> <i>symbol</i>	produce a screened-hydrogenic model <i>problem.z01</i> for element <i>symbol</i>
<b>dca</b>	make the model comparable to DCA (see <b>modeltype dca</b> )
<b>dca_v18</b>	make the model comparable to DCA from RETIN version 2_18
<b>atoms</b> <i>symbol</i>	produce a binary atomic model <i>problem.adf</i> from the ascii model <i>problem</i>
<b>reconst</b>	produce an ascii atomic model <i>problem.z00</i> from the binary model <i>problem</i>
<b>radlist</b>	produce a sorted list of radiative transitions for the atomic model <i>problem</i>

# Export Controlled Information

## Opacity dump execute line options:

When using these options, CRETIN will construct LTE opacities or NLTE opacities / emissivities for each zone for the frequency grid contained in the generator. Temperatures and densities should be initialized on a 2-d logical mesh, with temperatures varying in the k-direction and densities varying in the l-direction. Conditions and control settings specified for initialization will be used for a single timestep. If Linear Response Matrix (LRM or JRM) information is requested, one additional timestep per frequency will be used to construct derivatives.

All dumps except SERVER and TABOP format contain EOS information. TABOP and expanded TABOP formats produce ascii dumps, while all other dumps are binary (PDB or HDF5).

<b>opacity</b>	LTE opacities
<b>opacity+</b>	NLTE opacities and emissivities
<b>server</b>	LTE opacities in server format
<b>server+</b>	NLTE opacities and emissivities in server format
<b>opcserver</b>	same as <b>opacity</b> and <b>server</b>
<b>opcserver+</b>	same as <b>opacity+</b> and <b>server+</b>
<b>tabop</b>	LTE opacities in TABOP format
<b>tabopeos</b>	LTE opacities and EOS in expanded TABOP format
<b>busquet</b>	NLTE opacities and emissivities, and LTE opacities for $T_e$ with matching $\langle Z \rangle$
<b>lrm</b>	LTE opacities and emissivities plus LRM derivatives
<b>jrm</b>	NLTE opacities and emissivities plus derivatives for $J = f^*B(T_e)$
<b>jrm1</b>	NLTE opacities and emissivities plus derivatives for $J = B(f^*T_e)$
<b>jrm2</b>	NLTE opacities and emissivities plus derivatives for $J = B(T_r)$

# Export Controlled Information

## Spectral Postprocessing

CRETIN can be run in a manner so as to postprocess a series of restart or spectral dumps to only perform spectral calculations. In this manner, the results of a single simulation can be used to produce spectral information for different code options or edit requests without rerunning the kinetics/transfer simulation. CRETIN will step through each timestep in all restart/spectral dumps with the same base name.

When running in this manner, the generator file must include all definitions of the spectral bins and groups and all edit definitions that are desired. These will not be carried over from the existing dumps. If any **stark** commands are included in the generator, previous definitions concerning these options contained within the dumps will be ignored. If no **stark** commands are included, the previous definitions will be used. Commands dealing with materials, mesh, atomic kinetics, continuum transfer and line transfer are not executed. Parameter settings controlling atomic kinetics, continuum transfer and line transfer will have no effect. If **tstart** and **tquit** are not specified in the generator, the values in the first restart/spectral dump will be used.

For this mode, with first member *name.sxx* (or *name.rxx* for a restart dump) and generator file *generator*, the execute line is

**CRETIN** *name.sxx generator spectra [options]*

If the commands in *generator* set switch(92) to <0, the spectral opacities and emissivities will be read from a dump file *name.dxx* with the same base *name* and integer suffix *xx*.

CRETIN can also be run in a manner so as to postprocess a series of dump files which contain mesh information and spectral absorption coefficients and emissivities to produce spatially-resolved spectral images, similar to those produced by the Drat and Hex codes.

For this mode, with first member *name.dxx* and generator file *generator*, the execute line is

**CRETIN** *name.dxx generator drat [options]*

The dump files can be produced by a run which includes the appropriate **dump** commands. The generator for the postprocessing will consist of primarily **drat** commands.

# Export Controlled Information

## Runtime Control

Under most operating systems, you can suspend execution and interact with CRETIN during a run. Under WINDOWS, hit the right mouse button. Under DOS (with LAHEY defined), hit the escape key. Under most other systems, hit CTRL-C once (two successive CTRL-C's will stop execution). CRETIN will pause at the end of the current timestep. The following commands are available:

<b>help</b>	- produces this list of options
<b>fin / end</b>	- finish timestep and stop
<b>quit / kill</b>	- quit immediately without updating output files
<b>go [n]</b>	- continue execution [for <i>n</i> cycles]
<b>go for <math>\Delta t</math></b>	- continue execution for time $\Delta t$
<b>go until <i>t</i></b>	- continue execution until time <i>t</i>
<b>when cycle N</b>	- start definition of a “when” clause
<b>when time <i>t</i></b>	- start definition of a “when” clause
<b>end</b>	- finish definition of a “when” clause
<b>edits</b>	- make time-independent edits
<b>edits on / off</b>	- start / stop making time-independent edits
<b>restart</b>	- make a restart dump
<b>restart on / off</b>	- start / stop making restart dumps
<b>sdump</b>	- make a spectral dump
<b>sdump on / off</b>	- start / stop making spectral dumps
<b>sense on / off / all / <i>i1:i2:i3 ...</i></b>	- start / stop / define zones for sensitivity dumps
<b>nltdump on / off / all / rates <i>i1 i2 i3</i> ...</b>	- start / stop / define zones for NLTE workshop dumps
<b>ratedump on / off / all / <i>i1:i2:i3 ...</i></b>	- start / stop / define zones for rate dumps
<b>gainlist on / off / all / <i>i1:i2:i3 ...</i></b>	- start / stop / define zones for gainlist edits
<b>emislist on / off / all / <i>i1:i2:i3 ...</i></b>	- start / stop / define zones for emislist edits
<b>reconst <i>iz</i> / all</b>	- produce an ascii atomic datafile for model <i>iz</i> or all models
<b>binary <i>iz</i> / all</b>	- produce a binary atomic datafile for model <i>iz</i> or all models
<b>tquit</b>	- print value of quit time
<b>tquit <i>t</i></b>	- change quit time to <i>t</i>
<b>time</b>	- print the current time
<b>dtime</b>	- print the current timestep
<b>dtc</b>	- print the current cycle, time and timestep
<b>dtinfo</b>	- print the current timestep constraint
<b>reset</b>	- reset to initial time and initial timestep
<b>reset <i>t</i></b>	- reset to time <i>t</i> and initial timestep
<b>reset <i>t dt</i></b>	- reset to initial <i>t</i> and timestep <i>dt</i>
<b>size</b>	- print parameter values and memory usage
<b>timing</b>	- print cpu usage
<b>param <i>i</i></b>	- print value of param <i>i</i>
<b>param <i>i value</i></b>	- change value of param <i>i</i> to <i>value</i>
<b>switch <i>i</i></b>	- print value of switch <i>i</i>
<b>switch <i>i value</i></b>	- change value of switch <i>i</i> to <i>value</i>
<b>echo <i>var</i></b>	- print the numerical value of alias <i>var</i>
<b>trace</b>	- print trace information during execution
<b>list titles</b>	- print titles of currently defined edit requests
<b>list <i>i</i></b>	- print table defined by edit request <i>i</i>
<b>list <i>yvar xvar i1 i2 i3 i4</i></b>	- print table of <i>yvar</i> vs. <i>xvar</i>
<b>edit <i>i1:i2:i3 j1:j2:j3 k1:k2:k3 l1:l2:l3</i></b>	- print the requested values of <i>edit</i>
<b>define table / plot / edit / display</b>	- define a new edit request

## Export Controlled Information

If LAHEY is defined or XGRAFIX is included, the following commands are also available:

<b>plot</b> <i>i</i>	- plot curves defined by edit request <i>i</i>
<b>plot</b> <i>yvar xvar i1 i2 i3 i4</i>	- plot <i>yvar</i> vs. <i>xvar</i>
<b>display</b> <i>i</i>	- plot curves defined by edit request <i>i</i>
<b>display</b> <i>yvar xvar i1 i2 i3 i4</i>	- plot <i>yvar</i> vs. <i>xvar</i>

With LAHEY defined, **plot** will send the requested plot to the screen. When a plot appears on the screen, hitting any key will return control to the keyboard. **Display** has the same effect as **plot**, except that CRETIN will continue running and will update the screen at every timestep, sending each defined display to the screen in succession. Hitting the escape key at any point will return control to the keyboard at the end of the current timestep.

With XGRAFIX included, **plot** and **display** have the same effect. Each will open an additional window with the requested plot. There may be up to MDISP displays defined, including those defined in the generator, subject to the constraint that the total number of edits (**edit**, **table**, **plot**, and **display** commands) plus the total number of displays may be no more than MEDIT. The QUIT button has the same effect as **kill**. The SAVE button has the same effect as CTRL-C. The SPECIAL button has no effect.

All interactive commands are available for execution at predetermined times with the **when** command except for **define** and **when**.

# Export Controlled Information

## Generator Commands

All commands should be in lower case. Blank lines can be freely interspersed within the generator file. Lines beginning with *c* (followed by a space) are interpreted as comments. All input on a line following an exclamation point (!) will be interpreted as a trailing comment. Arguments within brackets are optional. Extra arguments are ignored and can also be used as trailing comments, but may be misinterpreted if the command syntax is extended in the future.

A rudimentary macro interpreter is available through the **alias** command, described in the [Macros](#) section. Macro values may also be set from the command line, allowing simple scripts to generate and run a series of problems from a single generator file.

Commands may be conditionally included through use of **#ifdef**, **#ifndef**, **#elseif** (or **#elif**), **#else**, and **#endif**, clauses. **#ifdef X** will be evaluated as true if X has been given a non-zero numerical value through the use of **#define** or **alias** before the clause is tested. The value may also be set either earlier in the generator file or on the command line.

Multiple files may be incorporated into the generator through the use of **#include** statements. Note that all path information will be interpreted relative to the location of the orginal generator file.

### Materials:

The following commands apply to the last region defined before the command is encountered:

**rho, element, level, material, background, ne, econd, bfield, opacity, esource, esourci, vacuum**

The following commands apply to the last atomic model defined before the command is encountered:

**isorange, modeltype, atoms+, subatoms**

```
atoms [iz=iz] filename z a
atoms [iz=iz] filename symbol [reconst|binary|pdb|hdf5]
atoms [iz=iz] hydrogenic[_nmax] symbol
atoms [iz=iz] hydrogenic(isomin:isomax)/_nmax symbol
```

Specify an atomic model contained in the datafile *filename*. The atomic number and weight are given by *z* and *a*, or are assigned according to the standard elemental *symbol* (e.g. "h" or "fe"), if that is specified instead. The symbols "h2", "h3", "d", and "t" are also valid. These values will be overwritten if specified with an "atom" command in the datafile. If the optional *iz=* argument is present, the model will be given the specified index *iz*. Otherwise, the model will be given an index *iz* corresponding to its order of appearance in the generator file. 'Null' is a valid *filename* and can be used for removing a model without renumbering the remaining models. If a datafile is specified, it must be available at generation time in the named directory if *filename* includes a path, or if no path is included, in either the working directory or the directory specified by the environment variable AFILEDIR.

If **hydrogenic** is specified as the datafile, the data appropriate for a screened-hydrogenic model for atomic number and weight given by *z* and *a* (or *symbol*) will be generated internally. All isosequences, from bare nucleus to neutral atom will be included unless minimum and maximum isosequences are specified with the (*isomin:isomax*). By default, levels are identified only by principal quantum number, with the maximum principal quantum number given by *nmax*. If *nmax* is not specified, the default is *nmax*=10. The maximum principal quantum number allowed is *nmax*=100.

# Export Controlled Information

Orbital angular momentum quantum numbers and/or differing values of *nmax* may be specified by isosequence through the use of the **isorange** command.

If *reconst* is specified, the atomic model will be written out to a text file. The file name will have the same base as the problem name and the suffix *.zxx*, where *xx* is the atomic index *iz*.

If *binary*, *pdb*, or *hdf5* is specified, the atomic model will be written out to a binary, PDB or HDF5 file, respectively. The file name will have the same base name as the original ascii datafile and the suffix *.adf*. The binary atomic datafile can then be used in place of the original datafile.

```
isorange iso1 iso2    nmax [lmax n_val nhole nhole_tot nx_val nx_tot n_n2n n_term nx nx_k nx_l]
isorange all           nmax [lmax n_val nhole nhole_tot nx_val nx_0 n_n2n n_term nx nx_k nx_l]
```

Specify the maximum principal quantum number *nmax* and orbital angular quantum number *lmax* to be used for isosequences from *iso1* to *iso2* when constructing a screened hydrogenic model. This command applies to the screened hydrogenic model specified with the last **atoms** command. If **all** is specified, these values will be applied to all isosequences in the atomic model. The maximum orbital angular momentum quantum number allowed is *lmax*=5. For hydrogen (Z=1) only, the maximum is *lmax*=14.

The optional arguments *n\_val*, *n\_hole* and *n\_hole\_tot* control the generation of states with inner shell holes for this range of isosequences and override the values of switch(126) and switch(127). Inner shell holes will be allowed down to a principal quantum number which is *n\_hole\_tot* below that of the ground configuration, with excitations up to a principal quantum number of *n\_val*. The model will explicitly include those states with inner shell holes down to a principal quantum number which is *n\_hole* below that of the ground configuration, with the default value of 0. The remaining states will be included implicitly through rates in an **augis** section.

The optional argument *nx\_val* controls the generation of multiply excited states. Multiply excited states will be generated based on the ground configuration with excitations up to a principal quantum number of *nx\_val*, with the default value of *nx\_val* given by switch(128).

The optional argument *nx\_0* is not used but must be present if the following arguments are present.

The optional argument *n\_n2n* controls the generation of transitions with  $\Delta n = 0$ . These transitions are automatically included for principal quantum numbers  $n \leq (l_{max} + 1)$ . Transitions will be generated in an **augis** section for all ground configurations and singly-excited states with excitations up to a principal quantum number of *n\_n2n*, with the default value of *n\_n2n* given by switch(130).

The optional argument *n\_term* controls term splitting. Transitions specified by switch(168) will be split up to the maximum principal quantum number *n\_term*, with the default value of *n\_term* given by switch(160).

The optional arguments *nx*, *nx\_k*, and *nx\_l* control the maximum number of excitations allowed, with *nx* specifying the maximum total number of excitations, with default value given by switch(187), *nx\_k* specifying the maximum number of excitations from the K-shell, and *nx\_l* specifying the maximum number of excitations from the L-shell, with the default values specified by switch(188).

```
modeltype type1 type2 ...
```

Specify attributes *type1*, *type2*, ... of the screened hydrogenic model specified by the last **atoms** command.

If *type*="fly", He-like and Li-like atomic structure and rates will match those used by FLY (R.W. Lee and J.T. Larsen, JQSRT **56**, p. 535, 1996).

## Export Controlled Information

If *type*=”term”, photoexcitations will be split into multiple transitions between terms with transitions energies and oscillator strengths coming from tabulated data in the file *termxx.dat*, where *xx* is the atomic number of the element (02-99). This file must be available at generation time in either the working directory, the directory specified by the environment variable CRDATA or the directory specified by AFILEDIR.

If *type*=”dca”, the atomic structure and rates will closely match those produced by DCA. This attribute is equivalent to the following command:

```
isorange all 10 0 5 4 4 10 10 6 6 2 1 1
```

which is the same as the default atomic structure.

Setting *type*=”dca” sets **switch**(168)=11002 and **switch**(188)=21.

Setting *type*=”dca\_v18” produces a model identical to that produced by code version 2.18 with modeltype “dca”.

If *type*=”radonly”, only photoexcitation transitions will be generated. The resulting model will be suitable for use as sublevel information.

If *type*=”sublevel” and Z=1, the model will be produced with principal quantum numbers only, along with sublevel information up to the specified *lmax*.

If *type*=”johnson” and Z=1, transition data will be constructed using formulas from L.C. Johnson, Ap. J. 174, p. 227, 1972.

### **atoms+** *filename*

Specify an atomic datafile *filename* which contains information to be added to that from the file specified in the previous **atoms** command. The resulting atomic model will contain data from the combined files. Only one **atoms+** command may be used with each **atoms** command.

### **subatoms** *filename*

Specify an atomic datafile *filename* which contains information to be added to be used as sublevel information with the data from the file specified in the previous **atoms** command. The data in *filename* should be in the standard atomic model format with level and transition specifications for a set of isosequences contained in the primary atomic model. All levels will be treated as sublevels of the primary level with the same isosequence and principal quantum number. All transitions will be treated as sublevel transitions. Only one **subatoms** command may be used with each **atoms** command, and may not be used together with an **atoms+** command.

### **zwindow** *iz1 iz2*

For steady-state kinetics calculations, initialize the isosequence window for atomic model *iz1* with a steady-state calculation done with atomic model *iz2*.

### **region** [*ireg*=*ireg*] *ir1 ir2* [**lte**] *tev* [*tiv* [*trv*]] [**vel**=*vel*]

Nodes *ir1* to *ir2* will be initialized with electron temperature *tev*, ion temperature *tiv* and radiation temperature *trv* (in eV). If *tiv* (*trv*) is absent, *tev* will also be used for the ion (radiation) temperature. The range of nodes *ir1* to *ir2* will apply to all commands from this section (except **atoms**) which do not specify a range of nodes or regions. The optional **ireg** argument can be used to assign region number *ireg* to these

# Export Controlled Information

nodes. If this argument is absent, the nodes will be assigned the next consecutive region number. If **lte** is specified, all material in this region will be treated as being in LTE. The optional **vel** argument can be used to assign velocities to these nodes

**regionkl** [/**ireg**=*ireg*] *k1 k2 l1 l2* [/**lte**] *tev [tiv [trv]]*

Same as **region** but applying to nodes within the 2-d logical region bounded by (*k1,l1*) and (*k2,l2*)

**regionklm** [/**ireg**=*ireg*] *k1 k2 l1 l2 m1 m2* [/**lte**] *tev [tiv [trv]]*

Same as **region** but applying to nodes within the 3-d logical region bounded by (*k1,l1,m1*) and (*k2,l2,m2*)

**reg** *ireg* [/**lte**] *tev [tiv [trv]]*

Same as **region** but applying to nodes within region *ireg*. The region index *ireg* can be assigned through a **regmap** section in an xfile, a **regmap** command, or is assigned consecutively to each **region**, **regionkl**, or **regionklm** command.

**regmult** [/**ireg**=*ireg*] *type value*

Multiply regional quantities of type *type* by *value*. *Type* may be "absn", "emis", "emate", "emati", "cve", "cvi", "pmate", "pmat", "esrcc", "esrci", "econd", "bcond", "cei", "cee", "cij", "eint" and "cspd". These multipliers are used for quantities which are calculated internally rather than specified and apply to nodes defined with the current region, or to nodes in region *ireg*, if the optional **ireg** argument is specified.

**regmap** *ireg1 ireg2 ireg3 ireg4 ...*

Assign region numbers *ireg1*, *ireg2*, *ireg3*, *ireg4*, ... to successive nodes in the problem. The node count starts at 1 with the first **regmap** command and continues consecutively through each **regmap** command. This is much the same as the **regmap** section in an xfile, except each line is prefaced with **regmap**.

**rho** *rho*

Assign total mass density *rho* (g/cm<sup>3</sup>) to the material in this region. If this command is present, elements and background materials specified with **element**, **level**, **material**, or **background** commands will be normalized to produce the specified mass density.

**element** *iz ytot [iso1 iso2] [/lte] [/te=te] [/ti=ti] [/vi=vi]*

Assign element *iz* an initial ion population of *ytot* (1/cm<sup>3</sup>). Multiple **element** commands can be used in the same region. If *iso1* and *iso2* are present, they specify the range of isoelectronic sequences which will be used for the element *iz* in this region (the default is all isoelectronic sequences). If **lte** is present, this element will be calculated assuming LTE conditions. This command cannot be used in the same region with **level** commands. The optional **te**, **ti**, and **vi** arguments can be used to assign electron temperatures, ion temperatures, and ion velocities to electron and/or ion distributions associated with this element.

**level** *iz iso i y [iso1 iso2]*

Assign the state of element *iz* with (isoelectronic sequence, level) (*iso,i*) an initial ion population of *y* (1/cm<sup>3</sup>). If *iso1* and *iso2* are present, they specify the range of isoelectronic sequences which will be used for the element *iz* in this region (the default is all isoelectronic sequences). Multiple **level** commands can be used in the same region. This command cannot be used in the same region with **element** commands.

# Export Controlled Information

## **material** *rho ab zb z2b*

Define a background mass density of *rho* ( $\text{g}/\text{cm}^3$ ) with average atomic number *ab*, average charge *zb*, and average squared charge *z2b*. This is equivalent to a **background** command, specifying mass density rather than ion density.

## **background** *yb neb ab zb z2b*

Define a background ion density of *yb* ( $1/\text{cm}^3$ ), electron density of *neb* ( $1/\text{cm}^3$ ). The background ions have average atomic number *ab*, average charge *zb*, and average squared charge *z2b*. This is equivalent to a **material** command, specifying ion density rather than mass density.

## **ne** *ne\_value*

Set the electron density to *ne\_value* ( $1/\text{cm}^3$ ). This will include any background electron density, thermal and non-thermal distributions. This value can be used for initialization purposes or can be fixed for the duration of the run.

## **nehot** *value1 value2 value3 ...*

Set the electron density for additional thermal (or monoenergetic) electron distributions to *value1*, *value2*, *value3* ... ( $1/\text{cm}^3$ ). These values must be accompanied by corresponding temperatures (or energies) specified with **tehot** commands.

## **tehot** *value1 value2 value3 ...*

Set the electron temperatures (or energies) for additional electron distributions to *value1*, *value2*, *value3* ... ( $1/\text{cm}^3$ ). These values must be accompanied by corresponding number densities specified with **nehot** commands.

## **econd** *value*

Set the thermal conduction coefficient to *value* ( $\text{cm}^2/\text{s}$ ).

## **bfield** *bfield [bx by bz]*

Set the magnetic field value to *bfield* (G). If any of *bx*, *by*, *bz* are specified, these will be used as the magnetic field components and the magnitude will be set by the components, i.e. *bfield* will be ignored.

## **opacity** *file filename*

Specify a file to use for LTE opacities and thermodynamic information for all zones in the current region. The file must be present in either the working directory or the directory specified by the environment variable OFILEDIR. The energy group structure must be the same as for the current run. This form of the command is for opacity files generated by CRETIN using the **opacity** runtime option.

## **opacity** *server filename*

Specify a file to use for LTE opacities for all zones in the current region. The file must be present in either the working directory or the directory specified by the environment variable OFILEDIR. The energy group structure must be the same as for the current run. This form of the command is for opacity files generated by the opacity server.

If the file is not present, the opacity server will generate a file. In this case, the composition of the region

# Export Controlled Information

must be specified with **atoms** and **element** commands. The model type may be specified as “null” to avoid generating atomic data and populations.

**opacity henke**  
**opacity cold**  
**opacity exro**

Specify that LTE opacities for all zones in the current region will come from the Henke tables (ref. B. L. Henke, E. M. Gullikson, and J. C. Davis, Atomic Data and Nuclear Data Tables Vol. 54 No. 2 July 1993).

Data files for all elements in the current run will be read at generation time. The files (name *XX.nff* where *XX* is the elemental symbol) must be present in either the working directory or the directory specified by the environment variable CRDATA. The composition of the region must be specified with **atoms** and **element** commands. The model type may be specified as “null” to avoid generating atomic data and populations.

The Henke opacities may also be accessed for comparison purposes through the **opac\_hk**, **absn\_hk**, **opac\_sp\_hk**, and **kappa\_sp\_hk** edits. When accessed in this manner, the opacities need not be used in the simulation.

Data files are available at [http://henke.lbl.gov/optical\\_constants](http://henke.lbl.gov/optical_constants).

**opacity** *type form p1 p2 p3 p4 e1 e2*  
**opacity spectral** *type form p1 p2 p3 p4 e1 e2*

Set the frequency-dependent absorption coefficient (in cm<sup>-1</sup>) according to the formula specified by *type* and *form*, using the parameters *p1-p4*. This specification holds for photon energies (in eV) between *e1* and *e2*, or for all energies if *e1* and *e2* are absent. *Type* may be “kinetics” or “analytic”. If *type* is “kinetics”, the absorption coefficients calculated by the atomic kinetics will be used. This is the default behavior.

If **spectral** is specified, the opacities will be used for spectral radiation transport. Otherwise, the opacities will be used for continuum radiation transport

If *type* is “analytic”, *form* may be “constant”, “power-law”, “exponential”, “gaussian”, or “cosine”. In addition, if *type* has “+” appended, the absorption coefficient will include a correction for stimulated emission.

If *form* is “constant”, the absorption coefficient value is given by *p1*.

If *form* is “power-law”, the absorption coefficient value is given by

$$p1 \rho^{p2} T^{p3} e^{p4}$$

If *form* is “exponential”, the absorption coefficient value is given by

$$p1 \rho^{p2} T^{p3} \exp\{ -(e-p4)/(p5) \}$$

If *form* is “gaussian”, the absorption coefficient value is given by

$$p1 \rho^{p2} T^{p3} \exp\{ -[(e-p4)/(p5)]^2 \}$$

If *form* is “cosine”, the absorption coefficient value is given by

$$p1 \rho^{p2} T^{p3} \cos[(e-p4)/(p5)]$$

# Export Controlled Information

If *form* is "cutoff", the absorption coefficient value is given by

$$\begin{array}{ll} p1 \rho^{p2} T^{p3} e^{p4} & e < p6 \\ p1 \rho^{p2} T^{p3} e^{p4} \cdot p5 & e \geq p6 \end{array}$$

where  $\rho$  is the mass density (in g/cm<sup>3</sup>),  $T$  is the electron temperature (in eV), and  $e$  is the photon energy (in eV).

**eos type**

**eos leos leosnum**

Specify the equation of state (EOS) treatment for the current region. This affects both the electron and ion energy densities and pressures, plus derivatives with respect to temperature and density.

If *type* is "gas", both the electrons and ions will be treated as non-interacting ideal gases. The electrons may be degenerate, according to the value of switch(151). There will be no contribution from internal energy.

If *type* is "kinetics", the EOS calculated by the atomic kinetics will be used, including ions treated as an ideal gas. This is the default EOS treatment.

If *type* is "cowan", the electron EOS will come from the atomic kinetics while the ion EOS will include liquid and solid phases as done in QEOS (ref: More, Warren, Young and Zimmerman, Phys. Fluids 31, 3059, 1988).

If *type* is "leos", the electron and ion EOS will come from LEOS material *leosnum*.

**eosfile filename**

Specify that the EOS information for the current region should be read from file *filename*. This command currently applies only to regions using LEOS and is not required to use the default LEOS datafile.

**esource esrce**

Set the free electron heating rate to *esrce* (erg/cm<sup>3</sup>/sec).

**esourci esrci**

Set the ion heating rate to *esrci* (erg/cm<sup>3</sup>/sec).

**qstart**

Set all the zones in the current region to use a quiet start for hydrodynamics, which requires the density to change sufficiently (fractional change = param(18)) or the ion temperature to increase sufficiently (fractional change = param(19)) before hydrodynamic motion will be allowed.

**vacuum**

Treat the current region as a vacuum. This sets **regmult** values for "emate" and "emati" to one, values for "cve" and "cvi" to a large value, and the remaining values to zero.

# Export Controlled Information

**scale** *type* [*func*] *ir1 ir2 a0 a1 r1 dr1 b1 c1 a2 r2 dr2 b2 ...*

Scale quantities specified by *type* from nodes *ir1* to *ir2* with position by the factor

$$a_0 + \sum a_i f(r_i, dr_i, b_i, c_i)$$

where

$$f(r_i, dr_i, b_i, c_i) = [(r-r_i)/(dr_i)] b_i \quad \text{if } c_i = 0$$

or

$$f(r_i, dr_i, b_i, c_i) = \exp \{ c_i [(r-r_i)/(dr_i)] b_i \} \quad \text{if } c_i \neq 0$$

*Type* may be "density", "temperature", "te", "ti", "tr", "velocity", or "current". A type of "density" will scale all mass and number densities. A type of "temperature" will scale electron, ion and radiation temperatures. This command will be overridden by use of special profiles – see **switch 85**.

A single elementary function *func* may be applied to the resulting factor. Available values for *func* are "inverse", "absn", "exp", "ln", "log", and "sqrt".

This command does not need to be associated with a **region** command.

**sgauss** *type* [*func*] *ir1 ir2 a0 a1 r1 dr1 b1*

Scale quantities specified by *type* from nodes *ir1* to *ir2* with position by the factor

$$a_0 + a_1 \exp \{ -[(r-r_i)/(dr_i)]^2 \}$$

Other arguments and behavior are the same as for the **scale** command.

**stanh** *type* [*func*] *ir1 ir2 a0 a1 r1 dr1 b1*

Scale quantities specified by *type* from nodes *ir1* to *ir2* with position by the factor

$$a_0 + (a_1/2) \{ 1 - \tanh [(r-r_i)/(dr_i)] \}$$

Other arguments and behavior are the same as for the **scale** command.

**scalekl** *type* [*func*] *k1 k2 l1 l2 a0 a1 x1 dx1 bx1 cx1 y1 dy1 by1 cy1 a2 x2 dx2 bx2 cx2 y2 dy2 by2 cy2 ...*

Scale quantities specified by *type* for nodes within the 2-d logical region bounded by (*k1,l1*) and (*k2,l2*) by the factor

$$a_0 + \sum a_i f(x_i, dx_i, bx_i, cx_i) f(y_i, dy_i, by_i, cy_i)$$

where  $f(x, dx, b, c)$  is as described for the **scale** command.

*Type* may be "density", "temperature", "te", "ti", or "tr". Other arguments and behavior are the same as for the **scale** command. This command does not need to be associated with a **regionkl** command.

**scaleklr** *type* [*func*] *k1 k2 l1 l2 a0 a1 r1 dr1 b1 c1 ...*

Scale quantities specified by *type* for nodes within the 2-d logical region bounded by (*k1,l1*) and (*k2,l2*) by the factor

$$a_0 + \sum a_i f(r_i, dr_i, b_i, c_i)$$

# Export Controlled Information

where  $r_i = (x_i^2 + y_i^2)^{1/2}$  and  $f(r, dr, b, c)$  is as described for the **scale** command.

*Type* may be "density", "temperature", "te", "ti", or "tr". Other arguments and behavior are the same as for the **scale** command. This command does not need to be associated with a **regionkl** command.

**scaleklm** *type* [*func*] *k1 k2 l1 l2 m1 m2 a0 a1 x1 dx1 bx1 cx1 y1 dy1 by1 cy1 z1 dz1 bz1 cz1 ...*

Scale quantities specified by *type* for nodes within the 3-d logical region bounded by (*k1,l1,m1*) and (*k2,l2,m2*) by the factor

$$a_0 + \sum a_i f(x_i, dx_i, bx_i, cx_i) f(y_i, dy_i, by_i, cy_i) f(z_i, dz_i, bz_i, cz_i)$$

where  $f(x, dx, b, c)$  is as described for the **scale** command.

*Type* may be "density", "temperature", "te", "ti", or "tr". Other arguments and behavior are the same as for the **scale** command. This command does not need to be associated with a **regionklm** command.

**scaleklmr** *type* [*func*] *k1 k2 l1 l2 a0 a1 r1 dr1 b1 c1 ...*

Scale quantities specified by *type* for nodes within the 3-d logical region bounded by (*k1,l1,m1*) and (*k2,l2,m2*) by the factor

$$a_0 + \sum a_i f(r_i, dr_i, b_i, c_i)$$

where  $r_i = (x_i^2 + y_i^2 + z_i^2)^{1/2}$  and  $f(r, dr, b, c)$  is as described for the **scale** command.

*Type* may be "density", "temperature", "te", "ti", or "tr". Other arguments and behavior are the same as for the **scale** command. This command does not need to be associated with a **regionklm** command.

**tprofile** [*ireg*] *type* [*func1 func2 ...*] *time1 time2 a0 a1 t1 dt1 b1 c1 a2 t2 dt2 b2 c2 ...*  
**tprofile zonal** *type* [*func1 func2 ...*] *time1 time2 [a0 a1 t1 dt1 b1 c1 a2 t2 dt2 b2 c2 ...]*  
**tprofile from** *ix ic1 ic2*

Scale quantities specified by *type* [for nodes in region *ireg*] with time by the factor

$$a_0 + \sum a_i f(t_i, dt_i, b_i, c_i)$$

where

$$f(t_i, dt_i, b_i, c_i) = [(time - t_i)/(dt_i)]^{b_i} \quad \text{if } c_i = 0$$

or

$$f(t_i, dt_i, b_i, c_i) = \exp \{ c_i [(time - t_i)/(dt_i)]^{b_i} \} \quad \text{if } c_i \neq 0$$

*Type* may be "density", "temperature", "te", "ti", or "tr". A type of "density" will scale all mass and number densities. A type of "temperature" will scale electron, ion and radiation temperatures. If no region is specified, the scaling applies to all regions.

Multiple elementary functions *func<sub>i</sub>* may be applied to the resulting factor. Available values for *func<sub>i</sub>* are "inverse", "absn", "exp", "ln", "log", or "sqrt".

If **zonal** is specified, a separate set of coefficients *a<sub>i</sub>, t<sub>i</sub>, dt<sub>i</sub>, b<sub>i</sub>, and c<sub>i</sub>* will be used for each problem zone. The coefficients may be contained in the "coeff" section of an xfile. This command must be followed by a **tprofile from** command, specifying the xfile index *ix* and the first and last coefficient indices from this xfile. Values corresponding to coeff *ic1* will be assigned to *a<sub>0</sub>*, *ic1+1* to *a<sub>1</sub>*, *ic1+2* to *t<sub>1</sub>*, and so on through

# Export Controlled Information

coeff *ic2*. Coefficients not found in the xfile will be assigned the value contained in the argument list, or zero if no corresponding argument is present.

## Mesh:

If a node-centered mesh is used (switch 42 = 0), the nodes specified in the following commands should cover the entire range (1, nr) or (1, kmax) and (1, lmax). If a zone-centered mesh is used, they should cover the range (2, nr-1) or (2, kmax-1) and (2, lmax-1). CRETIN will add boundary nodes to fill out the ranges. The nodes specified in the **region**, **regionkl** and **regionklm** commands should always cover the entire range.

Equivalents of the 1-d zoning commands **r**, **rlin**, **rlog**, **rgeom**, and **rexp** may also be used for zoning simple 2-d and 3-d meshes in which the x-, y-, and z-values are the same along each k-, l-, and m-line, respectively. The equivalent commands begin with **x**, **y**, and **z** as documented below and must be combined with a **prodmesh** command.

### **geometry type**

Specify geometry for transfer and thermal conduction calculations. Options for *type* include ‘none’ in 0-d, ‘plane’ (or ‘slab’), ‘cylinder’, ‘sphere’ and ‘wedge’ in 1-d, ‘xy’ and ‘rz’ in 2-d, and ‘xyz’ in 3-d. If no **geometry** command is present, the default geometry will be ‘plane’ if an **r**, **rlin**, **rlog**, or **rgeom** command is present, ‘xy’ if a **quad**, **quadb** or **wedge** command is present, ‘xyz’ if a **hex** or **cone** command is present, and ‘none’, otherwise.

```
r r1 r2 r3 ...
x x1 x2 x3 ...
y y1 y2 y3 ...
z z1 z2 z3 ...
```

Specify positions of nodes. Multiple commands may be used to specify the positions of nodes, which are numbered consecutively in order of appearance.

```
rlin ir1 ir2 rmin rmax
xlin k1 k2 xmin xmax
ylin l1 l2 ymin ymax
zlin m1 m2 zmin zmax
```

Linearly interpolate positions between *rmin* at node *ir1* and *rmax* at node *ir2*.

```
rlog ir1 ir2 rmin rmax ratio
xlog k1 k2 xmin xmax ratio
ylog l1 l2 ymin ymax ratio
zlog m1 m2 zmin zmax ratio
```

Equal ratio zoning between *rmin* at node *ir1* and *rmax* at node *ir2* with ratio *ratio*.

```
rgeom ir1 ir2 rmin rmax drmin slope
xgeom k1 k2 xmin xmax dxmin slope
ygeom l1 l2 ymin ymax dymin slope
zgeom m1 m2 zmin zmax dzmin slope
```

Equal ratio zoning between *rmin* at node *ir1* and *rmax* at node *ir2* with smallest zone size *drmin*. Zone widths increase from *ir1* to *ir2* if *slope* > 0, decrease from *ir1* to *ir2* if *slope* < 0.

# Export Controlled Information

```
rexp ir1 ir2 rmin rmax scale  
xexp k1 k2 xmin xmax scale  
yexp l1 l2 ymin ymax scale  
zexp m1 m2 zmin zmax scale
```

Exponential zoning between  $rmin$  at node  $ir1$  and  $rmax$  at node  $ir2$  with e-folding length  $scale$ . Zone widths increase from  $ir1$  to  $ir2$  if  $scale > 0$ , decrease from  $ir1$  to  $ir2$  if  $scale < 0$ .

## product\_mesh

Produce a 2-d or 3-d mesh from the x-, y-, and z-positions specified with 1-d meshing commands.

```
quad k1 k2 l1 l2 xmin xmax ymin ymax ratiok ratiol
```

Equal ratio zoning between  $(xmin,ymin)$  at node  $(k1,l1)$  and  $(xmax,ymax)$  at node  $(k2,l2)$  with ratios  $(ratiok,ratiol)$ .

```
quadb k1 k2 l1 l2 x1 y1 x2 y2 x3 y3 x4 y4 ratiok ratiol
```

Equal ratio zoning between  $(x1,y1)$  at node  $(k1,l1)$ ,  $(x2,y2)$  at node  $(k2,l1)$ ,  $(x3,y3)$  at node  $(k2,l2)$ , and  $(x4,y4)$  at node  $(k1,l2)$  with ratios  $(ratiok,ratiol)$ .

```
wedgek k1 k2 l1 l2 rmin rmax phimin phimax ratiok ratiol [x0 y0]  
wedgel k1 k2 l1 l2 rmin rmax phimin phimax ratiok ratiol [x0 y0]
```

Equal ratio zoning of a wedge between  $(rmin,phimin)$  at node  $(k1,l1)$  and  $(rmax,phimax)$  at node  $(k2,l2)$  with ratios  $(ratiok,ratiol)$ . The angles  $phimin$  and  $phimax$  are measured in degrees clockwise from the +y-axis. The wedge will be centered at the origin, or at  $(x0,y0)$  if these coordinates are specified. If **wedgek** is specified, the radial lines will have increasing  $k$  (lines of constant  $l$ ). If **wedgel** is specified, the radial lines will be have increasing  $l$  (lines of constant  $k$ ).

```
hex k1 k2 l1 l2 m1 m2 xmin xmax ymin ymax zmin zmax ratiok ratiol ratiom
```

Equal ratio zoning between  $(xmin,ymin,zmin)$  at node  $(k1,l1,m1)$  and  $(xmax,ymax,zmax)$  at node  $(k2,l2,m2)$  with ratios  $(ratiok,ratiol,ratiom)$ .

```
cone k1 k2 l1 l2 m1 m2 rmin rmax phimin phimax thetamin thetamax  
ratiok ratiol ratiom [x0 y0 z0]
```

Equal ratio zoning of a portion of a sphere (not really a cone) between  $(rmin,phimin,thetamin)$  at node  $(k1,l1,m1)$  and  $(rmax,phimax,thetamax)$  at node  $(k2,l2,m2)$  with ratios  $(ratiok,ratiol,ratiom)$ . The angles  $thetamin$  and  $thetamax$  are measured in degrees clockwise from the +z-axis. The angles  $phimin$  and  $phimax$  are measured in degrees clockwise from the +y-axis. The cone will be centered at the origin, or at  $(x0,y0,z0)$  if these coordinates are specified.

```
velocity ir1 ir2 vel
```

Nodes  $ir1$  to  $ir2$  will be initialized with 1-d velocity  $vel$ .

```
dvdr ir1 ir2 delv
```

Nodes  $ir1$  to  $ir2$  will be initialized with 1-d velocity gradient  $delv$ .

# Export Controlled Information

**u2d**  $k1 k2 l1 l2 uvalue$

**v2d**  $k1 k2 l1 l2 vvalue$

**w2d**  $k1 k2 l1 l2 wvalue$

Nodes within the 2-d logical region bounded by  $(k1,l1)$  and  $(k2,l2)$  will be initialized with velocity x- (or r-) component *uvalue*, y- (or z-) component *vvalue*, or z- (or phi-) component *wvalue*, for xy (or rz) geometry.

**u3d**  $k1 k2 l1 l2 m1 m2 uvalue$

**v3d**  $k1 k2 l1 l2 m1 m2 vvalue$

**w3d**  $k1 k2 l1 l2 m1 m2 wvalue$

Same as **u2d**, **v2d**, **w3d** but applying to nodes within the 3-d logical region bounded by  $(k1,l1,m1)$  and  $(k2,l2,m2)$

## Continuum radiation:

**frequency**  $f1 f2 f3 \dots$

Define group structure for continuum radiation in terms of frequencies (Hz). The lowest group will extend from the lowest previously defined group boundary (or zero) to  $f1$ , the next group will extend from  $f1$  to  $f2$ , etc. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **frequency** commands as necessary to define all the groups.

**energy**  $e1 e2 e3 \dots$

Define group structure for continuum radiation in terms of energies (eV). The lowest group will extend from the lowest previously defined group boundary (or zero) to  $e1$ , the next group will extend from  $e1$  to  $e2$ , etc. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **energy** commands as necessary to define all the groups.

**energy0**  $e0$

Set lower boundary of the lowest continuum radiation group to  $e0$ .

**wavelength**  $w1 w2 w3 \dots$

Define group structure for continuum radiation in terms of wavelengths (1/cm). The lowest group will extend from the lowest previously defined group boundary (or zero) to  $w1$ , the next group will extend from  $w1$  to  $w2$ , etc. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **wavelength** commands as necessary to define all the groups.

**fbins**  $n f1 fn ratio$

Define group structure for continuum radiation in terms of frequencies (Hz). This command will add  $n-1$  groups ( $n$  group boundaries) extending from  $e1$  to  $en$  with successive group widths increasing by *ratio*. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **fbins** commands as necessary to define all the groups. These may be combined with **frequency** commands.

**ebins**  $n e1 en [ratio]$

Define group structure for continuum radiation in terms of energies (eV). This command will add  $n-1$

## Export Controlled Information

groups ( $n$  group boundaries) extending from  $e1$  to  $en$  with successive group widths increasing by  $ratio$ . If  $ratio$  is absent (or zero), the group boundaries will be logarithmically spaced. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **ebins** commands as necessary to define all the groups. These may be combined with **energy** commands.

**ebins2**  $n$   $e1$   $en$  [ $ratio$ ]

Define a secondary group structure for radiation in terms of energies (eV). This group structure may only be defined when calculating an LTE opacity table. The primary group structure is used for the kinetics calculations and the resulting opacities are averaged onto the secondary group structure for the table.

**wbins**  $n$   $w1$   $wn$  [ $ratio$ ]

Define group structure for continuum radiation in terms of wavelengths (1/cm). This command will add  $n-1$  groups ( $n$  group boundaries) extending from  $w1$  to  $wn$  with successive group widths increasing by  $ratio$ . If  $ratio$  is absent (or zero), the group boundaries will be logarithmically spaced. Transfer will be done at the group centers, evenly spaced between the group boundaries. There may be as many **wbins** commands as necessary to define all the groups. These may be combined with **wavelength** commands.

The following commands specify group boundaries in units of keV. These duplicate commands are used in other codes and may be used to ensure identical group structures to those codes.

**pbins**  $n1$   $n2$   $e1$   $e2$   
**rbins**  $n1$   $n2$   $e1$   $e2$

Define group structure for continuum radiation in terms of energies (keV). This command sets group boundary values for groups  $n1$  to  $n2$  to extend from  $e1$  to  $e2$  with successive group widths logarithmically spaced. The lower group boundary of the first group corresponds to  $n = 0$ .

**pbins2**  $n1$   $n2$   $e1$   $e2$   
**rbins2**  $n1$   $n2$   $e1$   $e2$

Define a secondary group structure for radiation in terms of energies (keV). See the discussion of **ebins2** for restrictions on the use of these commands.

**pbinlin**  $n1$   $n2$   $e1$   $e2$   
**rbinlin**  $n1$   $n2$   $e1$   $e2$

Define group structure for continuum radiation in terms of energies (keV). This command sets group boundary values for groups  $n1$  to  $n2$  to extend from  $e1$  to  $e2$  with equal group widths. The lower group boundary of the first group corresponds to  $n = 0$ .

**angles**  $nmu$  [ $nphi$ ]

Number of angles used to set rays for transfer problems. In planar geometry,  $nmu$  is the number of rays. In cylindrical geometry,  $nmu$  is the number of angles in the theta direction and rays in the phi direction are spaced no further apart than  $\pi/(2 nphi)$ . In spherical geometry, rays are spaced no further apart in mu than  $1/nmu$ .

**opacfile** *filename*

Specify the name of the opacity / linear response file to use for the current run. This file will be used as the source of opacities for those regions which are not using kinetics or analytic forms for opacities.

## Export Controlled Information

**xfilebc** *ix mu phi multiplier isotropy [/kl kmin kmax lmin lmax mmin mmax]*  
**xfilebc** *ix mu phi multiplier isotropy [/xy xmin xmax ymin ymax zmin zmax]*

Assign boundary conditions to the radiation densities (*pbins*) in xfile *ix*. The incident radiation is assumed to be streaming in the direction (*mu,phi*) if *isotropy*=0, or is assumed to be isotropic if *isotropy* is non-zero. The radiation densities in the xfile will be multiplied by *multiplier* for this boundary condition only. The same xfile may be used for multiple boundary conditions.

For 1-d geometries, boundary conditions with *mu*>0 are interpreted as streaming in the direction of decreasing r and are applied at maximum r while boundary conditions with *mu*<0 are applied at minimum r.

For 2-d and 3-d geometries, the boundary conditions will be applied to the mesh boundary lying within (*k1,l1,m1*) and (*k2,l2,m2*) if **kl** is specified, or within (*x1,y1,z1*) and (*x2,y2,z2*) if **xy** is specified.

NOTE: The preferred command for setting boundary conditions is **boundary**.

Directions (rays) are defined as follows:

For 1-d geometries,  $mu=\cos(\theta)$ , where  $\theta$  is the angle of the ray relative to the normal;  $mu>0$  indicates the direction of increasing r,  $mu<0$  indicates the direction of decreasing r. For cylindrical geometry, *phi* is the angle between the normal to the axis and the "x-axis" (chosen so that a ray through the origin has *phi*=0.) and runs from 0. to  $\pi/2$ .

For xy or rz geometry,  $mu=\cos(\theta)$ , where  $\theta$  is the angle between the ray and the z-axis, and *phi* is the angle between the ray and the y-axis, measured clockwise from the +y-axis. For xy geometry, *mu* runs from 0 to 1 and *phi* runs from 0 to  $2\pi$ . For rz geometry, *mu* runs from -1 to 1 (*mu*>0 indicates the direction of increasing z, *mu*<0 indicates the direction of increasing z) and *phi* runs from  $-\pi/2$  to  $\pi/2$ .

For 3-d (xyz) geometry,  $mu=\cos(\theta)$ , where  $\theta$  is the angle between the ray and the +z-axis, and *phi* is the angle between the ray and the y-axis, measured clockwise from the +y-axis. *mu* runs from -1 to 1 and *phi* runs from 0 to  $2\pi$ .

### Spectral radiation:

**spectrum** *n e1 en [ratio]*

Define spectral group structure for radiation in terms of energies (eV). This command will add *n* energies extending from *e1* to *en* with successive spacings increasing by *ratio*. If *ratio* is absent (or zero), the group boundaries will be logarithmically spaced. Each successive **spectrum** command will effectively add *n*-1 energies, replacing the last previously defined energy with *e1*. There may be as many **spectrum** commands as necessary to define all the spectral groups. The energies defined by successive **spectrum** commands must be monotonically increasing.

The inclusion of one or more **spectrum** commands will result in a detailed spectral calculation being done for each timestep for which edits are requested. If the geometry is 'none', spectral opacities and emissivities will be calculated but not intensities.

**spectrum resolution** [*tresolve nresolve*] [*iz*] [*iso1 iso2 iso3*]

The presence of the **resolution** option on a **spectrum** command results in energies being added to the

# Export Controlled Information

spectral group structure corresponding to the energy of each photoexcitation (which falls within the limits defined by other **spectrum** commands) from every atomic model. If *tresolve* and *nresolve* are present, energies will be added to resolve a doppler lineshape corresponding to a temperature of *tresolve* (eV), with *nresolve* energies on each side of line center. If *iz* is present, only photoexcitations from atomic model *iz* will be considered. If *iso1*, *iso2*, *iso3* are also present, only photoexcitations within the isoelectronic sequence range *iso1-iso3* will be included, while photoexcitations for isoelectronic above *iso2* will be included with *nresolve=0*.

This may eventually be expanded to give the user the capability of specifying the desired resolution of the spectrum in a more meaningful manner.

## **spectrum lines** [*tresolve nresolve iz iso1 iso2 iso3*]

The presence of the **lines** option on a **spectrum** command results in energies being added to the spectral group structure corresponding to the energy of each photoexcitation singled out with a **line** command (for which the energy falls within the limits defined by other **spectrum** commands). If *tresolve* and *nresolve* are present, energies will be added to resolve a doppler lineshape corresponding to a temperature of *tresolve* (eV), with *nresolve* energies on each side of line center. If both **lines** and **resolution** options are present, the maximum values of *tresolve* and *nresolve* will be used. If *iz* is present, only photoexcitations from atomic model *iz* will be considered. If *iso1*, *iso2*, *iso3* are also present, only photoexcitations within the isoelectronic sequence range *iso1-iso3* will be included, while photoexcitations for isoelectronic above *iso2* will be included with *nresolve=0*.

The energy corresponding to a particular line center can be specified in edits with a negative index, where the absolute value of the index gives the line index.

## **spectrum ebins**

Define the spectral group structure for radiation to be the same as the continuum group structure. This is mostly useful when using tabulated opacities. This command is not compatible with any other **spectrum** commands.

## **spectral-group** *is el en*

Define a spectral range (spectral-group) *is* which extends from *el* to *en* (in eV). Spectral edits can be restricted to a specific energy range by specifying a spectral range index. The full spectrum, defined by all **spectrum** commands, has spectral range index 0. The limits of the spectral range will actually be those energies from the full spectrum which are closest in value to *el* and *en*.

## **isofilter absorption** *iz iso1 iso2 [bb | bf | ff]*

## **isofilter emission** *iz iso1 iso2 [bb | bf | ff]*

## **isofilter both** *iz iso1 iso2 [bb | bf | ff]*

This command restricts the contributions to the spectral absorption and/or emission from atomic model *iz* to those transitions involving isosequences between *iso1* and *iso2*. Multiple **isofilter** commands may be used for each atomic model to specify all the isosequences desired. Isosequences which are not included in any **isofilter** command remain unaffected. Each **isofilter** command may have an optional argument specifying that the command applies only to bound-bound (**bb**), bound-free (**bf**) or free-free (**ff**) absorption and/or emission. Free-free absorption/emission is modified by including/excluding isosequence populations – the electron density is not affected. These options can be useful for investigating how different elements / isosequences contribute to spectral intensities. Contributions to spectral absorption coefficients and emissivities are available directly through various spectral edits.

For example, the emission from element *iz* can be restricted to bound-bound emission for isosequences

# Export Controlled Information

between  $iso1$  and  $iso2$  by

**isofilter emission iz**  $iso1\ iso2\ bb$   
**isofilter emission iz**  $0\ -1\ bf$   
**isofilter emission iz**  $0\ -1\ ff$

Lineshapes:

**stark transition iz**  $iso\ i1\ i2$   
**stark manifold iz**  $iso\ n1\ n2$

Define radiative transitions to be treated with the TOTAL code. If **transition** is specified, the single radiative transition connecting lower state  $(iso,i1)$  with upper state  $(iso,i2)$  in model *iz* will be treated. If **manifold** is specified, all radiative transitions connecting lower states with principal quantum number *n1* with upper states with principal quantum number *n2* within isosequence *iso* in model *iz* will be treated.

Line radiation:

**line iline iz**  $iso1\ i1\ iso2\ i2\ [de]$

Define a line with index *iline* as the radiative transition connecting lower state  $(iso1,i1)$  with upper state  $(iso2,i2)$  in model *iz*.

The optional argument *de* can be used to single out a particular transition when there are multiple transitions of the same type between the two specified levels. Including this argument specifies the transition with energy difference closest to *de*.

If *de* is not specified and there are multiple radiative transitions connecting these two states, each transition will be denoted as a line with index  $(1000*iline+i)$ , with *i* running from 0 to *n*-1 for *n* radiative transitions. Line identifications will appear in the *.tbl* file. Each of these lines will be treated with the options and bin structure specified for *iline*, and will be treated as interacting and overlapping where appropriate.

**linetype** *type1 type2 ...*

Define method(s) to use in treating the line defined with the preceding **line** command. Options for *type* include:

	formal	- formal transfer, Feautrier formalism
+	complin	- Rybicki linearization
	rybicki	- Rybicki linearization
	feautrier	- Feautrier linearization
	formald	- formal transfer, integral formalism
+	complind	- linearization, integral formalism
	approximate	- tridiagonal approximate operator
+	local	- diagonal approximate operator
	escape	- escape factor treatment
	none	- no line transfer treatment
*	nodoppler	- don't include bulk doppler shifts
	doppler	- include bulk doppler shifts
*	crd	- complete redistribution
	prd	- angle-averaged partial redistribution (RIIA)

# Export Controlled Information

r2a	- angle-averaged partial redistribution (RIIA)	
r2	- angle-dependent partial redistribution (RII)	
*	voigt	- lineshape assumed to be Voigt profile
	lorentzian	lineshape assumed to be Lorentzian profile
	gaussian	lineshape assumed to be Gaussian profile
	total	- lineshape calculated with TOTAL
	totalb	- lineshape calculated with TOTAL, including magnetic fields
	beline	lineshape obtained from BELINE tables

Default values are marked with a \* and may be changed with a **linedefault** command. No **linetype** command is necessary to use the default values. The values marked with a + are possible default values depending on geometry and doppler shifts. In 1-d, the default method is *complin* without doppler shifts and *complind* with doppler shifts. In 2-d and 3-d, only an integral formalism with a local approximate operator (*local*) is available.

**linedefault** *type1 type2 ...*

Define default method(s) to use in treating lines. Any valid options for **linetype** commands may be specified.

**lbins** *n1 del ratio1 n2 de2 ratio2 ...*

Define the bin structure of the line defined with the preceding **line** command. Starting from line center, there will be *n1* bins spanning an energy *del* with equal ratio widths using *ratio1*, followed by *n2* bins spanning an energy *de2* with equal ratio widths using *ratio2*, etc. The bin structure will be symmetric about the line center. If this line overlaps with other lines, a combined bin structure will be constructed using the individual line structures. In this case, the total number and positions of the final bins will not match those specified by the **lbins** command. There may be at most one **lbins** command for each line.

**resonance** *iline1 iline2 de*

Define lines *iline1* and *iline2* to be overlapping with the line center energy of *iline2* greater than that of *iline1* by *de* (in eV). For multiple overlapping lines, a series of **resonance** commands may be used, each with the same *iline1*.

**resonant** *iline1 iline2 iline3 ...*

Define lines *iline1*, *iline2*, *iline3*, ... to be overlapping with line center energies taken from the atomic models.

**joinline** *iline1 iline2 iline3 ...*

Specify that lines *iline1*, *iline2*, *iline3*, ... do not overlap in energy but should be transferred simultaneously.

**r2file** *filename* [**generate**]

Specify the data file *filename* containing RIIA table to use for PRD calculation. This table is required for either RIIA or RII redistribution. The default is "r2atable". If **generate** is specified, a file of this name will be generated with either specified or default values for Voigt parameters.

**rprd** *rmax nrprd*

## Export Controlled Information

Specify the values of  $r = \frac{1}{2}[x-x']$  to use in generating the RIIA table. There will be  $nrprd+1$  values, evenly spaced from 0. to  $rmax$ .

**sprd**  $smax\ nsprd$

Specify the values of  $s = \frac{1}{2}[x+x']$  to use in generating the RIIA table. There will be  $nsprd+1$  values, evenly spaced from 0. to  $smax$ .

**aprd**  $a1\ a2\ a3\ a4\ ...$

Specify the values of the Voigt parameters  $a1, a2, a3, a4, ...$  to use in generating the RIIA table. If 0. is not included in the list, it will be added. There may be multiple **aprd** commands.

### Electron distribution:

**eenergy**  $e1\ e2\ e3\ ...$

Define group structure for the electron distribution in terms of energies (eV). The lowest group will extend from the lowest previously defined group boundary to  $e1$ , the next group will extend from  $e1$  to  $e2$ , etc. There may be as many **eenergy** commands as necessary to define all the groups.

**eebins**  $n\ e1\ en\ [ratio]$

Define group structure for the electron distribution in terms of energies (eV). This command will add  $n-1$  groups ( $n$  group boundaries) extending from  $e1$  to  $en$  with successive group widths increasing by *ratio*. If *ratio* is absent (or zero), the group boundaries will be logarithmically spaced. There may be as many **eebins** commands as necessary to define all the groups. These may be combined with **eenergy** commands.

### Laser Raytrace:

**laser**  $ilas\ wvl/xn\ value/rate/integral\ history/constant\ id/value\ mul\ [ppol]$

Define an external laser identified by index *ilas* with wavelength *wvl* (in cm). The wavelength specification may also be in the form *xn*, in which case *n* will be interpreted as a multiplier on the reference wavelength of 1.06 μm. The laser power (in erg/sec) will be obtained from time history *id* (if **history** is specified) or from *value* (if **constant** is specified). The values will be interpolated to the current time if **value** is specified, or will be integrated (and divided by the timestep) to ensure conservation if **rate** is specified. If **integral** is specified, the values are assumed to already be a time integral of the desired quantity. An additional multiplier of value *mult* will be applied to the laser power. The p-polarization fraction *ppol* may also be specified.

**lasray**  $r0\ mu\ phi\ frac\ [res\_frac]$

Define a laser ray with entrance position *r0*, direction (*mu,phi*), and fractional power *frac* for the previously defined laser. The fractional power is relative to other rays defined for the same laser, as the power associated with each ray will be normalized so that the total for the laser is as specified on the **laser** command. If switch(47) = 0, *res\_frac* will be used as the resonance absorption fraction associated with that ray.

# Export Controlled Information

## Edits:

```
transition itr iz iso1 i1 iso2 i2 [de]
transition itr iz1 iso1 i1 iz2 iso2 i2 [de]
```

Define transition index *itr* for edit purposes. All edits having to do with a transition between states  $(iso1,i1)$  and  $(iso2,i2)$  in model *iz* will refer the index *itr*. The states must be specified with the lower energy state first and the higher energy state second. For edits which sum or average over states, a value of 0 for *i1* will produce an average over all states in isosequence *iso1*, while a value of 0 for *i2* will produce a sum over all states in isosequence *iso2*. For edits which run over multiple states (i.e. using “level” as an independent variable”), the appropriate *iso* and *i* should both be zero. Since  $(iso,i)=(0,0)$  is used for running over multiple states, it may not be used to sum or average over states within *iso*=0. Setting  $(iso,i)=(0,1)$  will produce the same results.

The optional argument *de* can be used to single out a particular transition when there are multiple transitions of the same type between the two specified levels. Including this argument specifies the transition with energy difference closest to *de*.

The second form must be used to specify charge exchange transitions between states in model *iz1* and states in model *iz2*. This is distinguished from the first form by the number of arguments, so additional arguments may not be added to these commands.

```
editray idir mu phi type [r]
```

Define the direction index *idir* for edit purposes. If *type*=0, the direction implied by *idir* will be the direction of that ray which is closest to  $(mu, phi)$ . If *type*>0, the direction will be that specified by  $(mu, phi)$ .

```
detector idtct x y [z] mu phi domega [nmu] [nphi]
detector idtct x y [z] mu phi domega extent dr [nr] [dx]
```

Define a detector index *idtct* for edit purposes (for 2-d and 3-d geometries only). The detector must be located outside the entire mesh, as it views boundary information only. The detector is located at position  $(x, y)$  for 2-d or  $(x,y,z)$  for 3-d, is oriented to capture photons headed in the direction specified by  $(mu, phi)$ , and subtends a solid angle *domega*. If *nmu* and *nphi* are present, the detector solid angle will be discretized with these numbers of angles; otherwise a single direction will be used. If **extent** is specified, the detector is assumed to extend a distance *dr* on either side of the midpoint  $(x, y)$  in the direction normal to  $(mu, phi)$  and the detector output is averaged over this distance. In this case, a single direction will be used to discretize the solid angle and *nr* segments will be used to discretize each side of the detector (with a default value of *nr*=1). For rz geometry, if *dx* is specified, the detector is assumed to extend a distance *dr* in the direction normal to  $(mu, phi)$  in the rz-plane and a distance *dx* in the perpendicular direction.

```
drat mu phi x0 y0 z0 xpmin xpmax ypmin ypmax nx ny [doppler] [info level]
```

Define a set of rays for spectral ray trace calculations, similar to those done by Drat or Hex, resulting in the intensity *inu* and optical depth *tau* as a function of photon energy for each ray. (These have the same units and meaning as *isp* and *tausp*, respectively). The reference ray passes through the reference point  $(x0)$  for 1-d,  $(x0,y0)$  for 2-d or  $(x0,y0,z0)$  for 3-d in the direction given by  $(mu,phi)$ . All three coordinates must be present for 1-d, 2-d or 3-d geometries; extraneous coordinate values will be ignored. The spectral rays are parallel to the reference ray, passing through a plane perpendicular to the reference ray, with *nx* rays uniformly arranged at positions  $[xpmin,xpmax]$  and *ny* rays uniformly arranged at positions  $[ypmin,ypmax]$  in this plane, with the xp-direction in the plane defined to be perpendicular to the z-axis. If the reference ray is perpendicular to the z-axis (*mu*=0), the (xp,yp) directions are the same as the (x,y) directions. In 1-d cylindrical or 2-d xy-geometry, *ny* corresponds to an ignorable coordinate and will be set to 1, but values

# Export Controlled Information

for *ypmin*, *ypmax*, and *ny* must be present. If **doppler** is specified, then doppler shifting due to material velocities will be included in the ray trace calculations.

The results will be put in a separate PDB or HDF5 output file with extension “drt”. For each **drat** command, results will include the intensity *inu* and optical depth *tau*, as a function of photon energy *evsp* and pixel. If **info** and *ilev* $\neq$ 0 are specified, raytrace information for the ray corresponding to the central pixel will also be included. The raytrace information will include the intensity *inuray*, optical depth *tauray*, distance along the ray *sray*, and coordinates along the ray (e.g. *xray*, *yray*, *zray*), plus material information (temperatures and number densities). If more than one **drat** command was specified, the result labels will be include *\_n*, where the integer *n* denotes the *n*<sup>th</sup> **drat** command.

Raytrace information along additional rays will also be included in the output file according to the value of *level*. A value of *level* $=\pm 1$  provides ray trace information along the central ray. For *level* $=\pm 2$ , ray trace information along the corner rays (in x-y space perpendicular to the reference ray) will be included, for a total of 5 rays. For *level* $=\pm 3$ , rays at the midpoints of each x-y boundary (which are most useful in r-z geometry) will also be included, for a total of 9 rays. For *level* $=\pm 4$ , all rays on the x-y boundary will be included. If *level* $>0$  material information will only be included along the central ray. If *level* $<0$ , material information will be included along each diagnostic ray. Setting *level* $=0$  turns off the inclusion of raytrace information.

The set of **drat** commands may be used as output options in any generator incuding spectral calculations, or they may be used for postprocessing a dump file with the **drat** runtime command. See the discussion of [Spectral Postprocessing](#), and the **dump** command.

```
table ["title"] [axes]
plot ["title"] [axes]
edit ["title"] [axes]
dumpedit ["title"] [axes]
```

Define a table/plot with columns/curves to be defined in the following **xvar** and **yvar** commands. If **edit** is specified, both a table and a plot are defined. If **dumpedit** is specified, the edits will appear in the dump file, but not in the ascii or PDB output files. The plot will be labeled with *title* in the ascii output file but not in the PDB output file (*title* must be in quotes). The numerical value of the final argument *axes* will be interpreted as defining the axis types to be used in interactive displays according to 0: linear-linear, 1: linear-log, 2: log-linear, 3: log-log, other: linear-linear.

```
display ["title"] [axes]
```

Define a display to be sent to the screen with curves to be defined in the following **xvar** and **yvar** commands, labeled with *title*, with axis types defined by *axes* (as above).

**send-to-dump**

Specify that the current edit (table/plot/edit/display) be put in the dumpfile. Both x-variables and y-variables will appear in the file.

```
time-integrated [snapshot] [/time1 [/time2]]
```

Specify that the current edit (table/plot/edit/display) be time-integrated, starting at *time1* (if defined) or problem initialization and ending at *time2* (if defined) or problem termination. If **snapshot** is specified, the time-integrated edit will be output at each edit interval, integrated to the current time, otherwise, the time-integrated edit will be output at problem termination.

# Export Controlled Information

## **time-averaged [snapshot] [*time1* [*time2*]]**

Specify that the current edit (table/plot/edit/display) be time-averaged, starting at *time1* (if defined) or problem initialization and ending at *time2* (if defined) or problem termination. If **snapshot** is specified, the time-averaged edit will be output at each edit interval, integrated to the current time, otherwise, the time-averaged edit will be output at problem termination.

## **time-maximum [*time1* [*time2*]]**

## **time-minimum [*time1* [*time2*]]**

Specify that the current edit (table/plot/edit/display) be the maximum or minimum value, over the interval starting at *time1* (if defined) or problem initialization and ending at *time2* (if defined) or problem termination.

## **integrated [*ireg*]**

## **averaged [*ireg*]**

## **mass-averaged [*ireg*]**

## **maximum [*ireg*]**

## **minimum [*ireg*]**

Specify that the current edit (table/plot/edit/display) be integrated, averaged, mass-averaged, maximum value or minimum value over space. If *ireg* is present, the spatial region will be that of region *ireg*.

## **rad-units *type1* *type2* ...**

Specify the units to use for continuum and spectral radiation quantities for all edits defined after this command. This also applies to line radiation quantities which do not include photons/mode in their units. The default units are c.g.s., i.e. ergs for energies and Hz for frequencies. Available options for *type* are “ergs”, “ev”, “kev” and “photons” for the energy unit and “per-hz”, “per-ev”, “per-kev”, “per-cm”, and “per-angstrom” for the frequency or wavenumber unit. These units will be used for appropriate y-variables only. The units of the x-variables will remain unchanged.

## **energy-integrated [*energy1* [*energy2*]]**

## **energy-averaged [*energy1* [*energy2*]]**

Specify that the current edit (table/plot/edit/display) be integrated or averaged over energy/frequency, starting at *energy1* (if defined, in eV) or the first energy and ending at *energy2* (if defined, in eV) or the last energy. The limits of integration will actually be those energy values which are closest in value to *energy1* and *energy2*.

## **sp-integrated [*energy1* [*energy2*]]**

## **sp-averaged [*energy1* [*energy2*]]**

Specify that the current spectral edit (table/plot/edit/display) be integrated over energy/frequency, starting at *energy1* (if defined, in eV) or the first spectral energy and ending at *energy2* (if defined, in eV) or the last spectral energy. The limits of integration will actually be those spectral energy values which are closest in value to *energy1* and *energy2*.

## **broadened [*resolution*]**

Specify that the current spectral edit (table/plot/edit/display) be broadened, i.e. convolved with a gaussian of fwhm (in eV) *resolution* (if defined) or param(82) if *resolution* is not defined.

# Export Controlled Information

**slice** *type index*

For a 3-d problem, restrict the current edit (table/plot/edit/display) be to a single plane in (k,l,m) space, specified by *type*, which may be either “k” or “m” and *index*.

**xvar** *xedit i1 i2 i3 i4 [multiplier]*

Define the x-variable for the plot defined by the preceding **plot** command. *Xedit* is the ascii name of the edit and must be one of the variables listed below. There may be up to three **xvar** commands for each plot. If any plots specify more than one **xvar** command, those plots can only be viewed with PDBVIEW; all other plots can be viewed with either ULTRA or PDBVIEW. NOTE: For a 2-d problem, “**xvar r**” is expanded to “**xvar x2d**” and “**xvar y2d**”. For a 3-d problem, “**xvar r**” is expanded to “**xvar x3d**”, “**xvar y3d**” and “**xvar z3d**”. If *multiplier* is present, it will be applied to the value(s) output for *xedit*.

**yvar** *yedit i1:j1:k1 i2:j2:k2 i3:j3:k3 i4:j4:k4 [multiplier]*

Define the y-variable for a curve in the plot defined by the preceding **plot** command. *Yedit* is the ascii name of the edit. There may be multiple **yvar** commands for each plot. Triplet notation *i:j:k* is optional for any index in a **yvar** command. If *multiplier* is present, it will be applied to the value output for *yedit*.

Independent variables for *xedit*:

cycle, iter, time  
ir, r, cdens, x2d, y2d, z2d, x3d, y3d, z3d, xy, k, kx, ky, kz, kr, l, lx, ly, lz, lr, m, mx, my, mz, mr  
ifr, energy, freq, wvl, ebins, fbins, wbins, ifrline, evline, isp, sp\_energy, sp\_freq, sp\_nu, sp\_wvl, iso, ziso  
level, elev

## Miscellaneous:

**tstart** *t*

Start problem at time *t*.

**tquit** *t*

Stop problem at time *t*.

**xfile** *filename*

Specify supplementary datafile (xfile) *filename*. The xfile will be given an index *ix* corresponding to its order of appearance in the generator file. ‘Null’ is a valid *filename*. If a datafile is specified, it must be available at runtime (for generation or restart) in the named directory if *filename* includes a path, or if no path is included, in either the working directory or the directory specified by the environment variable XFILEDIR. If the specified file is a text file, a binary version will be constructed in the same directory. A binary version of a particular type can be made with the **source** execute line command. A text version can be made from a binary version with the **reconst** execute line command.

**xmult** *ix type multiplier*

Specify a multiplier to be applied to data of type *type* in xfile *ix*. *Type* may be (almost) any of the data types allowable in the xfile, i.e. **pbins**, **pbcgs**, **r**, **x2d**, **y2d**, **z2d**, **u**, **u2d**, **v2d**, **w2d**, **durdr**, **d**, **te**, **ti**, **tr**, **bx**, **by**, **bz**, **bfield**, **ne**, **ni**, **yiso**, **nehrot**, **tehot**, or **edist**. The multipliers for **x2d** also are applied to **x3d** sections.

# Export Controlled Information

**newxfile** *type1 type2 ...*

Construct a supplementary datafile (xfile) from the current run, including data types *type1*, *type2*, .... The xfile name will have the same base as the current output files. Valid types include **mesh**, **velocity**, **jnu**, **rho**, **tev**, **tiv**, **trv**, **temperatures**, **ne**, **ni**, **zbar**, **nehot**, **edist**, **bfield**, and **all**. A type of **none** or **off** can be used to prevent data from being sent to the xfile. Times to be included will be controlled by switch(122) and param(101).

A type of **zones=i:j:k** specifies a restricted set of zones in triplet notation for which data will be sent to the xfile.

A type of **ascii** will produce an ascii file with a suffix of “xfa”, while other types will have a suffix of “xfb”. A type of **binary** will produce a binary file. A type of **history** will produce a Yorick history file. A type of **directored** will produce a PDB or HDF5 file with a directory structure for each timestep. The default type is **ascii**. Yorick history files are not available with HDF5 files.

**boundary package type ir** [**history id**] **multiplier** **value** or  
**boundary package type k1 k2 l1 l2** [**history id**] **value** or  
**boundary package type k1 k2 l1 l2 m1 m2** [**history id**] **value**

Specify boundary conditions of a given *type* for a particular physics *package*, which can be “radiation”, “conduction”, “hydro”, “velocity”, “pressure”, “divertor”, “current” or “all”. The *type* may be “value”, “gradient”, “flux”, “reflecting”, “mirror” (equivalent to “reflecting”), “milne”, “isotropic”, “streaming” or “recycling”. This boundary condition applies at node *ir* (if 1-d), along the line connecting (*k1,l1*) and (*k2,l2*) (if 2-d), or along the plane specified by (*k1,l1,m1*) and (*k2,l2,m3*) (if 3-d). For 2-d (3-d) geometries, the indices must specify a line of constant *k* or *l* (plane of constant *k*, *l*, or *m*). If **history** is present, time history *id* will be used for the boundary condition, with *value* applied as a multiplier.

The default boundary condition for problem boundaries (nodes, lines or planes) with no **boundary** specifications is:

radiation:	free
conduction:	reflecting
hydro:	free
divertor:	reflecting

Radiation boundary condition of types “streaming” and “isotropic” are equivalent to **xfilebc** commands. Type “streaming” requires two additional arguments which specify the direction of the boundary radiation:

boundary radiation streaming *mu phi* ...

Boundary conditions of type “velocity” and “pressure” apply to hydrodynamics only.

Almost all combinations of *package* and *type* are allowed. Boundary conditions of *type* “recycling” are only allowed for *package* “divertor”. Boundary conditions of *type* “milne” are only allowed for *package* “radiation”, or “divertor” with *value*=0. Those combinations that are not allowed will result in an error message.

**source** *type* ...

Specify values for various quantities, depending on *type*. Current valid types are “jbndry”, “ni”, “yiso”, “y”, “te”, “ti”, “tr”, “esree”, “esrci”, “esrcr”, “rho”, “ne”, “r”, “x2d”, “x3d”, “y2d”, “y3d”, “z2d”, “z3d”, “vel”, “u2d”, “u3d”, “v2d”, “v3d”, “w2d”, “w3d”, “bfield”, “bx”, “by”, “bz”, “jnu”, “jnu2”, “laser”,

# Export Controlled Information

“lpump”, and “param”. Others may be added in the future.

**source jbndry ix black-body tev [multiplier]**

This form of the **source** command is equivalent to specifying an xfile of index *ix* which includes photon energy densities appropriate for a blackbody of temperature *tev* (eV). The energy densities can be used for boundary conditions as specified by an **xfilebc** command and will be multiplied by *multiplier*, if present.

**source ni ix iz z [a] [multiplier] [ireg] [ymin]**

This form of the **source** command will set the number density of element *iz* to that contained in xfile *iz* if that xfile contains an **ni** section with appropriate identifiers. The identifier in the **znuc** section must match *z*. If an **anuc** section is present and *a* is non-zero, then the section with the identifier which most closely matches *a* will be used. The number density will be multiplied by *multiplier*, if present. If *ireg* is present and is non-zero, then the section labeled with *ireg* will be used. If *ymin* is present, number densities less than *ymin* will be set to zero. This can prevent spending time doing calculations on materials with minuscule number densities - also see param(124). If *z*<0, the **znuc** and **anuc** sections will be ignored and the mass density in xfile *iz* will be used, and the user should specify a multiplier to correct the values.

**source yiso ix iz iso z a [multiplier] [ireg] [ymin]**

This form of the **source** command will set the number density of isosequence *iso* of element *iz* to that contained in xfile *iz* if that xfile contains an **ni** section with appropriate identifiers. The identifier in the **znuc** section must match *z* and the identifier in the **isonuc** section must match *iso*. The other arguments have the same meaning as for **source ni**.

**NOTE:** Use of the **source ni** form implies that the user is specifying all desired element densities and electron densities, so the electron density and related quantities will not be updated after calculating populations (equivalent to switch(58) = 1).

**source type [i1 i2 ...] value/rate/integral/initial xfile/history/profile/svlist/constant id/value multiplier [ireg/k1 k2 l1 l2 m1 m2]**

This form of the **source** command identifies time history or profile *id* as specifies a quantity *type* to be applied in region *ireg* or into the nodes bounded by *k1 k2 l1 l2 m1 m2*, if either of these is present, or to all regions, if neither specification is present. The value will be obtained from time history *id* (if **history** is specified) or from *value* (if **constant** is specified). The values will be interpolated to the current time if **value** is specified, or will be integrated (and divided by the timestep) to ensure conservation if **rate** is specified. If **integral** is specified, the values are assumed to already be a time integral of the desired quantity.

**NOTE:** Each source will be applied on each timestep unless the 10’s digit of switch(90) is non-zero and **history** is not specified. In this case, the source will be applied only at cycle 0.

If *type* is one of “te”, “ti”, “tr”, “rho”, “ne”, “r”, “x2d”, “x3d”, “y2d”, “y3d”, “z2d”, “z3d”, “vel”, “u2d”, “u3d”, “v2d”, “v3d”, “w2d”, “w3d”, “bfield”, “bx”, “by”, or “bz” this quantity will be set in the same manner as if it were present in an xfile. If *type* is one of “esrci” or “esrci”, it will be treated as an energy source into the free electrons (for “esrci”) or ions (for “esrci”).

If *type* is “jnu” or “jnu2”, it must be followed by two values *e1 e2* denoting the energy range over which this **source** command applies. In addition, if the word **black-body** follows the energy range, each value will be interpreted as a temperature (eV) for a blackbody distribution. Otherwise, the value(s) will be interpreted as angle-averaged radiation intensities (erg/cm<sup>2</sup>/sec/Hz). If the word **power-law** follows the energy range, it must be followed by 3 additional arguments,  $\alpha$ ,  $\beta$ , and  $\epsilon_0$ , and the radiation intensity at

## Export Controlled Information

energy  $\epsilon$  (in eV) will be multiplied by  $(\epsilon - \epsilon_0)^\alpha \exp(\beta(\epsilon - \epsilon_0))$ .

If *type* is “jbndry”, it must be followed by an index denoting an index *i* plus two values *e1 e2* denoting the energy range over which this **source** command applies. The index *i* can be used to match the intensities with boundary conditions. The options for **black-body** and **power-law** distributions are available as for *type* “jnu”.

If *type* is “edist”, it must be followed by two values *e1 e2* denoting the energy range over which this **source** command applies. In addition, if the word **maxwellian** follows the energy range, each value will be interpreted as a temperature (eV) for a Maxwellian distribution. Otherwise, the value(s) will be interpreted as distribution function values (#/cm<sup>3</sup>/eV). If the word **power-law** follows the energy range, it must be followed by two additional arguments,  $\alpha$  and  $\beta$ , and the number density at energy  $\epsilon$  (in eV) will be multiplied by  $\epsilon^\alpha \exp(\beta\epsilon)$ .

If *type* is “opacity”, it must be followed by two values *e1 e2* denoting the energy range over which this **source** command applies. The next three values gives the exponents for the density, temperature and frequency, according to the prescription for an analytic power law opacity.

If *type* is “laser”, it must be followed by a single value *wvl* giving the laser wavelength (in cm). The value may also be in the form *xn*, in which case *n* will be interpreted as a multiplier on the reference wavelength of 1.06 μm. The values specified will be interpreted as laser intensities (erg/cm<sup>2</sup>/sec) which can produce heating via inverse bremmstrahlung absorption.

If *type* is “lpump”, it must be followed by a single integer *i1* corresponding to the source *index* on an **lpump** command. The values specified will be interpreted as photon intensities (in photons/mode) for the radiative transition on the **lpump** command.

If *type* is “param”, it must be followed by a single integer *i1*, and will have the effect of setting the value of *param(i1)* on every timestep.

**history** *id* [*value\_multiplier*] [*time\_multiplier*] [*type p1 p2 p3 ...*]

Identify the following **tv** commands as a time history with identifier *id*. All following **tv** commands will have identifier *id* until another **history** command is encountered. The values will be multiplied by *value\_multiplier* and the times will be multiplied by *time\_multiplier*, if one or both of these values are present. If *type* is present, the values *p1, p2, p3 ...* will be used as parameters in an analytic form for the time history and no **tv** commands are required.: If **tv** commands are present, *type* will be ignored.

The available analytic forms are:

*type* = ”gaussian” :    value =  $\exp(-x^2)$  , where  $x = (\text{time}-p1) / p2$ .

**tv** *time value*  
**tp** *time value*

Add a (*time,value*) pair to the time history identified by the previous **history** command. **tp** is an alternative syntax equivalent to **tv**.

**svlist** *id* [*energy\_multiplier*] [*value\_multiplier*]

Identify the following **sv** commands as an spectrum defined by (energy,value) pairs with identifier *id*. All following **sv** commands will have identifier *id* until another **svlist** command is encountered. The energy values will be multiplied by *energy\_multiplier* and the values will be multiplied by *value\_multiplier*, if one or both of these values are present.

# Export Controlled Information

**sv** *energy value*

Add a (*energy,value*) pair to the spectrum identified by the previous **svlist** command.

**restart** [**pdb|hdf5**]

Make restart dumps with a frequency controlled by param(87) and switch(65). Restarts may also be made on command in interactive mode.

If **pdb** or **hdf5** is specified, the file will a PDB or HDF5 file, respectively.

**sdump** [**spectre**] [**pdb|hdf5**] [*filename*]

Make spectral dumps with a frequency controlled by param(92) and switch(73). Spectral dumps may also be made on command in interactive mode. If **spectre** is specified, ascii dumps for the code SPECTRE will be made, otherwise, binary dumps for CRETIN will be made. If *filename* is specified, this will be used as the base for the sequence of spectral dumps, rather than using the generator name.

If CRETIN is running in spectral postprocessing mode from a restart dump and is reading , this command specifies the full filename of the first spectral dump. The default file is the spectral dump made at the time of the restart dump.

If **pdb** or **hdf5** is specified, the file will a PDB or HDF5 file, respectively.

**dump** [**directored|history**] [**pdb|hdf5**] *variable1 variable2 ...*

Specify variables to output to a PDB or HDF5 file after each timestep for which edits are done. In most cases, the variable is also an edit variable and is described in the edit section. Any combination of the following variables may be specified:

mesh	- nodal positions (r if 1-d, (x2d,y2d) if 2-d, (x3d,y3d,z3d) if 3-d)
r	- nodal positions
velocity	- nodal velocities
jnu	- continuum radiation intensity
absn, emis	- continuum absorption, emission
jbar	- line strengths
radiation	- jnu + jbar
tev, tiv, trv	- electron, ion, radiation temperatures
temperatures	- tev + tiv + trv
rho	- mass densities
ne, ni	- electron, ion number densities
yiso, y	- iso-sequence, atomic level populations
populations	- ni + yiso + y
jsp	- spectral radiation intensity
kappa_sp, emis_sp	- spectral absorption, emission
spectrum	- jsp + kappa_sp + emis_sp
jbar	- line strength
lines	- jbar + ylinel + ylineu + sigma
radiation	- jnu + jbar
drat	- mesh + kappa_sp + emis_sp + velocity
all	- all of the above
none	- none of the above
edits	- include all edits

# Export Controlled Information

Edits for which **send-to-dump** have been specified will also appear in the dumpfile.

If **directored** is specified, the file will have a directory containing data for each time in the dumpfile. This is the default format if HDF5 files are being produced.

If **history** is specified, the file format will be that of a Yorick time history file. This is the default format if PDB files are being produced.

If **pdb** or **hdf5** is specified, the file will a PDB or HDF5 file, respectively.

## **dumpfile** *filename*

Specify the name of the first dumpfile to write or read for the current run to be *filename*. This will be used as the base for the sequence of dumps, rather than using the generator name.

If CRETIN is running in spectral postprocessing mode from a restart or spectral dump reading spectral opacities from a dump file , this command specifies the full filename of the first dump file. The default dump file is the one made at the time of the restart/spectral dump.

**sense** *i1:j1:k1 i2:j2:k2 ...*  
**sense all**

Make sensitivity dumps (PDB) for zones specified by *i1:j1:k1 i2:j2:k2 ...*, or for all zones if **all** is specified. Triplet notation (*i:j:k*) is optional. A single family of dumps will hold information for all zones specified. Sensitivity dumps may also be made on command in interactive mode.

**nltdump** *i1:j1:k1 i2:j2:k2 ...*  
**nltdump all**  
**nltdump rates**  
**nltdump spectrum | spectrum-wvl [time-integrated | transmission]**  
**nltdump spectrum-iso**  
**nltdump user** *username*  
**nltdump code** *codename*

Make NLTE workshop dumps (ASCII) for zones specified by *i1:j1:k1 i2:j2:k2 ...*, or for all zones if **all** is specified. Triplet notation (*i:j:k*) is optional. If **rates** is specified, rate matrix dumps will also be produced. If **spectrum** (**spectrum-wvl**) is specified, emission spectra will be included as functions of photon energy (wavelength). If **time-integrated** is specified, the emission spectra will be time-integrated. If **transmission** is specified, transmissions ( $\exp(-\tau)$ , where  $\tau$ =optical depth) will be included in place of emission spectra. If **spectrum-iso** is specified, separate emission spectra will be included for populated each isosequence. User and code identification can be specified through **user** and **code**. Files will be named according to workshop specifications.

**opacdump** [*codename*] [*tag*]

Make opacity workshop dumps (ASCII) for all zones. Code identification and case tag can be specified through *codename* and *tag*, if these items are present.

**ratedump** *i1:j1:k1 i2:j2:k2 ...*  
**ratedump all**

Dump rates to the ascii output (.tbl) file for zones specified by *i1:j1:k1 i2:j2:k2 ...*, or for all zones if **all** is specified. Triplet notation (*i:j:k*) is optional.

# Export Controlled Information

**gainlist** *i1:j1:k1 i2:j2:k2 ...*  
**gainlist all**

Send list of gains ( $\text{cm}^{-1}$ ) in descending order for zones specified by *i1:j1:k1 i2:j2:k2 ...* to the *.tbl* file, or for all zones if **all** is specified. Triplet notation (*i:j:k*) is optional. Only transitions with gains greater than param(97) will be included. All zones for which gainlists are desired must be specified at generation time, but gainlists for each specified zone may be turned on and off in interactive mode or through **when** commands.

**emislist** *i1:j1:k1 i2:j2:k2 ...*  
**emislist all**

Send list of emissivities ( $\text{erg}/\text{cm}^3/\text{sec}/\text{Hz}/\text{ster}$ ) in descending order for zones specified by *i1:j1:k1 i2:j2:k2 ...* to the *.tbl* file, or for all zones if **all** is specified. Triplet notation (*i:j:k*) is optional. Only transitions with emissivities greater than param(97) will be included. All zones for which emislists are desired must be specified at generation time, but emislists for each specified zone may be turned on and off in interactive mode or through **when** commands.

**lpump** *iz iso1 i1 iso2 i2 value [de]*  
**lpump** *iz iso1 i1 iso2 i2 source index [de]*

Specify photon intensity of *value* (in photons/mode) for the radiative transition between states (*iso1,i1*) and (*iso2,i2*) in model *iz*. If **source** is specified, the values will be specified by a **source** command which has type “lpump” and identifier *i1=index*. If an energy width *de* (in eV) is specified, the photon intensity can be used to drive photoionizations and other photoexcitations, depending on the value of switch(32).

**param** *i value*

Set param(*i*) to *value*. All parameters are floating-point numbers.

**switch** *i value*

Set switch(*i*) to *value*. All switches are integers.

**when cycle** *i*  
**when time** *t [limit]*  
**when always**

This command specifies a time or cycle for executing a series of interactive commands. The commands to be executed follow the **when** command and are terminated by a **done** command. They will be executed at cycle *i*, or at the first cycle for which the time is greater than or equal to *t*. They will be executed each cycle if **always** is specified. Most commands will be executed at the beginning of the cycle, but certain commands (making dumps or edits, **dte** or **dtinfo**) will be executed at the end of the cycle. If **limit** is specified, the timestep will be altered (subject to constraints on the timestep) in an attempt to end up at time *t*. The interactive commands available for execution are listed in the section on [Runtime Control](#).

**end**

This command ends a series of interactive commands identified with a **when** command.

Macros:

# Export Controlled Information

CRETIN has a rudimentary macro capability which can be very useful in constructing generators by parameterizing zoning, binning, code options or edit specifications. Macro values may also be set from the command line, allowing simple scripts to generate and run a series of problems from a single generator file.

**alias** *word1 word2*  
**alias** *word1 expression*

Substitute *word2* for *word1* for each occurrence in the generator. This is especially useful for setting up zoning and edit requests which can be quickly changed later. Simple arithmetic expressions may be used to calculate a value to be substituted for *word2*. An expression may include numbers, symbols defined with another **alias** command, operators, and parentheses.. The expression is evaluated using algebraic notation and normal operator precedence. Trigonometric arguments are in radians. Operators may be, but no longer need to be, space delimited. The evaluation of **alias** commands does not depend on order of appearance, but circular definitions are not allowed and will produce error messages. An expression with characters that may be interpreted as operators (e.g. / in a path or – used as a dash) will be evaluated numerically and will not be used as a character string. This may produce error messages. To prevent such an expression from being evaluated numerically, enclose it in double quotes.

Defining the symbol *word* also defines the symbols *word+* and *word-*, which will evaluate numerically to the values of (*word* + 1) and (*word* – 1), respectively.

The allowable operators and some predefined values are listed below.

**echo** *symbol*

This will print the numerical value assigned to *symbol* by an **alias** command.

**#define** *word1 [word2]*

This command is a special form of the **alias** command. If *word2* is not present, it is given the default value of 1. Expressions are not allowed in this command.

Predefined values:

PI	= 3.141592653589793	$\pi$
AVGD	= 6.02214129 x 10 <sup>23</sup>	Avogadro's number
CS	= 2.99792458 x 10 <sup>10</sup>	speed of light (cm/s)
KB	= 1.3806488 x 10 <sup>-16</sup>	Boltzmann constant (erg/K)
KBEV	= 1.602176565 x 10 <sup>-12</sup>	Boltzmann constant (erg/eV)
HP	= 6.62606957 x 10 <sup>-27</sup>	Planck constant (erg sec)
HPEV	= 4.135667516 x 10 <sup>-15</sup>	Planck constant (eV sec)
EMASS	= 9.10938291 x 10 <sup>-28</sup>	electron mass (g)
PMASS	= 1.672621777 x 10 <sup>-24</sup>	proton mass (g)
AMASS	= 1.660538921 x 10 <sup>-24</sup>	AMU mass (g)
QELEC	= 4.8032045 x 10 <sup>-10</sup> (= 10 <sup>-8</sup> KBEV*CS)	elementary charge (statcoulomb)
ABOHR	= 0.52917723 x 10 <sup>-8</sup>	Bohr radius (cm)
RYD	= 13.605692	Rydberg (eV)

The following sizes are defined and available for use as long as they are independent of the specifications of the spatial and energy meshes:

KMAX

number of nodes in K-direction

CRETIN

May 20, 2019

49

# Export Controlled Information

## Export Controlled Information

LMAX	number of nodes in L-direction
MMAX	number of nodes in M-direction
NT	total number of nodes
NGP	number of continuum energy groups

Operators:

Unary: + - sqrt exp ln log sin cos tan asin acos atan int nint  
Binary: + - \* / ^ \*\* min max (Either ^ or \*\* may be used for exponentiation)

# Export Controlled Information

## Code Options

Note: In case of a discrepancy between a description in this section and the corresponding description in the summary of switches and parameters, the summary is more likely to be up to date.

### Atomic Kinetics:

The atomic kinetics LTE or NLTE populations under either steady-state or time-dependent conditions. For LTE, populations are calculated using the Saha equation. For NLTE, populations are calculated with an implicit solution of a first-order (in time) rate equation.

Populations may be initialized for either LTE or steady-state NLTE conditions, or may be initialized explicitly in the generator.

Electron distributions may be a combination of thermal (Maxwellian or Fermi-Dirac) and non-thermal (mono-energetic or arbitrary).

High-density fudges include continuum lowering via a Stewart-Pyatt or Ecker-Kroll formula, and multiple knobs for controlling natural line widths.

Relevant switches and parameters: (see also [Escape Factors](#))

**switch 13**      *default value = 0*

If switch(13)  $\neq 0$ , then the atomic kinetics calculations (on a zonal basis) will be sorted according to computational time as measured on the previous cycle (or by an estimate on the initial cycle). If switch(13)  $> 0$ , the kinetics calculations will be done in decreasing order (most expensive first), while if switch(13)  $< 0$ , they will be done in increasing order. Either option has no effect on the results, but may run faster when using multiple threads.

**switch 14**      *default value = 0*

If switch(14) = 0, populations will be saved between cycles and stored in restart (and other) dumps. If switch(14)  $\neq 0$ , fractional populations (relative to that element's populations) will be saved and stored. This option is primarily for compatibility with (and testing of) data usage by other codes.

**switch 15**      *default value = 0*

If switch(15)  $> 0$ , the atomic kinetics is allowed to subcycle in time, when doing time-dependent kinetics, up to a maximum number of subcycles given by switch(15). In this case, each zone will subcycle independently and the kinetics will not affect the overall code timestep.

If switch(15)  $< 0$ , use a fixed number of kinetics subcycles given by abs(switch(15)).

**switch 20**      *default value = 1*

Controls choice of LTE / NLTE and radiation used by kinetics:

If switch(20) = 0, calculate LTE populations. In this case, the radiative rates will be evaluated only to calculate the opacities and emissivities. Collisional rates will not be evaluated. This choice is inconsistent with calculating heating rates or heat capacities.

# Export Controlled Information

If  $\text{switch}(20) < 0$ , calculate NLTE populations assuming fixed electron density.

If  $\text{switch}(20) > 0$ , calculate NLTE populations assuming fixed ion densities (i.e. fixed mass density).

If  $\text{switch}(20) = \pm 1$ , calculate NLTE populations using rate matrices. The radiation intensities used to calculate the continuum rates will be (in order of priority):

- (1) intensities calculated by continuum transfer
- (2) intensities from an xfile
- (3) zero

If  $\text{switch}(20) = \pm 2$ , the radiation intensities used to calculate the continuum rates will be Planckian, corresponding to the local electron temperature,  $T_e$ , with a multiplier of param(47).

If  $\text{switch}(20) = \pm 3$ , the radiation intensities used to calculate the continuum rates will be Planckian, corresponding to the local radiation temperature,  $T_r$ , with a multiplier of param(47). The radiation temperature can currently only be set from the generator or an xfile.

**switch 21**      *default value = 1*

If  $\text{switch}(21) = 0$ , the radiative emission from any transition will be calculated using the energy difference between the participating levels. If  $\text{switch}(21) \neq 0$ , the radiative emission will be calculated using the transition energy, which may be different, and the emission spectrum will be modified so that a Planckian source function would result if the level population ratio matched the LTE value.

**switch 22**      *default value = 0*

If  $\text{switch}(22) = 0$ , transition rates calculated and stored during the previous kinetics calculation will be used to calculate level lifetimes and photoexcitation broadening. If  $\text{switch}(22) = 1$ , the first iteration during a cycle will start with zero lifetime broadening. This option facilitates comparisons between calculations starting up at different times. If  $\text{switch}(22) > 1$ , all iterations during a cycle will start with zero lifetime broadening. If a kinetics calculation starts with zero lifetime broadening, transition rates (including a first pass at photoexcitations) calculated without lifetime broadening will provide lifetime broadening for photoexcitation broadening.

**switch 24**      *default value = 1*

Populations will be calculated assuming fixed ion densities if  $\text{switch}(24) \geq 0$ , or assuming fixed isosequence ion densities if  $\text{switch}(24) < 0$ .

**switch 25**      *default value = 1*

Populations will be calculated assuming steady-state if  $\text{switch}(25) = 0$ , or assuming time-dependent if  $\text{switch}(25) > 0$ . No kinetics calculations will be done if  $\text{switch}(25) < -1$ .

If  $\text{switch}(25) = 2$ , calculate approximate LTE (QLTE) and steady-state NLTE (QSS) isosequence distributions with quick approximate calculations and use these to determine whether to use LTE or NLTE to calculate populations (see param(32)).

If  $\text{switch}(25) = -1$ , calculate QLTE and QSS isosequence distribution (for edit purposes only).

**switch 28**      *default value = 0*

Controls initialization of populations and radiation:

## Export Controlled Information

If  $\text{switch}(28) < 0$ , the initial populations will be LTE for the given electron density (e.g. as fixed by an xfile)

If  $\text{switch}(28) = 0$ , the initial populations will be LTE for the given ion density (i.e. mass density).

If  $\text{switch}(28) = 1$ , the initial populations will be steady-state, consistent with radiation for the initial time (either zero intensities, or set from an xfile).

If  $\text{switch}(28) = 2$ , the initial populations will be steady-state, but the radiation intensities will be initialized to zero.

If  $\text{switch}(28) = 3$ , no initialization calculation will be done for populations. The initial populations must be set from the generator with **level** commands. The radiation will be initialized to boundary condition values.

If  $\text{switch}(28) > 3$ , no initialization calculation will be done for populations or for radiation intensities.

**switch 32**      *default value = 1*

Controls coupling of line intensities to kinetics:

Line intensities will not be seen by any rate calculations (except the line transition) if  $\text{switch}(32) = 0$ .

Line intensities will be averaged onto the continuum bins and will participate in photoionizations if  $\text{switch}(32) > 0$ .

Line intensities will be averaged onto the continuum bins and will participate in photoexcitations if  $\text{switch}(32) > 1$ . The appropriateness of this is left to the user to decide.

**switch 55**      *default value = 1*

Controls continuum lowering options (see also  $\text{switch}(155)$  and  $\text{switch}(176)$ ):

If  $\text{switch}(55) = -1$ , do an approximate accounting for missing Rydberg levels by lowering the ionization level of the next isosequence to the position of the first missing level, estimated by scaling with the principal quantum number.

If  $\text{switch}(55) = 0$ , no continuum lowering.

If  $\text{switch}(55) = 1$ , use the Stewart-Pyatt or Ecker-Kroll model with a formula for degeneracy lowering (see  $\text{param}(79)$ ).

If  $\text{switch}(55) = \pm 2$ , use the Stewart-Pyatt or Ecker-Kroll model with microfield degeneracy lowering.

+2: microfield appropriate for the average charge state

-2: separate microfield for each charge state

If  $\text{switch}(55) = \pm 3$ , use microfield degeneracy lowering without continuum lowering.

+3: microfield appropriate for the average charge state

-3 separate microfield for each charge state

For other values of  $\text{switch}(55)$  use the Stewart-Pyatt or Ecker-Kroll model without degeneracy lowering.

## Export Controlled Information

If  $\text{switch}(55) > 10$  or  $< -10$ , then use the maximum of the lowering given by the approximate accounting for missing Rydberg levels and the lowering given by the model specified by the 1's digit.

**switch 56**      *default value = 2*

If continuum lowering is turned on, the LTE calculations will be iterated a maximum of  $\text{switch}(56)$  times to converge the populations.

**switch 57**      *default value = 0*

Controls the calculation of natural line widths for photoexcitations:

If  $\text{switch}(57) \geq 0$ , the natural width will include the lifetimes of both upper and lower states.  
If  $\text{switch}(57) < 0$ , the natural width will include the lifetime of the upper state only.

If  $\text{switch}(57) = 0$  or  $\pm 1$ , the natural width will include contributions from all transitions, but will not include the lifetime of a ground state.

If  $\text{switch}(57) = \pm 2$ , the natural width will include contributions from all transitions except recombinations, but will not include the lifetime of a ground state.

If  $\text{switch}(57) = \pm 3$ , the natural width will include contributions from excitations only, but will not include the lifetime of a ground state.

If  $\text{switch}(57) = \pm 4$ , the natural width will include contributions from all transitions.

If  $\text{switch}(57) = \pm 5$ , the natural width will include contributions from photoexcitations only (including those arising from augxs and augis rates).

**switch 58**      *default value = 0*

Controls updating of material properties (ne, zbar, etc.) after kinetics calculations:

If  $\text{switch}(58) < 0$ , assume constant electron densities, scale ion densities accordingly and update material properties.

If  $\text{switch}(58) = 0$ , assume constant ion densities and update material properties.

If  $\text{switch}(58) > 0$ , do not update material properties such as the electron density – This option is set automatically when ion and electron densities are set from the xfile.

**switch 60**      *default value = 0*

If  $\text{switch}(60) \neq 0$ , the atomic kinetics will be combined with a particle transport model to compute populations (if available). These options are available in 1-d only.

If  $\text{switch}(60) = 1$ , transport neutral ions only with a diffusion model.

If  $\text{switch}(60) = 2$ , transport neutral ion and electrons with the partially-ionized plasma (PIP) transport package.

# Export Controlled Information

**switch 77**      *default value = 1*

Do not allow negative continuum absorption coefficients if switch(77) ≠ 0.

**switch 91**      *default value = 1*

If switch(91) ≠ 0, make a second pass through the kinetics with updated populations to calculate opacities and emissivities.

**switch 97**      *default value = 0*

If switch(97) > 0, kinetics calculations will be done for node number switch(97) only. All other nodes will be ignored.

**switch 101**      *default value = 0*

Apply an approximate Debye shielding treatment for collisional rates if switch(101) ≠ 0.

**switch 102**      *default value = 0*

Controls use of level populations read from xfile as specified by **source y** commands:

If switch(102) ≠ 0, those level populations set by the xfile are interpreted as constraints on the kinetics solution. If switch(102) > 0, conservation of the total ion population is enforced as well.

If switch(102) = 0, level populations are calculated normally.

**switch 111**      *default value = 0*

If switch(111) ≠ 0, skip further kinetics calculations on those zones which have already converged. If required, one final iteration will be done for edit purposes.

**switch 112**      *default value = 0*

1-10's digit: choose linear algebra solver for rate matrices:

<0: use explicit solution

0: use LU decomposition if the number of levels being solved for at that time is less than or equal to switch(140) and use an iterative solver if the number of levels is larger.

1: use a block tridiagonal solver.

2: use an iterative solver.

3: use projective integration (if available).

4: use banded LU decomposition (same conditions as regular LU decomposition).

other: same as 0.

100's: time call to solver if ≠ 0

1000's: time rate calculations if ≠ 0

**switch 124**      *default value = 0*

Control iteration of electron densities.

If switch(124) = 0, use a linearization process.

# Export Controlled Information

If switch(124) > 0, use relaxation as described under param(96).

If switch(124) < 0, use relaxation and accelerate the convergence every fourth iteration by applying Euler's transformation for an oscillating series.

**switch 132**      *default value = 0*

Modify rate calculations by treating excitations to collectivized states as ionizations if switch(132) ≠ 0.  
See D.V. Fisher and Y. Maron, JQSRT **81**, 147 (2003).

**switch 136**      *default value = 1*

Control treatment of integrated radiation quantities for kinetics.

If switch(136) = 0, treat as histograms.

If switch(136) > 0, correct radiation and opacity for Planckian shape within group.

If switch(136) < 0, correct spontaneous emission terms.

**switch 140**      *default value = 2000*

Maximum number of levels for which a direct solution of rate matrices will be used – see switch(112).

**switch 141**      *default value = 0*

Control printing of linear solver diagnostics for the iterative kinetics solver:

If switch(141) = 0, do not print diagnostics.

If switch(141) < 0, print diagnostics to the screen.

If switch(141) > 0, print diagnostics to the ascii output file.

**switch 142**      *default value = 0*

Choose the type of iterative kinetics solver:

If switch(142) = 0, use biconjugate gradient with ilur preconditioning.

If switch(142) ≠ 0, use gmres with ilur preconditioning.

**switch 143**      *default value = 100*

The maximum number of iterations used by biconjugate gradient for the iterative kinetics solver.

**switch 144**      *default value = 100*

The maximum number of orthogonalizations used by gmres for the iterative kinetics solver.

**switch 145**      *default value = 4*

## Export Controlled Information

The maximum number of restarts used by gmres for the iterative kinetics solver. Set this to -1 for no restarts as 0 will result in the (default) value of 10 restarts.

**switch 146**      *default value = 0*

If switch(146)  $\neq 0$ , include multiphoton ionization rates. Only the lowest contributing order will be included and only for photon energies below threshold. The implementation uses the weak-field scalings given by Delone and Krainov, “Multiphoton Processes in Atoms”, p. 93.

**switch 149**      *default value = 0*

If switch(149)  $\neq 0$ , account for virtual levels separately from levels explicitly included in the atomic model. These are the virtual doubly-excited levels implied by augis sections in the model, which are assumed to have negligibly small populations. Their populations are always included in the total and iso-sequence populations. If switch(149) = 0, level populations are calculated assuming that the virtual populations are negligible. If switch(149)  $\neq 0$ , this assumption is not made. In this case, the level populations will not sum to the iso-sequence and total populations, with the difference accounted for by the virtual populations.

**switch 150**      *default value = 0*

If switch(150) = 0, photo-recombination rate integrals will be calculated by evaluating the integrands at the bin center. If switch(150) > 0, the integrals will use an average of  $\exp(-x)$  over each bin. If switch(150) < 0, only the threshold bin will average  $\exp(-x)$  over the bin.

If switch(150) =  $\pm 1$ , this will apply to calculations of both rates and opacities. For other values, this applies to calculations of rates only.

**switch 151**      *default value = 1*

If switch(151)  $\neq 0$ , include the effects of electron degeneracy when calculating rates, opacities and emissivities.

If switch(151) < 0, use analytic degeneracy corrections for collisional rates obtained by assuming the collision strength does not change with electron energy.

If switch(151) = 1, add an additional correction for collisional ionizations obtained from evaluating collision strengths at electron energies close to thermal and Fermi energies.

If switch(151) > 1, integrate the collision strength numerically to obtain corrections for collisional excitation.

If switch(151) > 2, integration the collision strength numerically to obtain corrections for collisional ionizations.

**switch 155**      *default value = 0*

If switch(155)  $\neq 0$ , do not change ground state degeneracies when applying continuum lowering. This has the effect of not allowing an entire isosequence to be lowered out of existence.

**switch 163**      *default value = 0*

If switch(163) = 0, interpret tehot values as temperatures describing a Maxwellian distribution. If switch(163)  $\neq 0$ , interpret tehot values as energies for a mono-energetic distribution.

# Export Controlled Information

**switch 170**      *default value = 0*

Options for modifying collisional ionizations for continuum lowering:

If switch(170) = 0, evaluate the collision strengths / rate coefficients using the lowered threshold values. In addition, scale the logarithmic and rational parts of the collision strengths to account for missing excitations as described in Fontes, Sampson & Zhang, Phys Rev A48 (1993) 1975. The scalings only apply to the parameterizations labeled “colon” and “sampson ionize”.

If switch(170) > 0, evaluate the collision strengths / rate coefficients using the lowered threshold values, but do not apply any scalings.

If switch(170) < 0, evaluate the collision strengths / rate coefficients using the original threshold values.

**switch 171**      *default value = 2*

If switch(171) = 0, initialize the isosequence window limits for a steady-state or time-dependent NLTE calculation by using the isosequence distribution from the previous cycle, incremented by +/- 1 sequence. If switch(171) > 0, initialize the isosequence window limits by using the isosequence distribution from the previous cycle, incremented by +/- switch(171) sequences. If switch(171) < 0, initialize the isosequence window limits with a calculation involving only approximate rates between sequences, incremented by +/- switch(171) sequences.

**switch 173**      *default value = 0*

Control the calculation of coulomb energy and pressure returned by the kinetics.

If switch(173) = 0, the coulomb pressure is assumed to be 1/3 of the change in internal energy density due to continuum lowering. No explicit coulomb energy is calculated, as that has been included in the internal energy density.

If switch(173) = 1, use an ion sphere model for each charge state to calculate coulomb energy and pressure.

If switch(173) = 2, use a single ion sphere calculated with the average value of  $Z^2$ .

**switch 175**      *default value = 0*

Control the calculation of collisional transitions by hot electrons and a non-thermal electron distribution.

If switch(175) = 0 or  $\pm 3$ , include transition rates for all collisional transitions.

If switch(175) =  $\pm 1$ , include excitation / de-excitation rates only.

If switch(175) =  $\pm 2$ , include ionization / recombination rates only. Recombination rates are currently only calculated for Maxwellian distributions.

If switch(175) < 0, omit transition rates induced by thermal electrons for transitions calculated with hot electrons and/or a non-thermal distribution.

## Export Controlled Information

**switch 176**      *default value = 0*

If switch(176) = 0, use Stewart-Pyatt continuum lowering. If switch(176) ≠ 0, use the total electron and ion densities in the ion sphere expression to limit to Ecker-Kroll continuum lowering.

**switch 178**      *default value = 0*

If switch(178) = 0, add a single iterative improvement pass to each direct solution of the rate equations. If switch(178) > 0, do not allocate memory for a second rate matrix and do not use iterative improvement. If switch(178) < 0, allocate memory for a second rate matrix, but do not use it for iterative improvement.

**switch 179**      *default value = 0*

If switch(179) ≠ 0, minimize the memory used in the atomic kinetics.

**switch 180**      *default value = 0*

If switch(180) ≠ 0, iterate the kinetics calculation for each zone to obtain a converged solution for the populations. This iteration is independent of, and in addition to, the timestep iterations controlled by switch(44).

If switch(180) < 0, iterate a total of abs(switch(180)) times for each kinetics evaluation.

If switch(180) > 0, iterate a maximum of switch(180) times for each kinetics evaluation.

**switch 181**      *default value = 0*

Control the calculation of charge exchange rates.

If switch(181) < 0, evaluate all rates, including charge exchange, on each kinetics iteration.

If switch(181) > 0, keep other rates fixed and re-evaluate only charge exchange rates on each kinetics iteration.

Control how charge exchange rates are evaluated. Each digit of switch(181) controls a different option:

1's digit: choose the model for evaluating charge exchange rates

*i* = 0: none

*i* = 1: use the classical overbarrier model

V.N. Ostrtovsky, J. Phys. B 28, 3901-3914 (1995)

F. Sattin, Phys Rev A 62, 042711 (2000)

F. Sattin, Phys Rev A 64, 034704 (2001)

*i* = 2: use a constant cross-section of  $10^{-15} Z \text{ cm}^2$  for an ion of charge  $Z$

10's digit: choose whether to include charge exchange rates between charge states of the same element

0: yes

>0: no

**switch 194**      *default value = 0*

If switch(194) ≠ 0, return LTE EOS information from the atomic kinetics. This option is included to aid in investigations of NLTE effects.

# Export Controlled Information

**param 26**      *default value = 4.00*

Maximum fractional change allowed in electron densities between iterations controlled by switch(124).

**param 32**      *default value = 0.00*

Minimum RMS difference between QLTE and QSS isosequence distributions required to calculate NLTE populations, or the minimum difference between steady-state and time-dependent QSS isosequence distributions required to calculate time-dependent populations instead of using a steady-state table lookup.

**param 33**      *default value = 0.00*

Evaluate the atomic kinetics in LTE if the electron temperature is less than param(33).

**param 39**      *default value = 0.00*

Minimum value of <Z> to use for timestep control.

**param 47**      *default value = 1.00*

Multiplier applied to the Planckian radiation intensities set by switch(20).

**param 56**      *default value = 1.00e-2*

Convergence criterion for isosequence populations when iterating timesteps.

**param 60**      *default value = 1.00e-6*

Isosequences with fractional population greater than param(60) will be considered when testing convergence of populations when iterating timesteps.

**param 61**      *default value = 1.00e-10*

If the fractional population of an isosequence falls below param(61), that isosequence will not be in the kinetics calculations (for that zone). Judicious use of this parameter can greatly speed up problems where the ionization balance varies widely over time or space, or where the extent of the atomic model is larger than necessary. The window of isosequences which is included can be monitored with the edits **isomin** and **isomax**.

**param 62**      *default value = 1.00*

When calculating natural line widths, each transition can contribute no more than param(62) times the transition energy.

**param 66**      *default value = 1.00*

Multiplier for rates from a *coes* section.

**param 67**      *default value = 1.00*

Multiplier for rates from a *coll excite* or *colex* section.

# Export Controlled Information

**param 68**            *default value = 1.00*

Multiplier for rates from a *coll ionize* or *colon* section.

**param 69**            *default value = 1.00*

Multiplier for rates from a *cois* section.

**param 70**            *default value = 1.00*

Multiplier for rates from a *phxs*, *phxs1* or *rad\_rate* section.

**param 71**            *default value = 1.00*

Multiplier for rates from a *phis* section.

**param 72**            *default value = 1.00*

Multiplier for rates from an *augxs* or *aut\_rate* section.

**param 73**            *default value = 1.00*

Multiplier for total rates from an *augis* section.

**param 74**            *default value = 1.00*

Multiplier for rates from a *rec* subsection of an *augis* section.

**param 75**            *default value = 1.00*

Multiplier for rates from a *rad* subsection of an *augis* section.

**param 76**            *default value = 1.00*

Multiplier for rates from a *aug* subsection of an *augis* section.

**param 77**            *default value = 1.00*

Multiplier for rates from a *cex* subsection of an *augis* section.

**param 79**            *default value = -2.00*

When using continuum lowering, the degeneracy of a state will be decreased by a factor  
 $\exp(-g \text{ param}(79))$

where

$$g = [(\Delta E_{\max} - \Delta E) / \Delta E_{\max}]$$

and

$\Delta E$  is the energy lowering calculated by the Stewart-Pyatt formula,  
 $\Delta E_{\max}$  is the energy lowering necessary for the state to disappear.

**param 89**            *default value = 1.00*

Multiplier for  $\Delta E$  calculated by continuum lowering

## Export Controlled Information

**param 95**      *default value = 0.00*

Minimum electron density used to calculate transition rates, expressed as a fraction of the total ion density for each zone (i.e. a minimum  $\langle Z \rangle$ ). When dealing with a completely neutral plasma, it can sometimes be helpful to set this to a small number to get things started.

**param 96**      *default value = 0.50*

Relaxation parameter for electron density iterations if switch(124) = 0. The value used for the next iteration will be

$$n_e^{i+1} = [1-\text{param}(96)] n_e^* + \text{param}(96) n_e^i$$

where  $n_e^*$  is the value obtained from the current kinetics calculation.

**param 107**      *default value = 1.00e-10*

Minimum fractional population in an element necessary for considering convergence. Elements with smaller fractional population than param(107) will not affect timestep iterations.

**param 108**      *default value = 0.20*

Maximum fractional intensity change used to calculate changes in absorption and emission when using the linear response model.

**param 109**      *default value = 1.00e-12*

Convergence criterion when using an iterative kinetics solver.

**param 111**      *default value = 0.90*

Maximum fractional change allowed in absorption and emission when using the linear response model.

**param 114**      *default value = 3.50*

Telescoping factor for projective integration.

**param 119**      *default value = 0.00*

Minimum (negative of) rate matrix diagonal to use when solving for steady-state populations.

**param 124**      *default value = 0.00*

Minimum population for which a kinetics calculation will be done, or for which a sourced population will be included.

**param 125**      *default value = 1.00*

Multiplier for UTA and configuration broadening of photoexcitations.

**param 130**      *default value = 1.00e-2*

Maximum allowed virtual population, relative to the population for the isosequence which contains the virtual state. Virtual populations larger than this will be capped at this value.

## Export Controlled Information

**param 131**      *default value = 0.00*

If param(113)  $\neq$  0., it denotes the maximum transition rate which will not produce a warning (if param(131)  $<$  0.) or error (if param(131)  $>$  0.). The transitions which have rates larger than this value will be identified in the warning/error messages. This can be useful in identifying anomalous transition rates.

**param 132**      *default value = 0.00*

Multiplier on the electron-neutral scattering cross section used for calculating the average electron-ion collision time ( $2e-14 \text{ cm}^2$ ).

**param 134**      *default value = 1.00*

Multiplier for rates from a *coll excite* or *colex* section induced by hot electrons or a non-thermal electron distribution.

**param 135**      *default value = 1.00*

Multiplier for ionization rates from a *coll ionize* or *colon* section induced by hot electrons or a non-thermal electron distribution.

**param 136**      *default value = 0.00*

Multiplier for recombination rates from a *coll ionize* or *colon* section induced by hot electrons or a non-thermal electron distribution.

**param 137**      *default value = 1.00*

Multiplier for recombination rates other than collisional induced by hot electrons or a non-thermal electron distribution.

**param 138**      *default value = 0.00*

Multiplier for charge exchange rates.

**param 147**      *default value = 0.00*

Multiplier for level widths used to modify collisional rates.

**param 149**      *default value = 1.00e-3*

Photoexcitation rates and associated quantities will be evaluated for a given transition by integrating the line profile over the continuum bins unless the maximum fractional contribution of that transition to the absorption or emission (compared to continuum quantities) is less than param(149), in which case the transition will be evaluated with a delta function profile at the transition energy.

### Atomic Models:

The following switches and parameters control the generation of screened hydrogenic atomic models:

# Export Controlled Information

**switch 126**      *default value = 5*

Default value for maximum principal quantum number for states with inner shell holes. This value can be overridden separately for each isosequence with the **isorange** command.

**switch 127**      *default value = 4*

Default value for maximum number of shells which will have inner shell holes. The first such shell will be the first filled shell below the last shell. Successive shells will have progressively lower principal quantum numbers. The total number of shells included is given by the 1's digit. The number of shells with holes as normal states is given the 10's digits, while the remainder will be included as virtual states through augis rate sections. These values can be overridden separately for each isosequence with the **isorange** command.

**switch 128**      *default value = 10*

Default value for maximum principal quantum number to use with excited states. This value can be overridden separately for each isosequence with the **isorange** command.

**switch 129**      *default value = 1*

Control treatment of transitions going through virtual states. The doubly-excited and inner-shell states controlled by this switch are those defined implicitly through these transitions.

If switch(129) < 0, do not include these transitions.

If switch(129) = 0, do not include transitions through virtual doubly-excited states.

If switch(129) = 1, construct an **augis** section using radiative excitation and ionization, collisional excitation and ionization, and autoionization rates.

If switch(129) = 2, construct an **augis** section as above, but omit collisional rates.

If switch(129) = 3, construct an **augis** section as above, but omit autonization rates.

If switch(129) = 4, construct an **augis** section as above, but omit radiative rates.

For other values of switch(129), do not include these transitions.

**switch 130**      *default value = 1*

If switch(130) < -1, include  $\Delta n=0$  transitions, using tabulated rates from D.E. Post, R.V. Jensen, C.B. Tarter, W.H. Grasberger and W.A. Lokke, Atomic Data and Nuclear Data Tables **20**, p. 397, 1977.

If switch(130) = -1, include  $\Delta n=0$  transitions. For  $Z \leq 18$ , use corrected tabulated rates taken from ADPAK. For  $Z > 18$ , use the tabulated rates from Post, et al.

If switch(130) = 0, do not include  $\Delta n=0$  transitions.

If switch(130) > 0, include  $\Delta n=0$  transitions for ground configurations and singly-excited levels for up to switch(130) principal quantum numbers, starting with that of the ground configuration. Levels characterized only by principal quantum numbers will be included, as split levels will include these transitions in other data sections. Level splittings and oscillator strengths by orbital angular momentum quantum number are obtained from tabulated values provided by B. Wilson.

# Export Controlled Information

**switch 133**      *default value = 2*

Control options for generating screened hydrogenic data. Each digit of switch(133) controls a different option:

1's digit: smooth energy levels to approach ionization limit starting at level  $i$  above ground configuration

- $i = 0$ : false
- $i > 0$ : true

10's digit: collisional excitation rates

- 0: use default (denoted with \*)
- 1: use Mewe's formula
- 2: use van Regemorter's formula
- \* 3: use JJATOM fits from H.-K. Chung, R.W. Lee, M.H. Chen, HEDP **3**, 342 (2007)
- 4: use fits to Sampson and Golden results
- other: use default

100's digit: collisional ionization rates

- 0: use default (denoted with \*)
- 1: use a Lotz formula
- 2: use a Mott formula
- \* 3: use fits to Sampson and Golden results
- 4: use CAM values from R.E.H. Clark, J. Abdallah, Jr., J.B. Mann, ApJ **381**, 597 (1991)
- 5: use BCF values from Q.Y. van den Berg, *et al.*, PRL **120**, 0550002 (2018) in CAM format
- other: use default

$10^3$ 's digit: ion collisional rates

- 0: do not include
- $\neq 0$ : include neutral-neutral ionization rates for Ar, Kr or Xe  
(ref. A.J. Kelly, J Chem Phys 45, 1723, 1966)

$10^4$ 's digit: photoionization transitions

- 0: include usual set of transitions,  $\neq 0$ : include all inner shell transitions through augis rates

$10^5$ 's digit: ionization energies

- 0: do not adjust
- 1: adjust to match Scofield tables
- 2: adjust to match LIMBO data (as used for term splitting)
- $>2$ : do not adjust

$10^6$ 's digit: ionization oscillator strengths

- 0: adjust to more nearly match tabulated values from H.-K. Chung,  $\neq 0$ : use hydrogenic values

$10^7$ 's digit: collisional excitations for

- $\neq 0$ : include non-dipole excitations using Mewe formula

**switch 134**      *default value = 0*

Further options for generating screened hydrogenic data. Each digit of switch(134) controls a different option:

1's digit: choose screening constants to use for constructing energy levels for principal quantum

# Export Controlled Information

number models:

- 0: use recent constants from R. More.
- 1: use constants from R.M. Marchand, S. Caille and Y.T. Lee, JQSRT **43**, p. 149, 1990.
- 2: use constants from R.M. More, JQSRT **27**, p. 345, 1982.

If any isosequence uses orbital angular momentum quantum numbers, or if **modeltype fly** is specified, the screening constants from G. Faussurier, C. Blancard, A. Decoster, JQSRT **58**, p. 233, 1997 will be used for all isosequences.

10's digit: fits to collisional and photoionization data

- 0: leave data in screened-hydrogenic forms
- 1: produce standard fits to rates (suitable for use with DCA)
- 2: account for missing principal quantum numbers in standard fits for collisional ionizations

$10^3$ 's digit: isoelectron sequence summaries

- 0: do not save
- 1: write final model sizes by isoelectronic sequence to *.tbl* file
- >1: write estimated and final model sizes by isoelectronic sequence to *.tbl* file

$10^4$ 's digit: control printing of virtual state occupations

- 0: do not include in ascii model
- >0: include in ascii model

$10^5$ 's digit: control printing of atomic level widths (configuration broadening)

- 0: do not include in ascii model
- >0: include in ascii model

**switch 160**      *default value = 6*

Default for maximum principal quantum numbers for term splitting. Data is available for term splitting up to n=7.

**switch 164**      *default value = 0*

If switch(164) ≠ 0, multiply-excited states will be included for all isosequences with iso >= switch(164).

**switch 165**      *default value = 0*

If switch(165) ≠ 0, states with multiple inner shell holes will be included for all isosequences with iso >= switch(165).

**switch 166**      *default value = 11*

If switch(166) ≠ 0, doubly-excited states with one excitation from an inner shell will be included for all isosequences with iso >= switch(166).

**switch 167**      *default value = 5*

If switch(167) ≠ 0, all but the highest excitation of a multiply-excited state will have a maximum principal quantum number up to switch(167). If switch(167) = 0, all excitations will have the same maximum principal quantum number, specified by switch(128) or by an **isorange** command.

# Export Controlled Information

**switch 168**      *default value = 0*

Options for using term data:

1's digit: photoionizations:

- 0: do not split.
- 1: use a single transition with threshold energy adjusted to that of the lowest term.
- 2: use multiple transitions.

10's digit: collisional excitations

- 0: do not split
- 1: split as photoexcitations

100's digit: collisional ionizations

- 0: do not split.
- 1: use a single transition with threshold energy adjusted to that of the lowest term.
- 2: use multiple transitions.

$10^3$ 's digit: configuration broadening

- 0: do not use term data
- 1: use term data

$10^4$ 's digit: oscillator strengths

- 0: do not use term data
- >0: use term data

**switch 187**      *default value = 2*

Generate excited states with up to switch(187) excited electrons.

**switch 188**      *default value = 11*

Generate excited states with up to K excitations from the K-shell and up to L excitations from the L-shell, where switch(188) = K +  $10^*L$ .

**param 104**      *default value = 1.00*

Minimum temperature for standard fits to collisional rates (eV).

**param 105**      *default value = 1.00e4*

Maximum temperature for standard fits to collisional rates (eV).

**param 112**      *default value = 0.50*

Multiplier for configuration broadening of levels, calculated from screened-hydrogenic energy levels.

**param 126**      *default value = 1.00*

**param 127**      *default value = 0.30*

**param 128**      *default value = 1.00*

If one or more of these parameters is non-zero, configuration broadening for the transition from level *i* to level *j* within isosequence *iso* is calculated as the minimum of the non-zero values of

## Export Controlled Information

param(126) ( $W(i) + W(j)$ )  
param(127) ( $E(j) - E(i)$ )  
param(128) ( $Z - iso$ )

where  $W(i)$  is the configuration broadening of level  $i$ ,  $E(i)$  is the energy of level  $i$ , and  $Z$  is the atomic number.

**param 146**      *default value = 1.00*

Multiplier for level widths used in constructing autoionization rates.

### Radiation Transfer:

CRETIN separates radiation into two components: continuum and lines. The continuum radiation is treated with formal transfer only, i.e. the radiation intensities are calculated using given, fixed opacities and emissivities. Angular scattering (but not energy scattering) off electrons is included, with the correct angular distribution in 1-d and in an isotropic approximation in 2-d. The line transfer enforces consistency between populations and line strengths through a complete linearization procedure which can handle multiple overlapping and interacting lines. Linearization is applied to individual lines or small groups of lines identified with **resonant**, **resonance**, or **joinline** commands.

In 1-d, Feautrier and integral formalisms are available. Both are applied along long characteristics (“rays”) in planar, cylindrical, spherical or “wedge” geometries. In 2-d and 3-d, only the integral formalism is available, applied along short characteristics for a fixed angle set, i.e. an  $S_N$  (discrete ordinates) method.

The following switches and parameters apply to all radiation transfer:

**switch 8**      *default value = 1*

Controls choice of angle set for radiation transfer:

- 1-d: use gaussian angles on  $[-\pi/2, \pi/2]$  if switch(8) = 0
  - use double-gaussian angles on  $[0, \pi/2]$  if switch(8)  $\neq 0$
- 2-d: use Carlson A angles if switch(8) = 0
  - use Carlson B angles if switch(8) = 1
  - use product ray set based on gaussian angles if switch(8) = 2
  - use a product ray set based on double-gaussian angles if switch(8) = 3 or switch(8)  $< 0$
  - use a level symmetry ray set if switch(8) = 4 (valid for nmu  $\leq 10$ )
- 3-d: same as for 2-d

**switch 34**      *default value = 0*

Controls symmetry boundary conditions:

If 1-d: non-symmetric if switch(34) = 0  
symmetric about r=0. if switch(34)  $\neq 0$

If 2-d:       $k_{sym} = \text{int}(\text{switch}(34)/10)$       (10's position)  
                 $l_{sym} = \text{mod}(\text{switch}(34), 10)$       ( 1's position)  
non-symmetric if switch(34) = 0  
symmetric about k = 1 if  $k_{sym} = 1$  or 3  
symmetric about k =  $k_{max}$  if  $k_{sym} = 2$  or 3

## Export Controlled Information

symmetric about  $l = 1$  if  $lsym = 1$  or 3  
symmetric about  $l = lmax$  if  $lsym = 2$  or 3

If 3-d:  
ksym = int(switch(34)/100) (100's position)  
lsym = int(mod(switch(34),100)/10) ( 10's position)  
msym = mod(switch(34),100) ( 1's position)  
non-symmetric if switch(34) = 0  
symmetric about  $k = 1$  if ksym = 1 or 3  
symmetric about  $k = kmax$  if ksym = 2 or 3  
symmetric about  $l = 1$  if lsym = 1 or 3  
symmetric about  $l = lmax$  if lsym = 2 or 3  
symmetric about  $m = 1$  if msym = 1 or 3  
symmetric about  $m = mmax$  if msym = 2 or 3

For 2-d and 3-d problems, symmetry boundary conditions can also be set with a **boundary** command.

**switch 74**      *default value = .0*

If a rayset error occurs in a 2-d/3-d simulation, skip the radiation transfer calculation and continue the simulation if switch(74)  $\neq 0$ .

**switch 77**      *default value = .0*

Don't allow negative continuum opacities if switch(77)  $\neq 0$ .

**switch 93**      *default value = .0*

Include re-entrant radiation in 2-d and 3-d if switch(93)  $\neq 0$ .

**switch 94**      *default value = .0*

Use a scratch file for storing 2-d and 3-d ray information if switch(94)  $\neq 0$ .

**switch 104**      *default value = .0*

Use same memory for continuum, line, and spectrum work spaces if switch(104)  $\neq 0$ .

**switch 109**      *default value = 0*

Allocate memory but do no work for transfer calculations if switch(109)  $\neq 0$ .

**param 52**      *default value = 4.00*

For the integral formalism, the maximum ratio of optical depths for two line segments to be used for a 2nd-order fit. If the ratio exceeds param(52), a 1st-order fit will be used instead.

### Continuum Radiation:

Relevant switches and parameters:

# Export Controlled Information

**switch 7**      *default value = 0*

If switch(7) = 0, use opacities and emissivities calculated by the atomic kinetics.

If switch(7) ≠ 0, use analytic opacities as specified by **opacity** commands. The emissivities will be given by the Planck function times the opacities.

**switch 9**      *default value = .0*

If switch(9) = 0, use geometric mean frequencies obtained from the specified group boundaries.

If switch(9) ≠ 0, use arithmetic mean frequencies.

**switch 17**      *default value = 1*

Controls contribution of photoexcitations to opacities and emissivities used in continuum transfer:  
don't include any contributions if switch(17) = 0  
include contributions from all transitions if switch(17) < 0  
include contributions from all transitions not identified as lines if switch(17) > 0.

If switch(17) = ±1, the contributions from a given transition will be averaged over each continuum bin failing with the transition profile. Otherwise, all contributions from the transition will be averaged over the continuum bin containing the transition energy.

**switch 23**      *default value = 0*

Controls choice of angular dependence of electron scattering:  
assume isotropic scattering if switch(23) = 0  
assume dipole phase function  $(1 + \cos^2 \theta)$  if switch(23) = 1

**switch 27**      *default value = 0*

Controls first-flight radiation transport. This option assumes that radiation with intensity *jbn2* is emitted normally from the surface of a sphere of radius param(29) located at the node given by switch(27). This radiation is tracked through the mesh with attenuation but no emission. The resulting radiation field *jnu2* is included in the total radiation field *jnu* for all other purposes.

If switch(27) = 0, do not do first-flight transport.

If switch(27) > 0, do steady-state first-flight transport.

If switch(27) < 0, do time-dependent first-flight transport.

**switch 35**      *default value = 0*

Controls contribution of free-free opacities and emissivities (bremsstrahlung) used in continuum transfer and laser absorption:  
calculate values using a coulomb logarithm if switch(35) < 0  
calculate values using a Gaunt factor if switch(35) = 0  
do not include free-free contributions if switch(35) > 0.

**switch 36**      *default value = 0*

Main control: continuum transfer calculations will be done only if switch(36) ≠ 0, using Feautrier

## Export Controlled Information

formalism if switch(36) = 1, and using integral formalism, otherwise. See switch(100) for alternative algorithms.

**switch 39**      *default value = 1*

Maximum number of iterations used to include angular scattering in continuum transfer.

**switch 77**      *default value = 1*

Do not allow negative continuum absorption coefficients if switch(77) ≠ 0.

**switch 100**      *default value = 0*

Use rad package routines which evolve temperature along with the continuum radiation if switch(100) ≠ 0 (see rad package documentation for options). Do not set both switch(36) and switch(100) to non-zero values. Setting switch(100) to a non-zero value will set switch(31) = 0.

**switch 156**      *default value = 0*

Do not iterate the continuum radiation transfer when iterating timesteps if switch(156) ≠ 0.

**switch 174**      *default value = 0*

Produce a radiation field by accumulating locally emitted photons if switch(174) ≠ 0. This only applies if continuum transfer is being used, i.e. switch(27) = 0, switch(36) = 0, switch(100) = 0.

**switch 195**      *default value = 0*

If switch(195) > 0, broaden continuum edges.

If switch(195) = 1, convolve the contribution from each transition with a Gaussian with a width including Doppler, collisional and Stark broadening.

If switch(195) = 2, convolve the contribution from each isosequence with the microfield for that isosequence.

If switch(54) = 3, convolve the continuum with a Voigt lineshape with a width appropriate for the last discrete Stark-broadened line (Inglis-Teller limit). This option applies only to spectral calculations.

**param 1**      *default value = 1.00*

Multiplier for analytic opacities. This value is appropriate for free-free absorption by a fully-ionized plasma with Z=1.

**param 2**      *default value = 2.00*

Density exponent for analytic opacities.

**param 3**      *default value = -3.50*

Temperature exponent for analytic opacities.

**param 4**      *default value = 1.00*

## Export Controlled Information

Multiplier on bound-free and bound-bound opacities and emissivities if switch(7) = 0.

Frequency exponent for analytic opacities if switch(7) ≠ 0.

**param 5** *default value = 1.00*

Multiplier for electron scattering (Klein-Nishina) cross-sections.

**param 6** *default value = 1.00*

Multiplier for analytic opacities for energies above the value given by param(7).

**param 7** *default value = 0.00*

Threshold energy for applying additional multiplier to analytic opacities.

**param 11** *default value = 0.00*

If param(11) ≠ 0.0, use param(11) as the free-free Gaunt factor. This can be useful for matching analytic solutions.

**param 24** *default value = 1.00*

Multiplier on radiation pressure.

**param 27** *default value = 1.00*

Multiplier on the collisional frequency used to limit low-frequency bremsstrahlung (Drude model).

**param 28** *default value = 1.00e-2*

Convergence criterion for radiation intensities when iterating continuum transfer.

**param 29** *default value = 0.00*

Source radius for first-flight transport – see switch(27).

**param 63** *default value = 1.00*

Multiplier on free-free opacities and emissivities.

**param 113** *default value = 1.0*

Multiplier for Doppler and collisional contributions to continuum broadening controlled by switch(195).

### Spectral Radiation:

Spectral radiation includes contributions from both continuum transitions (bound-free and free-free), but from all photoexcitations (bound-bound), as well. The spectral calculation uses a set of energy bins distinct from that used by the atomic kinetics and continuum transfer, and distinct from those used by the line transfer. The spectral bins may be very finely spaced to resolve many transitions without incurring extra expense in the kinetics calculations.

# Export Controlled Information

The spectral radiation is treated with a formal transfer only, using populations calculated by the kinetics and augmented by corrections calculated by the line transfer. The results of the spectral calculations are used for edit purposes only, and do not affect the atomic kinetics.

Relevant switches and parameters:

(**param 81, 82, 83, 91, 92** and **switch 72, 73** under [Output](#), and **switch 86, 87, 88, 90** under [Lineshapes](#))

**switch 52**      *default value = 0*

Include Stark broadening if switch(52)  $\neq 0$ . Individual lineshapes will still be Voigt profiles, but both the doppler width and natural width will have components due to Stark broadening. This switch does not affect transitions which are treated with TOTAL except for line transfer.

If switch(52)  $< 0$ , include Stark broadening for all photoexcitations.

If switch(52)  $> 0$ , include Stark broadening only for photoexcitations designated as “lines”.

**switch 53**      *default value = 0*

If switch(53)  $< -2$ , do not perform any spectral calculations.

If switch(53) = -2, calculate detailed spectral opacities and emissivities assuming a black-body radiation field at the local Tr, but do not perform a transfer calculation.

If switch(53) = -1, calculate detailed spectral opacities and emissivities, but do not perform a transfer calculation. The opacities and emissivities can be edited or sent to a dump file.

If switch(53) = 0, use only specified edit rays (and others as necessary) for the transfer calculation.

If switch(53) = 1, use all rays for the transfer calculation.

If switch(53)  $> 1$ , use all rays and include coherent electron scattering in the transfer calculation.

**switch 54**      *default value = 0*

If switch(54)  $< 0$ , include only special transitions (i.e. lines and Stark transitions) in the spectral calculation.

If switch(54) = 0, include continuum opacities and emissivities in the spectral calculation, with values obtained by interpolation from the continuum groups.

If switch(54)  $> 0$ , include only contributions from photoexcitations.

**switch 63**      *default value = 1*

Include effects of doppler shifts in spectral calculation if switch(63)  $\neq 0$ .

**switch 92**      *default value = 1*

Calculate spectral opacities and emissivities in chunks if switch(92) = 0, or calculate all frequencies together if switch(92)  $> 0$ . Using chunks requires more time, but less memory. If switch(92)  $< 0$ , do not calculate this information but use continuum opacities and emissivities if switch(92) = -1, or read it from existing dump files, otherwise.

# Export Controlled Information

**switch 125**      *default value = 0*

Average contributions of photoexcitations to opacities and emissivities over the spectral bin including the transition energy switch(125)  $\neq 0$ . This guarantees that the integrated emission will be correct, even without resolving the individual transition profiles.

**switch 159**      *default value = 0*

If switch(159)  $\neq 0$ , exclude certain transitions from opacities and emissivities according to the digits.

1's digit: exclude photoexcitations if  $\neq 0$

10's digit: exclude photoionizations if  $\neq 0$

100's digit : exclude augxs transitions if  $\neq 0$

1000's digit: exclude augis transitions if  $\neq 0$

10000's digit: exclude bremsstrahlung if  $\neq 0$

**switch 195**      *default value = 0*

If switch(195)  $> 0$ , broaden continuum edges.

If switch(195) = 1, convolve the contribution from each transition with a Gaussian with a width including Doppler, collisional and Stark broadening.

If switch(195) = 2, convolve the contribution from each isosequence with the microfield for that isosequence.

If switch(54) = 3, convolve the continuum with a Voigt lineshape with a width appropriate for the last discrete Stark-broadened line (Inglis-Teller limit). This option applies only to spectral calculations.

**param 78**      *default value = 1.e3*

The contribution of each photoexcitation to the opacity and emissivity will be included out to a minimum of param(78) Lorentz widths (or param(98) Doppler widths, if greater) from the transition center.

**param 81**      *default value = 0.50*

Limit the total natural width for each photoexcitation to be at most param(81) times the transition energy.

**param 85**      *default value = 1.0*

Multiplier for total opacity in spectral calculation.

**param 86**      *default value = 1.0*

Multiplier for total emissivity in spectral calculation.

# Export Controlled Information

**param 90**      *default value = 0.0 (eV)*

Minimum doppler width assigned to any photoexcitation in spectral calculation. This can be used to artificially broaden transitions to ensure that they will be resolved by the spectral binning.

**param 93**      *default value = 1.0*

Multiplier for line contributions to opacity and emissivity in spectral calculations (i.e. contributions due to photoexcitations designated as “lines”).

**param 98**      *default value = 6.*

The contribution of each photoexcitation to the opacity and emissivity will be included out to a minimum of param(98) Doppler widths (or param(78) Lorentz widths, if greater) from the transition center.

**param 113**      *default value = 1.0*

Multiplier for Doppler and collisional contributions to continuum broadening controlled by switch(195).

**param 121**      *default value = 1.0*

Multiplier for Doppler widths (including UTA contributions) of photoexcitations in spectral calculations.

**param 122**      *default value = 1.0*

Multiplier for natural widths of photoexcitations in spectral calculations.

**param 123**      *default value = 1.0*

Multiplier for Stark widths of photoexcitations in spectral calculations.

**param 133**      *default value = 1.0*

Multiplier for boundary condition values for spectral calculations.

## Lineshapes:

Selected radiative transitions can be given lineshapes calculated by the TOTAL code, if this was compiled into CRETIN. The radiative transitions can be lines, selected by the appropriate **linetype** command, or photoexcitations, selected by a **stark** command. TOTAL was developed at Universite de Provence at Marseille, France, under Bernard Talin, Roland Stamm and Louis Klein and was modified by Richard Lee. It calculates line profiles using a quasi-static ion microfield approximation and a binary electron impact collision model. Magnetic field effects were added by Mark Adams. A description of the code is given in the user manual, available from Richard Lee. The physics and computational approach are described in A. Calisti, *et al*, Phys. Rev. A **42**, 5433 (1990) and in M.L. Adams, *et al*, Phys. Rev. E **66**, 066413 (2002). Ion dynamics uses the FFM method from A. Calisti, *et al*, Phys. Rev. E **81**, 016406 (2010) as formulated in E. Stambulchik and Y. Maron, Phys. Rev. E **87**, 053108 (2013).

Relevant switches and parameters:

# Export Controlled Information

**switch 86**      *default value = 50*

Number of microfield values used by TOTAL in lineshape calculations.

**switch 87**      *default value = 0*

Grouping of transitions for TOTAL calculations will be done individually if switch(87) = 0, by principal quantum number if switch(87) = 1, and by isosequence if switch(87) = 2. Grouping transitions by isosequence requires all levels within that isosequence to have statistical weights of the form  $2J+1$ .

**switch 88**      *default value = 0*

Input options for TOTAL. Each digit of switch(88) controls a different option:

1's digit: eigenvalues

0: calculate real eigenvalues,  $\neq 0$ : calculate complex eigenvalues

10's digit: Doppler broadening

0: include,  $\neq 0$ : do not include

100's digit: natural width

0: use natural width as calculated by TOTAL,  $\neq 0$ : add broadening from Cretin

**switch 89**      *default value = 0*

Include line polarization shifts if switch(89)  $\neq 0$ . These shifts are currently available for Lyman lines only (ref. Nguyen, *et. al.*, Phys. Rev. A33, p. 1279, 1986).

**switch 114**      *default value = 0*

Input options for TOTAL. Each digit of switch(114) controls a different option:

1's digit: reduce option

0: none, 1: some, other: all

10's digit: onlyn

0: false,  $\neq 0$ : true

100's digit

0: false,  $\neq 0$ : true

1000's digit: nolowerb

0: false,  $\neq 0$ : true

**switch 115**      *default value = 0*

Controls angle-dependence of line profiles calculated by TOTAL when magnetic fields are present.

If switch(115)  $< 0$ , fix the cosine of the angle with respect to the magnetic field to have the value given by param(100).

If switch(115) = 0, ignore magnetic field effects.

## Export Controlled Information

If switch(115) = 1, calculate a fully angle-dependent line profile.

If switch(115) = 2, average parallel and perpendicular components.

If switch(115) = 3, calculate the parallel component only

If switch(115) = 4, calculate the perpendicular component only

If switch(115) = 5, calculate the pi component only

If switch(115) = 6, calculate the sigma-minus component only

If switch(115) = 7, calculate the sigma-plus component only

**switch 116**      *default value = 0*

Coupling scheme used by TOTAL to interpret atomic data. The LS scheme requires the magnetic dipole operator to be present in the atomic datafile to include magnetic field effects. The JJ scheme requires no additional information.

If switch(116) < 0, use the LS scheme and align the microfield along the z-axis / magnetic field. This behavior duplicates the older (non-magnetic) version of TOTAL.

If switch(116) = 0, use the LS scheme.

If switch(116) > 0, use the JJ scheme.

**switch 117**      *default value = 0*

Ignore warning messages from TOTAL if switch(117) = 0. Otherwise, route them to the appropriate location (terminal or .tbl file).

**switch 118**      *default value = 0*

For each region, use a single line profile for each line using TOTAL with magnetic fields. The profile will be from the first zone calculated from that region and will be used for that line for all successive zones in the same region. This option may give different results when using threads or multiple processors as the first zone chosen may differ by thread or processor.

**switch 192**      *default value = 0*

Report on all TOTAL calls during spectral calculations if switch(192) ≠ 0. If switch(192) < 0, do not execute the TOTAL calls.

**param 100**      *default value = 0*

If switch(115) < 0, fix the cosine of the angle with respect to the magnetic field to have the value given by param(100).

**param 110**      *default value = 4.00*

The energy range considered when constructing the lineshape will be param(110) times the energy range used when selecting transitions.

# Export Controlled Information

**param 148**      *default value = 0.50*

The fluctuation frequency used for ion dynamics with the FFM model will be param(148) times  $\langle v \rangle / \langle r \rangle$  (ref. E. Stambulchik and Y. Maron, Phys. Rev. E **87**, 053108, 2013).

## Line Radiation:

“Lines” are photoexcitations which have been singled out for special treatment with a **line** command. The exact treatment for each line can be specified by a combination of **linetype** commands and switch and parameter settings. Individual lines are assumed to have a Voigt shape (even for transitions which have been designated for treatment by TOTAL) unless **linetype total** is specified. Both the doppler width and natural width may have components due to Stark broadening.

Relevant switches and parameters:

(see also **switch 57**, **param 62**, **param 81** under [Atomic Kinetics](#) and **switch 52** under [Spectral Radiation](#))

**switch 37**      *default value = 0*

Main control: line transfer calculations will be done only if switch(37)  $\neq 0$ , regardless of any other commands or settings.

**switch 38**      *default value = 0*

If switch(38)  $\neq 0$ , continuum quantities will be assumed to have constant values over the line profile, corresponding to line center. In this case, only half of a symmetric line will be transferred, if (1) the line is not doppler-shifted, and (2) the line does not overlap any other lines (with non-zero offset of line centers).

**switch 40**      *default value = 13*

Maximum number of iterations for the approximate operator linearization.

**switch 41**      *default value = 4*

Number of iterations for the approximate operator before using Ng acceleration.

**switch 62**      *default value = 0*

If switch(62)  $\neq 0$ , electron scattering will be included as a loss mechanism in the line transfer calculations, under the assumption that the energy shift due to the scattering would result in removal of the photon from the line core. This is a good approximation for isolated, doppler-broadened lines, but may fail under other conditions.

**switch 98**      *default value = 0*

If switch(98)  $\neq 0$ , line profiles will be recalculated (including Doppler shifts) for each angle. The default behavior is to interpolated Doppler-shifted line shapes from non-shifted line shapes.

**param 50**      *default value = 10.*

For complete redistribution (CRD), the tail of the line absorption profile can be analytically integrated

# Export Controlled Information

to allow for scattering outside of the energy range covered by the line bins, obviating the need to bin optically-thick lines out to many doppler widths. For this purpose, CRD line wings will be assumed to start param(10) doppler widths from line center, or at the last line bin value, whichever is further from line center.

**param 51**      *default value = 1.00e-3*

Convergence criterion for approximate operator iterations.

**param 57**      *default value = 1.00e-2*

Convergence criterion for line strengths when iterating timesteps.

**param 80**      *default value = 1.00e-2*

Minimum optical depth for performing linearization. Lines with a smaller optical depth than param(80) will use formal transfer only, regardless of **linetype** option.

## Escape Factors:

Escape factor treatments are available for all lines and photoexcitations. Escape factors will be used under the following conditions:

- (1) switch(33) is non-zero      and
- (2) the transition has not been specified as a “line” and switch(33) > 0,      or  
the transition has been specified as a “line” with linetype “escape”

The escape factors are non-local, in that an integrated column density of populations is used to calculate the optical depth. If the transition is a “line”, the upper and lower level populations are integrated across the mesh. If the transition is not a “line”, the iso-sequence population is integrated across the mesh and then scaled by the ratio of the local level populations to the local iso-sequence population. For a 0-d (1-zone) problem, the column density must be set by the user. For a 1-d problem, the column densities in both directions are considered when calculating escape factors (see switch(81)). For 2-d and 3-d problems, column densities along one axis are considered – see switch(154).

For a 0-d problem without consideration of velocity gradients, the escape factors are averaged over optical depth. Setting the problem geometry to “none” will result in a 0-d treatment. In this case, the mean  $\frac{1}{2}$ -chord of the plasma can be set with param(53) while the line of sight can be set with param(139) - see Phillips, *et al*, HEDP **4**, 18 (2008) for a discussion of these quantities.

The static escape factors (switch(33) =  $\pm 1$ ) are available for either planar or spherical geometry. A cylindrical problem will use the planar escape factors except for the special case mentioned below. Spherical geometry uses an approximation derived from formulas given in Mancini, *et al*, J. Phys. B **20**, 2975 (1987). A Voigt, Doppler or Lorentzian profile may be used, according to the value of switch(78). If the transition is a line, the lineshape specified by the linetype will be used. Continuum opacities may be included according to the value of switch(79). The default method used with this setting involves integrating the escape factor over the line profile. If switch(80) is negative and continuum opacities are not included, analytic formulas or tabulated factors will be used as follows:

planar geometry, Doppler profile:      Hummer, JQSRT **26**, 187 (1981)

planar geometry, Voigt profile :      Apruseze, JQSRT **34**, 447 (1985)

cylindrical geometry, Doppler profile:      Bhatia and Kastner, JQSRT **60**, 543 (1998).

## Export Controlled Information

The Sobolev escape factors (`switch(33) = ±2`) assume a doppler-broadened line with no continuum opacities. These escape factors are available for planar and cylindrical geometries. A spherical problem will use the cylindrical escape factors (ref: Shestakov and Eder, JQSRT **42**, 483, 1989).

The escape factors which interpolate between the static and Sobolev cases (`switch(33) = ±3`) also assume a doppler-broadened line with no continuum opacities (ref: Hummer and Rybicki, Ap. J. **254**, 767, 1982). These escape factors are available only for planar geometry.

The generalized escape factors (`switch(33) = ±4`) interpolate between the static and Sobolev limits. These escape factors are available for either planar or cylindrical geometry. A spherical problem will use the cylindrical escape factors. A Voigt, Doppler or Lorentzian profile may be used, according to the value of `switch(78)`. If the transition is a line, the lineshape specified by the linetype will be used.. Continuum opacities may be included according to the value of `switch(79)`.

In 2-d and 3-d geometries, the column densities used for escape factors are the minimum of those calculated along all logical coordinate axes. If `switch(154) ≠ 0`, it specifies the logical coordinate axis to use as if the problem were 1-d along the axis specified.

Relevant switches and parameters:

**switch 33**      *default value = 0*

Controls type of escape factor treatment used:

- none if `switch(33) = 0`
- static if `switch(33) = ±1`
- Sobolev if `switch(33) = ±2`
- interpolation between static and Sobolev if `switch(33) = ±3`
- generalized escape factor if `switch(33) = ±4`
- none, otherwise

Controls application of escape factors:

- apply to all photoexcitations if `switch(33) > 0`
- only apply to lines with linetype “escape” if `switch(33) < 0`

**switch 78**      *default value = 0*

Assume a Voigt profile if `switch(78) = 0`, or a Doppler profile otherwise.

**switch 79**      *default value = 0*

Include continuum opacities if `switch(79) = 0`.

**switch 80**      *default value = 0*

Controls 1-d geometry used for escape factors

- planar geometry +allow use of special formulas if `switch(80) < 0`
- problem geometry if `switch(80) = 0`
- planar if `switch(80) = 1`
- cylindrical if `switch(80) = 2`
- spherical if `switch(80) = 3`

**switch 81**      *default value = 0*

## Export Controlled Information

Use double-sided escape factors if switch(81) = 0, single-sided otherwise. Double-sided escape factors consider the column densities to boundaries in two directions independently. Single-sided escaped factors consider the column density to the nearest boundary and assume the density in the other direction is infinite.

**switch 82**      *default value = 0*

Calculate column densities used for escape factors by integrating line populations (for lines) or isoelectronic populations (for other transitions) if switch(82) = 0. If switch(82) > 0, multiply the local density times the distance to boundary. If switch(82) < 0, multiply the local density times the absolute value of param(53).

**switch 154**      *default value = 0*

For 2-d / 3-d geometries, choose the axis to use when calculating column densities.

use the axis which includes the minimum column density to boundary if switch(154) = 0:

use the k-axis if switch(154) = 1

use the l-axis if switch(154) = 2

use the z-axis if switch(154) = 3

If switch(154) = 4, column densities will be calculated in each direction used by a radiation transfer calculation, as specified by the **angles** command.

**param 53**      *default value = 0.0 (g/cm<sup>2</sup> or cm<sup>-2</sup> or cm)*

If param(53) ≠ 0, it specifies the column density or distance to boundary used for escape factor calculations. If switch(82)=0, then the value will be interpreted as a mass column density if param(53) > 0, and a number column density if param(53) < 0. If switch(82) < 0, the value will be interpreted as a distance to boundary if param(53) > 0 and a number column density of the appropriate isosequence if param(53) < 0.

**param 54**      *default value = 1.00e-2*

For (static) optical depths less than param(54), an escape factor of 1.0 will be assigned.

**param 139**      *default value = 0.0 (cm)*

If switch(82) < 0 and param(53) > 0, then either param(139) (if > 0) or param(53) specifies the line of sight used for 0-d escape factor edits.

### Electron Distribution:

CRETIN currently assumes that the bulk of the free electrons are in a thermal distribution and that free electrons produced by ionizations instantaneously thermalize. There are two mechanisms for augmenting the bulk thermal distribution with non-thermal electrons. The first is to specify additional hot electron distributions described by (**tehot**, **nehohot**) pairs. Each value of **tehot** gives either the temperature of a thermal distribution or the energy of a monoenergetic distribution, while **nehohot** gives the total electron density in that distribution. The second is to specify a distribution **edist** as a function of electron energy **eenergy**.

There may be a single bulk thermal distribution (**ne**, **te**), a single distribution (**edist** vs **eenergy**), and an arbitrary number of (**tehot**, **nehohot**) pairs simultaneously. All electrons will participate in collisional excitations, de-exitations, and ionizations, radiative recombination and bremsstrahlung. By default, only the bulk thermal

## Export Controlled Information

electrons and the thermal hot electron distributions currently participate in collisional recombinations. The arbitrary electron distribution can also participate in collisional recombinations, but this must be turned on by setting param(136) to a non-zero value. Highly energetic electrons rarely provide significant recombinations, so the additional computational expense is not justified for many applications. Currently, only the bulk thermal electrons transport energy via conduction.

The hot electrons and the electron distribution may be specified in one or more xfiles and/or through source commands. If electron energy bins are specified, the total electron distribution may be edited with **nedist**.

Relevant switches and parameters:

see **switch 163**, **param 134-136** under [Atomic Kinetics](#)

### Temperature Calculations:

CRETIN can self-consistently calculate the thermal evolution of a plasma, with heating rates determined by atomic processes. The temperature is calculated by implicitly solving a linearized energy balance equation, including electron thermal conduction, and a single timestep can be iterated to assure convergence of the full non-linear equation.

Relevant switches and parameters:

(see also **switch 34** under [Radiation Transfer](#) for symmetry boundary conditions)

**switch 16**                  *default value = 0*

If switch(16) = 0, use the coulomb logarithm from Lee & More (Phys Fluids 27, p. 1273, 1984).

If switch(16) =1, use the BPS coulomb logarithm from Brown & Singleton (LA-UR-07-2154, 2007).

For other values of switch(16), set the coulomb logarithm to 1.0.

**switch 31**                  *default value = 0*

Controls type of temperature calculation

- none if switch(31) = 0
- time-dependent if switch(31) > 0
- steady-state if switch(31) < 0

Controls form of heating rates used

- use electronic heating rates if switch(31) = ±1 or unlisted values
- use energy balance (using internal energy rates) if switch(31) = ±2
- use energy balance (using internal energy deltas) if switch(31) = ±3

If switch(31) > 3, use a simple implicit temperature update which mimics the temperature update used by many radiation transport packages. This updates each zone independently and does not include conduction. See switch(169) for options pertaining to this temperature update.

If switch(31) < -3, also update the radiation field to be consistent with the temperature update obtained with the method of switch(31)>3.

The option using the electronic heating rates considers only atomic processes which directly affect the

## Export Controlled Information

free electron distribution. Options using energy balance obtain the (rate of) change in the free electron energy as the difference between the net energy change from radiation absorption and emission less the energy change in the internal energy of the material. The (rate of) change in the internal energy is obtained from either atomic transition rates or from the finite difference over the timestep.

**switch 48**      *default value = 0*

Control printing of linear solver diagnostics for electron thermal conduction:

If switch(48) = 0, do not print diagnostics.

If switch(48) < 0, print diagnostics to the screen.

If switch(48) > 0, print diagnostics to the ascii output file.

**switch 49**      *default value = 0*

Control calculation of electron and ion thermal conduction

no thermal conduction if switch(49) = 0

include thermal conduction if switch(49) ≠ 0

If 2-d:

use iccg if switch(49) = 1

use ilur if switch(49) = 2

use gmres with diagonal preconditioning if switch(49) = 3

use gmres with iccg preconditioning if switch(49) = 4

use gmres with ilur preconditioning if switch(49) = 5

use gmres with no preconditioning otherwise

**switch 50**      *default value = 0*

Flux limit the electron and ion conduction calculations if switch(50) ≠ 0

**switch 51**      *default value = 0*

Control options for electron-ion collision frequencies and conduction coefficients. Each digit of switch(51) controls a different option:

1's digit: choose treatment of electron conduction coefficients

0: Spitzer-Harm

1: Lee & More (Phys Fluids 27, p. 1273, 1984)

2: 2-phase model (Lee-More with solid density asymptote)

>2: fixed value of 1.0 (see **regmult** command for multipliers)

10's digit: include ion term in electron-ion collision frequency

0: no

≠ 0: yes

100's digit: include a z-dependent correction term on (Lorenz plasma) conduction coefficients

0: no

≠ 0: yes

If switch(51) < 0, the thermal conduction coefficient will be appropriate for radiative conduction and other quantities will take on default values.

# Export Controlled Information

**switch 66**      *default value = 100*

The maximum number of iterations used by iccg (if switch(49) =  $\pm 1$ ).

**switch 67**      *default value = 10*

The maximum number of orthogonalizations used by gmres (if switch(49) =  $\pm 2$  or  $\pm 3$ ).

**switch 68**      *default value = 4*

The maximum number of restarts used by gmres. Set this to -1 for no restarts as 0 will result in the (default) value of 10 restarts.

**switch 83**      *default value = 0*

Include additional energy sources if switch(83)  $\neq 0$ .

For switch(83) = 1, the energy source is taken to be

$$H = \text{param}(94) \langle Z \rangle [1 - (r/r_{\max})^2] / T^{1.5} \quad (\text{erg/cm}^3/\text{s})$$

where T is the electron temperature in eV

For switch(83) = 2, the energy source is obtained from a 1-d magnetic field / current evolution calculation.

For switch(83) = 4, the energy source is the same as that for switch(83) = 2, less the energy which would be lost by free-free emission (multiplied by param(94)).

For switch(83) = 5, the energy source comes from DD, DT, and DHe<sup>3</sup> thermonuclear reactions using existing number densities of these isotopes and assuming local energy deposition of charged particles (ref NRL Plasma Formulary, p45).

**switch 84**      *default value = 0*

Zone-center the electron conduction coefficients if switch(84)  $\neq 0$ .

**switch 110**      *default value = 0*

Allocate memory but do no work for temperature calculation if switch(110)  $\neq 0$ .

**switch 157**      *default value = 0*

Redo the radiative heating rates with updated intensities after the continuum transfer calculation if switch(157)  $\neq 0$ . The radiative heating rates are used in the temperature calculation if switch(31) = 2 or 3. This option duplicates the operational order of a combined continuum radiation / temperature algorithm.

**switch 158**      *default value = 0*

Redo the radiative heating rates and temperature calculation with updated temperatures and fixed intensities if switch(158)  $\neq 0$  and the fractional temperature change for the cycle was more than

## Export Controlled Information

param(99). If switch(158) < 0, subcycle this calculation in time. This option incorporates additional atomic kinetics calculations for each set of updated temperatures.

**switch 169**      *default value = 1*

Options for the simple radiation transport temperature update.

1's digit: temperature update method

- 0: no temperature update
- 1: use d/dT of the source function only
- 2: use d/dT of the net absorption
- 3: same as (1) using an emission opacity
- 4: same as (2) using an emission opacity
- 5: same as (3) with a partial temperature update
- 6: same as (5) with reversed group ordering

other digits: maximum number of iterations used to converge the temperature

**param 30**      *default value = 0.00*

If param(30) > 0., it will be used as the implicitness parameter for the temperature calculation. The default behavior calculates an implicitness parameter for each zone to achieve maximum accuracy in the absence of conduction.

**param 31**      *default value = 2.00*

The minimum allowed value for the Coulomb logarithm is given by param(31).

**param 36**      *default value = 1.0*

Multiplier for the ion thermal conduction coefficient

**param 37**      *default value = 1.0*

Multiplier for the ion thermal conduction flux limiter

**param 48**      *default value = 1.0*

Multiplier for the electron-ion coupling coefficient

**param 55**      *default value = 1.0e-2*

Convergence criterion for electron and ion temperatures when iterating timesteps

**param 64**      *default value = 1.0*

Multiplier for the electron thermal conduction coefficient

**param 65**      *default value = 1.0*

Multiplier for the electron and ion thermal conduction flux limiters

# Export Controlled Information

**param 88**      *default value = 1.0e-2*

Convergence criterion for iccg and gmres iterations

**param 94**      *default value = 0.00*

Coefficient for energy source term – see **switch 83**

**param 99**      *default value = 0.00*

If param(99) > 1.0, it will be applied as the maximum ratio between temperature iterates for steady-state temperature calculations (switch(31) < 0).

**param 102**      *default value = 0.00*

Minimum allowable electron temperature (eV).

**param 103**      *default value = 0.00*

Minimum allowable ion temperature (eV).

**param 120**      *default value = 1.00*

Multiplier on electrical conductivity.

## Radiation transfer / Temperature evolution:

When the temperature of optically thick material depends strongly on the absorption and emission of radiation, the iteration between the continuum radiation transfer and the temperature calculation can be slow to converge. Algorithms which account for this strong coupling and simultaneously compute radiation fields and material temperatures can converge much more quickly. Most algorithms of this type are designed for LTE conditions only. The RAD package contains algorithms which operate under LTE and non-LTE conditions, coupled to radiation diffusion (in 1D or 2D) or to S<sub>N</sub> radiation transport (in 1D).

Relevant switches and parameters:

(note that the RAD package has its own set of switches and parameters)

**switch 100**      *default value = 0*

Use the RAD package for radiation transfer if switch(100) > 0.

**rswitch 1**      *default value = 1*

Controls choice of radiation transfer algorithm:

If 1-d

  do flux-limited diffusion if rswitch(1) > 0

  do transport using Feautrier formalism if switch(1) = -1

  do transport using integral formalism if switch(1) = -2

If 2-d, use flux-limited diffusion with an iterative diffusion solver:

# Export Controlled Information

use iccg if rswitch(1) =  $\pm 1$   
use ilur if rswitch(1) =  $\pm 2$   
use gmres with diagonal preconditioning if rswitch(1) =  $\pm 3$   
use gmres with iccg preconditioning if rswitch(1) =  $\pm 4$   
use gmres with ilur preconditioning if rswitch(1) =  $\pm 5$   
use gmres with no preconditioning otherwise  
use left preconditioning if rswitch(1) < 0, right preconditioning if > 0

**rswitch 2**      *default value = 0*

Controls temperature evolution:  
do not evolve temperatures if rswitch(2) < 0  
evolve temperatures if rswitch(2) = 0  
evolve  $T^4$  instead of T if doing diffusion and rswitch(2) > 0

**rswitch 3**      *default value = 100*

The maximum number of iterations to be used when calculating radiation intensities and temperatures is given by switch(3). If switch(3) < 0, do not update the source function for each new iteration.

**rswitch 4**      *default value = 1*

Controls multigroup acceleration method to be used:  
no acceleration (or direct solution for 1 group) if rswitch(4) = 0  
grey acceleration if rswitch(4) = 1  
direct multigroup acceleration if rswitch(4) = 2  
direct solution (1-d only) if rswitch(4) = 3  
diagonal ALI multigroup acceleration (1-d only) if rswitch(4) = 4

**rswitch 5**      *default value = 1*

Set the speed of light to be infinite if rswitch(5) = 0.

**rswitch 6**      *default value = 1*

Use flux-limiting when doing radiation diffusion if rswitch(6)  $\neq 0$ .

**rswitch 7**      *default value = 0*

Re-evaluate opacities on each iteration if > 0. This is only available for analytic or tabulated opacities.

**rswitch 8**      *default value = 0*

Controls centering of zone-centered opacities used for diffusion coefficient:  
use furnished values if rswitch(8) < 0  
use harmonic average of nodal values if rswitch(8) = 0  
use straight average of nodal values if rswitch(8) = 1  
use minimum of nodal values if rswitch(8) = 2  
other values are equivalent to the default value

**rswitch 9**      *default value = 1*

Re-evaluate material temperatures from energies after the algorithm has converged if rswitch(9)  $\neq 0$ . If

## Export Controlled Information

rswitch(9) = 2, re-evaluate material temperatures from energies after each iteration instead of using a linearized expression with the specific heat.

**rswitch 10**      *default value = 100*

The maximum number of iterations used by the iccg or ilur linear solvers.

**rswitch 11**      *default value = 10*

The maximum number of orthogonalizations used by gmres for the iterative diffusion solver.

**rswitch 12**      *default value = -1*

The maximum number of restarts used by gmres for the iterative diffusion solver. Set this to -1 for no restarts as 0 will result in the (default) value of 10 restarts.

**rswitch 13**      *default value = 0*

Control printing of linear solver diagnostics for the iterative diffusion solver:

If rswitch(13) = 0, do not print diagnostics.

If rswitch(13) < 0, print diagnostics to the screen.

If rswitch(13) > 0, print diagnostics to the ascii output file.

**rswitch 14**      *default value = 0*

Fix the ion specific heat to be that of an ideal gas, instead of that specified by the EOS, if rswitch(14) ≠ 0.

**rswitch 15**      *default value = 0*

Use Ng acceleration if rswitch(15) > 0, in which case the number of iterations between accelerations is given by max(rswitch(15),4).

**rswitch 20**      *default value = 0*

Assume LTE if rswitch(20) = 0.

If switch(20) 0, use a non-LTE treatment::

do not include derivatives w.r.t.  $J_v$  if rswitch(20) = 1  
include derivatives w.r.t.  $J_v$  if rswitch(20) = 2

**rswitch 21**      *default value = 0*

Controls choice of Compton scattering method.

If rswitch(21) = 1, solve a Fokker-Planck (Kompaneets) equation for Compton scattering.

Otherwise, do not do Compton scattering.

# Export Controlled Information

**rswitch 22**      *default value = 0*

Maximum number of iterations for Compton scattering.

**rswitch 23**      *default value = 0*

Compton scattering options by digit.

1's digit: exclude relativistic effects if  $\neq 0$

10's digit: choose Gause-Hermite integration order when calculating diffusion coefficient:  
0 : 4, 1: 6, 2: 8, 3: 12

**rparam 1**      *default value = 1.0*

Multiplier on radiation diffusion coefficients.

**rparam 2**      *default value = 1.0*

Multiplier on radiation-matter coupling coefficients.

**rparam 3**      *default value = 1.0e-4*

Convergence criterion for material temperatures.

**rparam 4**      *default value = 1.0e-4*

Convergence criterion when using an iterative solver.

**rparam 5**      *default value = 1.0*

Automatic  $\Delta t$  control: maximum fractional change in radiation energy density per timestep.

**rparam 6**      *default value = 0.1*

Mimum radiation energy density / material energy density for considering convergence.

**rparam 11**      *default value = 1.0*

Multiplier on  $\partial S_v / \partial J_v$  when using derivatives w.r.t.  $J_v$  in non-LTE.

**rparam 12**      *default value = 1.0*

Multiplier on  $\partial \alpha_v / \partial J_v$  when using derivatives w.r.t.  $J_v$  in non-LTE.

**rparam 13**      *default value = 1.0*

Multiplier on  $\partial E / \partial J_v$  when using derivatives w.r.t.  $J_v$  in non-LTE.

**rparam 14**      *default value = 0.95*

Maximum allowed value of  $|\partial S_v / \partial J_v|$  when using derivatives w.r.t.  $J_v$  in non-LTE.

# Export Controlled Information

**rparam 15**      *default value = 0.0*

If rparam(15) > 0, the material energy density will be taken to be rparam(15) times the LTE radiation energy density,  $E = \text{rparam}(15) \cdot T^4$ . This option is provided for running the class of test problems which require this form for the material energy.

**rparam 16**      *default value = 1.0*

Multiplier on  $\partial\alpha_v/\partial T$ .

**rparam 17**      *default value = 1.0e-10*

The minimum absorption coefficient used for constructing diffusion coefficients is given by rparam(17).

**rparam 18**      *default value = 1.0e-2*

Convergence criterion for temperature change due to Compton scattering.

**rparam 19**      *default value = 1.0*

Multiplier on stimulated Compton term.

## Hydrodynamics:

The hydrodynamics in CRETIN is purely Lagrangian and is available for 1-dimensional geometries only. The equation of state (EOS) is provided by the atomic kinetics calculations.

Relevant switches and parameters:

**switch 2**      *default value = 0*

Main hydrodynamics control (1-d only):

If switch(2) = 0, do not do hydrodynamics.

If switch(2) = 1, do Lagrangean hydrodynamics.

If switch(2) = 2, do ideal magnetohydrodynamics

If switch(2) = 3, do resistive magnetohydrodynamics

If switch(2) < 0, evolve densities according to velocities.

**switch 3**      *default value = 100*

If the hydrodynamic timestep is smaller than the timestep required by the other algorithms, the hydrodynamics will use a maximum of switch(3) subcycles per timestep. If this number is exceeded, execution will stop.

# Export Controlled Information

**switch 4** *default value = 0*

Calculate hydrodynamic work using Fdu if switch(4) = 0, pdV if switch(4) ≠ 0.

**switch 5** *default value = 0*

Gradients in artificial viscosities will be limited to preserve monotonicity if switch(5) = 0.

**switch 6** *default value = 0*

Calculations of artificial viscosities will use a zonal mass density if switch(6) = 0, or a nodal mass density if switch(6) ≠ 0.

**param 18** *default value = 0.1*

In quiet start zones, allow hydrodynamic motion when the density changes by more than param(18) times its initial value.

**param 19** *default value = 0.1*

In quiet start zones, allow hydrodynamic motion when the ion temperature increases by more than param(19) times its initial value.

**param 20** *default value = 0.5*

CFL multiplier for timestep control: Δt will be limited to param(20) times Δx/c<sub>s</sub>.

**param 21** *default value = 0.5*

Multiplier α on the linear term of the artificial viscosity Q, which has the form:

$$Q = \rho [ \alpha c_s |\Delta v| + \beta (\Delta v)^2 ]$$

where ρ is the mass density, c<sub>s</sub> is the sound speed, and Δv is the velocity differential.

**param 22** *default value = 0.75*

Multiplier β on the quadratic term of the artificial viscosity Q.

**param 23** *default value = 0.0*

If param(23) > 0, the hydrodynamics will use an ideal gas equation of state with gamma = param(23).

## Laser Raytrace:

Relevant switches and parameters:

**switch 46** *default value = 0*

Controls treatment of multiplier for non-linear inverse bremsstrahlung absorption:

# Export Controlled Information

For switch(46) = 0, a constant multiplier (param(118)) will be used.

For switch(46) = 1, the multiplier  $\beta$  will be obtained from

$$\beta = 1 - 0.553 / (1 + 0.27/\alpha)^{0.75}$$

where

$$\alpha = \text{param}(117) <Z> (v_{\text{osc}} / v_{\text{th}})^2 / (1 - \omega_p^2/\omega^2)^{1/2}$$

For switch(46) = 2, the multiplier  $\beta$  will be obtained from

$$\beta = 1 / (1 + \alpha/3)^{1.5}$$

**switch 47**      *default value = 1*

Controls treatment of resonance absorption:

For switch(47) = 0, the fractional resonance absorption  $f$  for each ray will be obtained from the corresponding **lasray** command.

For switch(47) =  $\pm 1$ , the fractional resonance absorption  $f$  will be obtained from Ginzburg's formula

$$\begin{aligned}\phi &= 2.3 \tau \exp(-2 \tau^3 / 3) \\ \tau &= \sin \theta (n_e / \lambda |\nabla n_e|)^{1/3} \\ f &= \text{param}(117) 0.5 \phi^2\end{aligned}$$

where  $\theta$  is the angle between the ray and the surface normal.

For other values of switch(47), tabulated values of resonance absorption fraction vs. angle will be used. The tables must be specified by the user.

For switch(47) < 0, the resonance absorption will be smoothed using a 3-point stencil.

**param 115**      *default value = 1.e-3*

Ray tracing will cease when the power carried by the ray decreases below param(115) times the original value.

**param 116**      *default value = 1.0*

Multiplier on the value of  $\tau$  for Ginzburg's formula.

**param 117**      *default value = 1.0*

Multiplier on resonance absorption fraction.

**param 118**      *default value = 1.0*

Multiplier on inverse bremsstrahlung absorption if switch(46) = 0, multiplier on  $\alpha$  factor if switch(46) = 1 or 2.

# Export Controlled Information

## Timesteps and convergence:

There are three options for choosing timesteps: (1) the code can match timesteps to those contained in the (first) xfile, (2) the user can set a constant timestep (which can be changed interactively), or (3) timesteps can be chosen automatically by the code, given an initial timestep set by the user. For a steady-state problem, the timestep  $\Delta t$  is irrelevant but must still be provided, if only by accepting default values.

A single timestep can be iterated until the temperatures, iso-sequence populations and line strengths have either converged to user-set criteria or until a maximum number of iterations has been reached.

Relevant switches and parameters:

**switch 29**      *default value = 2*

Controls type of timesteps

match timesteps to xfile if switch(29) = 0  
use constant timesteps (of size  $\Delta t = \text{param}(41)$ ) if switch(29) = 1  
adjust  $\Delta t$  automatically if switch(29) = 2  
adjust  $\Delta t$  automatically with start and stop times limited by xfile times if switch(29) = 3  
increase t by a constant amount (param(41)) each cycle if switch(29) = 4

**switch 44**      *default value = 0*

Maximum number of iterations for a single timestep.

**switch 120**      *default value = 0*

Output convergence diagnostics to the terminal and .tbl file if switch(120)  $\neq 0$ . In addition, construct edits and send to output files every iteration if switch(120)  $< 0$ .

**param 25**      *default value = 1.00*

Automatic  $\Delta t$  control: maximum fractional change in density per timestep.

**param 38**      *default value = 0.00*

Minimum temperature considered when applying param(46) for  $\Delta t$  control. Changes to an electron temperature or an ion temperature less than this value will not affect the code timestep.

**param 39**      *default value = 0.00*

Minimum zbar considered when applying param(42) for  $\Delta t$  control. Zones with a zbar less than this value will not affect the code timestep. This has no effect if using kinetics subcycling.

**param 41**      *default value = 1.00e-12 (sec)*

Timestep  $\Delta t$ , if using constant timesteps, or initial timestep if using automatic timesteps.

## Export Controlled Information

**param 42**      *default value = 0.20*

Automatic  $\Delta t$  control: maximum change in zbar per timestep.

**param 43**      *default value = 1.50*

Automatic  $\Delta t$  control: maximum fractional change in  $\Delta t$  per timestep.

**param 44**      *default value = 1.00e-15 (sec)*

Automatic  $\Delta t$  control: minimum timestep.

**param 45**      *default value = 1.00 (sec)*

Automatic  $\Delta t$  control: maximum timestep.

**param 46**      *default value = 1.00*

Automatic  $\Delta t$  control: maximum fractional change in temperature per timestep.

**param 49**      *default value = 5.00*

Automatic  $\Delta t$  control: maximum value of temperature equation eigenvalue times  $\Delta t$ .

**param 55**      *default value = 1.00e-2*

Convergence control on temperatures: timesteps will be iterated until the maximum fractional change in electron and ion temperatures between two successive iterations falls below param(55).

**param 56**      *default value = 1.00e-2*

Convergence control on populations: timesteps will be iterated until the maximum fractional change in those iso-sequence populations with sufficient population between two successive iterations falls below param(56) - see param(60).

**param 57**      *default value = 1.00e-2*

Convergence control on line strengths: timesteps will be iterated until the maximum fractional change in line strengths between two successive iterations falls below param(57).

**param 60**      *default value = 1.00e-10*

Minimum fractional population in iso-sequence necessary for considering convergence. Iso-sequences with smaller fractional population than param(60) will not affect timestep iterations.

**param 107**      *default value = 1.00e-10*

Minimum fractional population in an element necessary for considering convergence. Elements with smaller fractional population than param(107) will not affect timestep iterations.

Output:

## Export Controlled Information

Edits defined in the generator will appear in the *.tbl*, *.plt*, and *.ult* files, while information requested with a **dump** command will appear in a *.dxx* file. Edits and dumps will be produced upon initialization (cycle 0), termination and at intervals controlled by switch(30) and param(40). (The exception to this is that a dump will not be produced upon initialization from a restart dump). Information for time-dependent and time-integrated edits will be saved for every timestep and sent to the appropriate files upon termination.

Restart dumps and spectral dumps can be controlled independently from the edits and dumps, via switch(65) and param(87) for restart dumps, and switch(73) and param(92) for spectral dumps. These dumps can also be produced interactively.

Spectral calculations are required to produce information for edits and (possibly) dumps, but not for restart dumps or spectral dumps. The minimum frequency of spectral calculations is also controlled by switch(30) and param(40). However, this may not update information frequently enough for time-dependent and time-integrated edits. More frequent spectral calculations can be requested via switch(72) and param(91) for this purpose.

Relevant switches and parameters:

**switch 11**      *default value = 0*

Choose file(s) for writing out plots

No files if switch(11)<0.

Default file (PDB *.ult* file if possible, ascii *.plt* file otherwise) if switch(11)=0

Both files (*.ult* and *.plt*) if switch(11)=1

Ascii *.plt* file only if switch(11)=2

**switch 12**      *default value = 0*

Controls names of *.ult* and *.plt* file(s)

Avoid overwriting existing files by incrementing suffix if switch(12)<0.

Produce a sequence of familied files if switch(12)>0. See switch(108) for controlling new files.

New names will have familied suffixes, e.g. *.u00*, *.u01*, *.u02*, etc.

**switch 26**      *default value = 0*

Tables in *.tbl* file will have 3 significant figures if switch(26)=0, 6 significant figures if switch(26)≠0.

**switch 30**      *default value = 1000*

Number of timesteps between edits / dumps.

**switch 61**      *default value = 0*

Control printing lists of radiative transitions, sorted by energy, in *.tbl* file for each atomic model

1's digit: include photoexcitations if ≠0

10's digit: include photoionizations if ≠0

100's digit: include augxs photoexcitations if ≠0

1000's digit: include augis photoexcitations if ≠0

## Export Controlled Information

**switch 65**      *default value = 1000*

Number of timesteps between restart dumps.

**switch 72**      *default value = 1000*

Number of timesteps between spectral calculations.

**switch 73**      *default value = 1000*

Number of timesteps between spectral dumps.

**switch 75**      *default value = 0*

Do not make initial timestep edits if switch(75)>0, do not make any time-independent edits if switch(75)<0.

**switch 99**      *default value = 22*

Window tiling for XGRAFIX. The 10's digit denotes the number of windows horizontally, while the 1's digit denotes the number of windows vertically. Windows beyond those which fill the tiling will overlap the tiled windows.

**switch 105**      *default value = 200*

Minimum size required to start a new dump (.dxx) file (in MB).

**switch 106**      *default value = 200*

Minimum size required to start a new spectral dump (.sxx) file (in MB).

**switch 107**      *default value = 200*

Minimum size required to start a new sensitivity dump (.dxx) file (in MB).

**switch 108**      *default value = 1000*

Minimum number of curves and mappings to start a new .uxx file if switch(12)>0.

**switch 120**      *default value = 0*

Output convergence diagnostics to terminal and .tbl file if switch(120) ≠ 0. In addition, construct edits and send to output files every iteration if switch(120) < 0.

**switch 122**      *default value = 1*

Number of timesteps between entries to xfile constructed from current run.

**switch 152**      *default value = 1001*

Maximum number of time points which can be displayed in an XGRAFIX window.

**switch 172**      *default value = 0*

## Export Controlled Information

Controls the type of binary files written. If switch(172) = 0, the default behavior will be used for each file according to the following priorities:

- 1) individual option specified for a given file (if available)
- 2) PDB (if available)
- 3) HDF5 (if available)

If switch(172) = 1, all binary files will be PDB files (if available). If switch(172) = 2, all binary files will be HDF5 files (if available).

**param 40**      *default value = 1.00e+10 (sec)*

Time interval between edits / dumps.

**param 82**      *default value = 0.00 (eV or cm<sup>-1</sup> or dimensionless)*

Instrument broadening (fwhm) to be used in each broadened spectral edit for which a value was not specified. If param(82) > 0, the broadening is constant and has units of eV for edits using frequency values and units of cm<sup>-1</sup> for edits using wavelength values. If param(82) < 0, the resolving power is constant with a value of 1./abs(param(82)).

**param 83**      *default value = 0.00*

Cosine of viewing angle with respect to z-axis for 1-d cylindrical *jsparea* edits.

**param 84**      *default value = 1.00 (cm)*

Length of cylinder to be used for calculating 1-d cylindrical *jsparea* edits.

**param 87**      *default value = 1.00e+10 (sec)*

Time interval between restart dumps.

**param 91**      *default value = 1.00e+10 (sec)*

Time interval between spectral calculations.

**param 92**      *default value = 1.00e+10 (sec)*

Time interval between spectral dumps.

**param 97**      *default value = 0.00*

Gain lists will include only transitions with gains greater than param(97).

**param 101**      *default value = 1.00e+10 (sec)*

Time interval between entries to xfile constructed from current run.

### Neutral Diffusion:

## Export Controlled Information

The neutral diffusion package couples the atomic kinetics equations with a set of diffusion equations for neutral levels, allowing an approximate treatment of neutral recycling in radiative divertors. The set of equations for the level populations  $y$  are

$$\frac{dy}{dt} + \nabla \cdot \frac{1}{m_{ion} v_x} \nabla(kTy) = \mathbf{A}y$$

where  $m_{ion}$  is the ion mass,  $v_x = 2 \times 10^{-8} n^4 n_{ion}$  is the charge-exchange frequency,  $n$  is the principal quantum number of the neutral level, and  $n_{ion}$  is the (non-neutral) ion density,  $T$  is the ion temperature, and  $\mathbf{A}$  is a matrix of rate matrices (for each node). Non-neutral levels do not experience diffusion, but are coupled to the neutral levels through the rate equations.

Boundary conditions for the neutral levels may be specified with a series of **boundary** commands with the *package* argument equal to “divertor”, i.e.

**boundary divertor type ir value [tevb]**                                    or  
**boundary divertor type k1 k2 l1 l2 value [tevb]**

Specifying a *type* of “recycling” produces a boundary flux = *-value*  $n_{ion} c_s$ , where  $c_s = [k(T_c+T_{ion})/m_{ion}]^{1/2}$ . Other possibilities for *type* include “value”, “gradient”, “flux”, “reflecting”, and “milne” (with *value*=0, which is equivalent to a free-streaming boundary condition). If *tevb* is specified, boundary values are apportioned among the ground state and excited states according to a Boltzmann distribution at temperature *tevb* (in eV). If *tevb* is not specified (or is zero), the boundary value applies to the ground state only and the excited states are treated with the same boundary condition with a zero value. The default boundary condition is zero flux (i.e. reflecting).

Relevant switches and parameters:

(note that the neutral diffusion package has its own set of switches and parameters)

**switch 60**    *default value* = 0

Couple kinetics and neutral diffusion if switch(60) = 1. Note that dswitch(1) must also be set  $\neq 0$ .

**dswitch 1**    *default value* = 0

Main control: neutral diffusion calculations will be done only if dswitch(1)  $\neq 0$ . Note that switch(60) must also be set  $\neq 0$ .

**dswitch 2**    *default value* = 0

Populations will be calculated assuming steady-state diffusion if dswitch(2) = 0, or assuming time-dependent diffusion if dswitch(2)  $\neq 0$ .

NOTE: The iterative solution currently works much better if both the diffusion and atomic kinetics are time-dependent.

**dswitch 5**    *default value* = 4

Control choice of preconditioning for gmres solver:

- no preconditioning if dswitch(5) = 0
- solve diffusion equations if dswitch(5) = 1
- solve rate equations if dswitch(5) = 2
- solve rate equations, then diffusion equations if dswitch(5) = 3
- solve diffusion equations, then rate equations if dswitch(5) = 4

## Export Controlled Information

dswitch(5) = 4 is the preferred setting. However, this may not work for all problems. dswitch(5) = 2 is the most robust setting.

**dswitch 6**      *default value = 0*

Control choice of solver for coupled diffusion/kinetics equations:

- 1-d: use tridiagonal direct solver if dswitch(6) = 0.
- use gmres if dswitch(6) ≠ 0.
- 2-d: use gmres for any value of dswitch(6).

**dswitch 7**      *default value = 100*

The maximum number of orthogonalizations used by gmres. The total number of orthogonalizations (including restarts) necessary for convergence is problem dependent. Using kinetics preconditioning (dswitch(5)=2), gmres should converge to near machine precision in a number of orthogonalizations equal to the number of spatial zones.

**dswitch 8**      *default value = -1*

The maximum number of restarts used by gmres. Set this to -1 for no restarts as 0 will result in the (default) value of 10 restarts.

**dswitch 9**      *default value = 0*

Set constraint for solving coupled diffusion/kinetics equations:

- conserve mass density at each node if dswitch(9) = 0
- conserve ion density at each node if dswitch(9) ≠ 0.

**dswitch 10**      *default value = 100*

The maximum number of iterations used by iccg (used to solve 2-d diffusion equations during preconditioning for gmres).

**dswitch 11**      *default value = 0*

Diffuse product of ion density times temperature if dswitch(11) = 0.  
Diffuse ion density if dswitch(11) ≠ 0 (i.e. move  $kT$  out of the gradient operator).

**dswitch 12**      *default value = 0*

Control printing of gmres diagnostics:

- no printing if dswitch(12) = 0
- print to .tbl file if dswitch(12) > 0
- print to screen file if dswitch(12) < 0.

**dswitch 13**      *default value = 0*

Control printing of iccg diagnostics:

- no printing if dswitch(12) = 0
- print to .tbl file if dswitch(12) > 0
- print to screen file if dswitch(12) < 0.

## Export Controlled Information

**dparam 2**      *default value = 1.0*

Multiplier for diffusion coefficients.

**dparam 3**      *default value = 0.0*

If dparam(3)  $\neq 0.0$ , use dparam(3) as the total ion density for each node in the problem.

**dparam 4**      *default value = 1.00e-2*

Convergence criterion for gmres iterations.

**dparam 5**      *default value = 1.00e-2*

Convergence criterion for iccg iterations.

# Export Controlled Information

## Summary of Switches

Switches which are not listed in the following table should not be used.

<u>Switch</u>	<u>Default</u>	
2	0	hydrodynamics control (1-d only) 0: none 1: Lagrangean hydrodynamics 2: ideal MHD 3: resistive MHD >3: none <0: evolve densities according to velocities
3	100	maximum number of hydro subcycles if >0 fixed number of hydro subcycles if <0
4	0	use FdU work term in hydro if =0, PdV work if ≠0
5	0	limit gradients in calculating Q if =0
6	0	use a zonal rho in calculating Q if =0, nodal rho if ≠0
7	0	use analytic opacities if ≠0 (see parameters 1-4) <0 : no stimulated emission term >0 : include stimulated emission term
8	1	choose angle set for radiation transfer 1-d: use gaussian angles if =0, double-gaussian angles if ≠0 2-d: use Carlson A angles if =0, Carlson B angles if =1 product ray set based on gaussian if =2, double-gaussian if ≥3 level symmetric set if =4 (for nmu≤=10) 3-d: same as 2-d
9	0	use geometric mean frequencies if =0, arithmetic mean if ≠0
10	0	when running hydro, do node-centered radiation if =0, zone-centered if ≠0
11	0	choose files to use for writing plots <0: no files 1's digit – type of file 0: default file (PDB .ult file if possible, ascii .plt file otherwise) 1: both files (.ult and .plt) 2: ascii .plt file only 10's digit – ascii file format 0: default (up to 10 columns per edit) 9>0: 1 y-variable only per edit
12	0	rename .ult and .plt files to avoid overwriting existing files if <0 family .ult and .plt files if >0
13	0	order for kinetics calculations by computational time 0: not sorted >0: decreasing order <0: increasing order
14	0	store populations if =0, fractional populations if ≠0
15	0	maximum number of kinetics subcycles if >0, fixed number if <0
16	0	coulomb logarithm 0: Lee-More 1: BPS 2: Baalrud (classical)

## Export Controlled Information

		other: fixed value (= 1.)
17	1	include line opacities and emissivities in continuum quantities for 0: no transitions >0: all transitions not identified as lines <0: all transitions
		±1: average over each continuum bin other: average over continuum bin containing transition energy
18	0	stop on non-convergence if ≠0
20	1	radiation control for kinetics 0: LTE <0: NLTE w/ fixed electron density >0: NLTE w/ fixed ion density
		±1: NLTE w/ given or calculated photon spectrum ±2: NLTE w/ black-body photons at T=Te ±3: NLTE w/ black-body photons at T=Tr
21	1	modify emission for transition energies to match level energies if ≠0
22	0	reuse stored destruction rates in kinetics if =0, start cycle with zero rates if >0, start each iteration with zero rates if >1
23	1	angular dependence of electron scattering 0: isotropic 1: dipole (1+cos <sup>2</sup> θ)
24	1	fixed total ion densities if >=0 fixed isosequence ion densities if <0
25	1	steady-state kinetics if =0 time-dependent kinetics if >0 use approx. LTE and QSS distributions to choose LTE or NLTE if >1 calculate approx. LTE and QSS distributions only if =-1 no kinetics if <-1
26	0	3-place tables if =0 6-place tables if ≠0
27	0	source node for first-flight radiation transport 0: no first-flight transport >0: steady-state transport <0: time-dependent transport
28	0	initialization control <0: LTE at fixed electron density 0: LTE at fixed ion density 1: steady-state w/ radiation transfer 2: steady-state kinetics w/o radiation transfer 3: no kinetics, broadcast boundary radiation >3: none
29	2	timestep control <0: use every N <sup>th</sup> xfile timestep, N=abs(switch(29)) 0: use xfile timesteps 1: use constant timesteps (param(41)) 2: use dynamic timesteps 3: use dynamic timesteps within first xfile limits 4: use times increasing by param(43)
30	1000	number of timesteps between snapshots
31	0	temperature calculation 0: none >0: time-dependent

# Export Controlled Information

		<0: steady-state
		1's: type of heating rate
		±1: use electronic heating rates
		±2: use energy balance (radiation + internal energy rates)
		±3: use energy balance (radiation + internal energy delta)
		4: mimic a radiation transport temperature update
		-4: same as +4 and update radiation field
		10's: type of algorithm
		0: temperature based
		>0: energy based
32	1	include line intensities in photoionizations if >0, in photoexcitations if >1
33	0	use escape factors for all photoexcitations if >0, for lines only if <0
		±1: static
		±2: Sobolev
		±3: interpolation
		±4: generalized
34	0	symmetry: 1-D
		0: non-symmetric
		<>0: symmetric about r=0.
		2-D symmetry: 2-digit integer kl
		3-D symmetry: 3-digit integer klm
		for each digit x (k,l,m)
		x = 0: non-symmetric in the x-direction
		x = 1: symmetric about x = 1
		x = 2: symmetric about x = xmax
		x = 3: symmetric about x=1 and x=xmax
35	0	calculate bremsstrahlung with a coulomb logarithm if <0, with a Gaunt factor if =0
		turn off bremsstrahlung if >0
36	0	do steady-state continuum transfer if >0
		do time-dependent continuum transfer if < 0
37	0	1-d: use Feautrier formalism if =1, integral formalism otherwise
		do steady-state line transfer if >0
		do time-dependent line transfer if < 0
38	0	assume constant continuum quantities and
		use symmetric line profiles when possible if ≠0
39	1	max. iterations for scattering in continuum transfer
40	13	max. iterations for approx. operator line transfer
41	4	iterations before acceleration in approx. operator line transfer
42	0	mesh treatment
		0: node-centering
		<0: zone-centering
		>0: staggered centering (node + zones)
44	0	max. iterations for single timestep
46	0	non-linear inverse bremsstrahlung treatment:
		0: constant multiplier – param(117)
		1: Langdon factor
		2: Kroll-Watson factor
47	1	resonant absorption fraction
		0: constant value for each ray from lasray
		±1: Ginzburg formula
		other: use tabulated values

## Export Controlled Information

		<0: smooth resonant absorption over neighboring zones control printing of linear solver diagnostics (for diffusion) 0: no printing <0: print to screen >0: print to ascii output file
48	0	include electron and ion thermal conduction if ≠0 if 2-d: 1: iccg 2: ilur 3: gmres w/ diagonal preconditioning 4: gmres w/ iccg preconditioning 5: gmres w/ ilur preconditioning other: gmres w/o preconditioning
49	0	include electron and ion thermal conduction if ≠0
50	0	include flux limiting on electron and ion thermal conduction if ≠0
51	0	electron-ion coupling / conduction coefficients by digit 1: type of conduction coefficients 0: Spitzer-Harm 1: Lee-More 2: 3-phase > 2: fixed value (= 1.) 10: do not include ions in electron-ion collision frequency if =0, include if ≠0 100: use z-dependent multiplier on conduction if =0, use 1.0 if ≠0
		<0: radiative thermal conduction, defaults for others
52	0	include Stark broadening <0: for all photoexcitations >0: for all lines
53	0	do detailed spectral calculation <-2: no spectral calculation -2: calculate opacities with black-body photons at T=Tr -1: calculate opacities and emissivities only 0: specified edit rays only 1: all rays >1: all rays including electron scattering
54	0	<0: include only special transitions (lines, Stark transitions) in spectrum 0: include continuum in spectrum, recalculated on spectral bins >0: only include photoexcitations
55	1	continuum lowering control (SP=Stewart-Pyatt, EK=Ecker-Kroll) -1: approximate accounting for missing Rydberg levels 0: no continuum lowering 1: Stewart-Pyatt with formula for degeneracy lowering ±2: Stewart-Pyatt with microfield degeneracy lowering ±3: microfield degeneracy lowering w/o continuum lowering ±>3: SP/EK w/o degeneracy lowering ±>10: use maximum of SP/EK and approximate accounting other: Stewart-Pyatt with no degeneracy lowering
56	2	number of LTE iterations w/continuum lowering
57	0	calculate natural width of photoexcitations including ≥0: both level lifetimes <0: upper level lifetime only
		0, ±1: all transitions (except ground states) ±2: no recombinations (except ground states) ±3: excitations only (except ground states) ±4: include ground state lifetimes ±5: photoexcitations only (including radiative augxs, augis)

## Export Controlled Information

58 0 update material properties (ne, zbar, etc.) after kinetics calculation  
<0: with fixed electron density  
0: with fixed ion densities  
>0: not at all

60 0 couple kinetics to particle transport if ≠0:  
±1: neutral diffusion  
±2: partially-ionized plasma transport (PIP)

61 0 send sorted radiative transition list to .tbl file if ≠0  
1's: photoexcitations  
10's: photoionizations  
100's: augxs photoexcitations  
1000's: augis photoexcitations

62 0 include electron scattering as loss in line transfer if ≠0

63 1 include doppler shifts in spectral calculation if ≠0

65 1000 number of timesteps between restart dumps

66 100 maximum number of iterations of iccg (for diffusion)

67 10 maximum number of orthogonalizations for gmres (for diffusion)

68 4 maximum number of restarts for gmres (for diffusion)

72 1000 number of timesteps between spectral calculations

73 1000 number of timesteps between spectral dumps

74 0 skip 2-d/3-d transfer calculations for timesteps with rayset errors if ≠0

75 0 do not do an initial timestep edit if >0, do not do any time-independent edits if <0

77 1 do not allow negative continuum opacities if ≠0

78 0 assume Voigt profile for escape factors if =0, Doppler profile if ≠0

79 0 include continuum opacities in escape factors if = 0

80 0 specify 1-d geometry used for escape factors if switch(79)=0  
0: problem geometry  
1: planar  
2: cylindrical  
3: spherical

specify escape factor formulation if switch(79) ≠0 according to line profile  
0: Hummer (Doppler)  
0: Apruseze (Voigt)  
-1: Apruseze (Doppler)  
-2: MacFarlane (Doppler)  
2: Bhatia & Kastner (Doppler)

81 0 use double-sided escape factors if = 0, single-sided if ≠0

82 0 use integrated density for escape factors if = 0, local density if >0, param(53) if <0

83 0 include additional energy sources in temperature calculation if ≠0

84 0 zone-center conduction coefficients if ≠0

85 0 use special scaling of initial densities and temperatures if ≠0

86 50 number of microfield values used in TOTAL

87 0 group transitions for TOTAL  
0: individually  
1: by principal quantum number  
2: by isosequence

88 0 TOTAL options by digit  
1: calculate real eigenvalues if =0, complex eigenvalues if ≠0  
10: include Doppler broadening if =0, do not include if ≠0  
100: use natural width from TOTAL if =0, add broadening from Cretin if ≠0

89 0 include line polarization shifts if ≠0

90 0 use specified data for initialization only if ≠0  
1: xfile  
10: source commands (w/o time history)

# Export Controlled Information

91        1        use updated populations for opacities, emissivities if ≠0  
92        1        calculate spectral opacities  
            -2: use continuum values  
            -1: read from dump files  
            0: in chunks  
            >0: all together  
93        0        include re-entrant radiation (in 2-d and 3-d) if ≠0  
94        0        use scratch file for 2-d and 3-d ray information if ≠0  
97        0        zone number for individual kinetics calculation if >0,  
            zone number to extract from restart dump if <0  
98        0        recalculate line profiles for each angle if ≠0  
99        23       window tiling for XGRAFIX (10's - vertical, 1's - horizontal)  
100       0        use continuum transfer routines that evolve temperature if ≠0  
101       0        include approximate Debye shielding treatment for collisional rates if ≠0  
102       0        fix level populations from xfile if ≠0  
            >0: fixed total ion densities  
            <0: allow total ion densities to vary  
103       0        include special pressure broadening of lines if ≠0  
104       0        overlap radiation transfer work spaces if ≠0  
105       200      approx. maximum size of dump files (MB)  
106       200      approx. maximum size of spectral dump files (MB)  
107       200      approx. maximum size of sensitivity dump files (MB)  
108       1000     approx. maximum number of curves/mappings in plot files  
109       0        allocate memory but do no transfer calculation if ≠0  
110       0        allocate memory but do no temperature calculation if ≠0  
111       0        skip further kinetics calculations for converged zones if ≠0  
112       0        control solver for rate matrices  
            1-10's digits – choose solver  
            0: LU decomposition for # levels ≤ switch(140)  
                    iterative for # levels > switch(140)  
            1: block tridiagonal  
            2: iterative  
            3: projective integration  
            4: banded LU decomposition  
            other: same as 0  
            100's digit – time calls to solver if >0  
            1000's digit – time calls to rate calculations if >0  
114       0        TOTAL options by digit:  
            1's: reduce (0: none, 1: some, other: all)  
            10's: onlyn (0: false, ≠0: true)  
            100's: degenr8 (0: false, ≠0: true)  
            1000's: nolowerb (0: false, ≠0: true)  
115       0        angle-dependence + components of TOTAL magnetic line profiles  
            <0: use param(100) as cosine of angle w.r.t. field  
            0: ignore magnetic fields  
            1: full angle-dependence  
            other: see full description in lineshape section  
116       0        coupling scheme used in TOTAL  
            <0: LS + align microfield along z-axis  
            0: LS  
            >0: jj  
117       0        ignore TOTAL messages if =0, route to output if ≠0  
118       0        calculate a single magnetic line profile per region if ≠0

## Export Controlled Information

119	0	line strength treatment upon setting densities from xfile <0: carry over from previous cycle 0: rescale with number densities >0: initialize to zero
120	0	output convergence diagnostics if ≠0 output edits on each iteration if <0
122	1	number of timesteps between xfile entries
123	0	check atomic data for errors 0: stop if errors are found >0: list errors in output files <0: do not list errors in output files ±1: issue warnings but continue operation other: modify data and continue operation
124	0	accelerate electron density iterations if ≠0, use relaxation if =0
125	0	average photoexcitation contributions over spectral bins if ≠0
126	5	put entire contribution into bin at line center if <0
127	44	max. principal quantum number for states w/ inner shell holes max. number of hydrogenic shells w/ inner shell holes 1s: total states (explicit + virtual through augis rates) 10s: explicit states
128	10	max. principal quantum numbers for multiply-excited states
129	1	treatment of transitions going through virtual states (augis) >0: include virtual multiply-excited and inner-shell states 0: do not include virtual multiply-excited states <0: do not include virtual multiply-excited or inner-shell states
		1: include all types of transitions (rad, aug, cex) 2: no collisional transitions (cex) 3: no autoionizing transitions (aug) 4: no radiative transitions (rad) other: no transitions (i.e. turn off this section)
130	6	generate delta n=0 transitions <-1: for ground configurations using Post, et al tables -1: for ground configurations using Post, et al tables plus corrections 0: for no configurations $n > 0$ : for principal quantum numbers up to $n$
132	0	include rates through collectivized state if ≠0
133	0	options for screened hydrogenic data types by digit $j$ : smooth energy levels starting with $j^{\text{th}}$ excited state 10: use default collisional excitation rates if =0, Mewe if =1, van Regemorter if =2, JJATOM if =3, Sampson-Golden if =4 (default = 3) 100: use default collisional ionization if =0, Lotz if =1, Mott if =2, Sampson-Golden if =3, CAM if =4, BCF if =5 (default = 3) 10 <sup>3</sup> : include neutral-neutral collisional rates if ≠0 10 <sup>4</sup> : include additional inner shell photoionizations through augis rates if ≠0 10 <sup>5</sup> : adjust ionization energies to match Scofield data if =1 adjust ionization energies to match LIMBO data if =2 do not adjust if =0 or >2 10 <sup>6</sup> : correct for non-hydrogenic oscillator strengths if =0 10 <sup>7</sup> : include non-dipole collisional excitations if ≠0
134	0	options for screened hydrogenic data types by digit 0: use screening constants from More (recent) if =0, Lee if =1, More if =2 10: fit collisional data to standard form if ≠0, include omitted PQNs if >1 100: fit photoionization data to standard form if ≠0

## Export Controlled Information

		$10^3$ : print final isosequence summaries if =1, estimate + final if =2 $10^4$ : include virtual state occupations in the ascii model if $\neq 0$ $10^5$ : include level widths in the ascii model if $\neq 0$
136	1	treatment of integrated radiation quantities for kinetics 0: treat as histograms >0: correct radiation and opacity for Planckian shape within group <0: correct spontaneous emission terms
137	0	calculate (diagonal) derivatives wrt $J_v$ if $\neq 0$
138	0	use derivatives from linear response tables if $\neq 0$ <0: use diagonal derivatives only >0: use derivatives by bit 1: correct absorption and emission 2: correct specific heat ( $dE/dT$ ) 4: correct $d(\text{absorption})/dT$ 8: correct $d(\text{emission})/dT$
139	0	send rate matrix values to output file instead of images if $\neq 0$
140	2000	maximum number of levels for direct kinetics solution (see switch(112))
141	0	control printing of linear solver diagnostics (for iterative kinetics solver) 0: no printing <0: print to screen >0: print to ascii output file
142	0	type of iterative kinetics solver 0: biconjugate gradient w/ ilur preconditioning $\neq 0$ : gmres w/ ilur preconditioning
143	100	maximum number of iterations for biconjugate gradient (for kinetics solver)
144	100	maximum number of orthogonalizations for gmres (for kinetics solver)
145	4	maximum number of restarts for gmres (for kinetics solver)
146	0	include multiphoton ionization if $\neq 0$
148	0	initialize dilution factor from astrophysical formula if $\neq 0$
149	0	account for virtual levels separately if $\neq 0$
150	0	control options for integrating rates over photon energy 0: evaluate integrand at bin center >0: integrate $\exp(-x)$ over bin <0: integrate $\exp(-x)$ over threshold bin only
		±1: apply to rates and opacities other: apply to rates only
151	1	include electron degeneracy if $\neq 0$ <0: ignore additional correction for ionizations >1: integrate collisional ionizations numerically >2: integrate collisional excitations numerically
152	1001	maximum number of points for XGRAFIX plots
154	0	directions to use for 2-d / 3-d escape factors 0: minimum over k-, l-, m-axes 1: k-axis 2: l-axis 3: m-axis 4: angles used for radiation transfer
155	0	do not continuum lower ground state degeneracies if $\neq 0$
156	0	do not iterate continuum transfer if $\neq 0$
157	0	redo radiative heating rates after continuum transfer if $\neq 0$
158	0	redo radiative heating if $>0$
		subcycle radiative heating if $< 0$
159	0	exclude transition data types from spectral opacities/emissivities if $\neq 0$ by digit

# Export Controlled Information

		1: photo-excitations 10: photo-ionizations 100: augxs 10 <sup>3</sup> : augis 10 <sup>4</sup> : bremsstrahlung 10 <sup>5</sup> : scattering 10 <sup>6</sup> : include timings
160	6	max. principal quantum number for term splitting
163	0	interpret tehot values as temperatures if =0, energies if ≠0
164	0	min. isosequence for including multiply-excited states if ≠0
165	0	max. isosequence for ionization consistency if ≠0
166	11	min. isosequence for including multiply-excited states with inner-shell holes if ≠0
167	5	max. PQN for all but highest excitation if ≠0 use same as highest excitation if =0
168	0	term splitting options 1's: photoionizations 0: do not split 1: adjust energy to lowest term >1: use individual terms 10's: collisional excitations 0: do not split >0: use individual terms 100's: collisional ionizations 0: do not split 1: adjust energy to lowest term >1: use individual terms 10 <sup>3</sup> : configuration broadening 0: do not use term splittings 1: use term splittings 10 <sup>4</sup> : oscillator strengths 0: do not use term data 1: use term data
169	1	options for mimicking radiation transport temperature update >0: include internal energy change <0: do not include internal energy change
		1's: temperature update method 0: none 1: use d/dT of source function only 2: use d/dT of net absorption 3: same as (1) using emission opacity 4: same as (2) using emission opacity 5: same as (3) with partial temperature update 6: same as (5) with reversed group ordering
170	0	other: number of iterations continuum lowering effect on collisional ionizations 0: use lowered threshold w/ scalings >0: use lowered threshold w/o scalings <0: use original threshold
171	2	options for initializing isosequence windows 0: use stored boundaries from previous cycle +/- 1 >0: use stored boundaries from previous cycle +/- switch(171) <0: use approximate steady-state approximation +/- switch(171)
172	0	type of binary files to be written

# Export Controlled Information

		0: default behavior 1 - individual option if set 2 - PDB if available 3 - HDF5 if available
		1: PDB 2: HDF5
173	0	coulomb energy and pressure treatment in kinetics 0: continuum lowering 1: ion sphere model using individual charge states other: ion sphere model using $\langle Z^2 \rangle$
174	0	accumulate emitted photons locally if $\neq 0$
175	0	calculate collisional transitions induced by hot / superthermal electrons 0: for all transitions 1: for excitations / de-excitations only 2: for ionizations only 3: for excitations / de-excitations / ionizations $\langle 0: \text{omit thermal electrons for transitions calculated with hot/superthermals}$ use Stewart-Pyatt for continuum lowering if $= 0$ , Ecker-Kroll if $\neq 0$
176	0	use net emissivity if $= 0$ , emissivity if $\neq 0$ when writing or reading LRM tables
177	0	do not use iterative improvement for rate matrix solution if $\neq 0$
178	0	$\langle 0: \text{allocate memory for second rate matrix}$ $>0: \text{do not allocate memory for second rate matrix}$ minimize kinetics memory usage if $\neq 0$
179	0	number of kinetics iterations $\langle 0: \text{fixed number of iterations}$ $>0: \text{maximum number of iterations}$
180	0	include charge exchange rates if $\neq 0$ $\langle 0: \text{iterate only charge exchange rates during kinetics iterations}$ $>0: \text{re-evaluate charge exchange rates during each kinetics iteration}$
		1's: charge exchange model 0: none 1: classical overbarrier method 2: constant cross section ( $10^{-15} Z \text{ cm}^2$ ) other: same as 1
		10's: include rates between charge states of the same element 0: yes $>0: \text{no}$
185	0	read LTE opacity server files directly if $\neq 0$
187	2	maximum number of excitations in a multiply-excited state
188	11	maximum number of K-shell excitations + 10x number from L-shells
189	0	combine photoionization transitions above threshold if $\neq 0$
190	0	balance energy for transition energies which do not match level energies 0: by modifying emission/absorption cross sections $\neq 0: \text{by modifying electron thermal energy}$
193	0	include transitions when generating hydrogenic models by digit (0=yes, 1=no) 1's: photoexcitation 10's: photoionization 100's: collisional excitation 10 <sup>3</sup> 's: collisional ionization 10 <sup>4</sup> 's: augxs 10 <sup>5</sup> 's: augis
194	0	use the LTE kinetics EOS in place of the NLTE EOS if $\neq 0$
195	0	broaden continuum edges if $> 0$

## Export Controlled Information

- 1: convolve with Gaussian for each transition
- 2: convolve with microfield for each isosequence
- 3: convolve with line shape (spectrum only)

# Export Controlled Information

## Summary of Parameters

Parameters which are not listed in the following table should not be used.

Param	Default	Description
1	3.7e+8	multiplier on analytic opacities
2	2.0	density exponent of analytic opacities
3	-3.5	temperature exponent of analytic opacities
4	1.00	multiplier on bound-free and bound-bound opacities and emissivities or frequency exponent of analytic opacities (if switch(7) ≠0)
5	1.00	multiplier for electron scattering
6	1.00	multiplier on analytic opacities for energies > param(7)
7	0.00	energy boundary for analytic opacity multipliers
11	0.00	free-free Gaunt factor if non-zero
18	0.10	fractional change in density required to turn off quiet start
19	0.10	fractional increase in temperature required to turn off quiet start
20	0.50	Courant-Friedrichs-Lax criterion multiplier
21	0.50	multiplier for linear artificial viscosity
22	0.75	multiplier for quadratic artificial viscosity
23	0.00	gamma for analytic EOS if > 0
24	1.00	multiplier for radiation pressure
25	1.00	max. fractional change in density per timestep
26	4.00	max. fractional change in electron density per iteration
27	1.00	multiplier on bremsstrahlung collision frequency
28	1.00e-02	convergence value for continuum radiation intensities
29	0.00	source radius for first-flight radiation transport (used if switch(27) ≠0)
30	0.00	implicitness parameter for temperature calculation (if >0.)
31	2.00	minimum value for Coulomb logarithm
32	0.00	minimum significant difference between LTE / NLTE isosequence distributions
33	0.00	minimum temperature for evaluating kinetics in NLTE
34	0.00	maximum temperature for tabular opacities in LTE
35	0.00	maximum temperature for tabular EOS
36	1.00	multiplier on ion thermal conduction
37	1.00	multiplier on ion thermal conduction flux limiter
38	0.00	minimum temperature to use for timestep control
39	0.00	minimum zbar to use for timestep control
40	1.00e+10	time interval between snapshots (sec)
41	1.00e-12	initial timestep (sec)
42	0.20	max. change in zbar per timestep
43	1.50	max. fractional increase in timestep
44	1.00e-15	minimum timestep (sec)
45	1.00	maximum timestep (sec)
46	1.00	max. fractional increase in temperature per timestep
47	1.00	multiplier for intensities set with switch(20)
48	1.00	multiplier for electron-ion coupling
49	5.00	temperature timestep control
50	10.0	min. # of doppler widths in CRD line wings
51	1.00e-03	convergence value for approx. operator line transfer
52	4.00	max. tau ratio for 2nd-order fit

## Export Controlled Information

53	0.00	column density or distance to boundary for escape factors (g/cm <sup>2</sup> or 1/cm <sup>2</sup> or cm)
54	1.00e-02	min. optical depth for escape factors
55	1.00e-02	convergence value for temperatures
56	1.00e-02	convergence value for iso-sequence populations
57	1.00e-02	convergence value for line strengths
58	0.00	opening angle for wedge geometry (radians)
59	0.00	r-offset for 1-d x-file positions
60	1.00e-6	threshold fractional iso-population for consideration in convergence test
61	1.00e-10	threshold fractional population for inclusion of iso-sequence (retain initial window if =0.)
62	1.00	multiplier on limit to line-width contribution of coll. excitations
63	1.00	multiplier on free-free opacity and emissivities
64	1.00	multiplier on electron thermal conduction
65	1.00	multiplier on electron thermal conduction flux limiter
66	1.00	multiplier on coes rates
67	1.00	multiplier on coll excite / colex rates
68	1.00	multiplier on coll ionize / colon rates
69	1.00	multiplier on cois rates
70	1.00	multiplier on phxs / rad_rate cross-sections
71	1.00	multiplier on phis / phot_cs cross-sections
72	1.00	multiplier on augxs cross-sections
73	1.00	multiplier on total augis rates
74	1.00	multiplier on augis rec rates
75	1.00	multiplier on augis rad rates
76	1.00	multiplier on augis aug rates
77	1.00	multiplier on augis cex rates
78	1.00e+3	max. # of Lorentz widths included in spectra
79	-2.00	exponent in degeneracy lowering formula
80	1.00e-02	min. optical depth for linearizing line transfer
81	0.50	limit on line widths as a fraction of line energy (for spectra)
82	0.00	instrument broadening fwhm for spectral edits (eV or cm <sup>-1</sup> if >0, 1/resolution if <0)
83	0.00	cosine of viewing angle w.r.t. z-axis for cylindrical <i>jsparea</i> edits
84	1.00	length for cylindrical <i>jsparea</i> edits (cm)
85	1.00	multiplier for opacity in spectral calculation
86	1.00	multiplier for emissivity in spectral calculation
87	1.00e+10	time interval between restart dumps (sec)
88	1.00e-04	convergence value for iccg or gmres calculations (for conduction)
89	1.00	multiplier for continuum lowering
90	0.00	minimum doppler width for spectral calculations (eV)
91	1.00e+10	time interval between spectral calculations (sec)
92	1.00e+10	time interval between spectral dumps (sec)
93	1.00	multiplier for line contributions to spectral opacities, emissivities
94	0.00	coefficient for energy source term
95	0.00	minimum <Z> for Saha or rate calculations
96	0.50	relaxation parameters for electron density iterations
97	0.00	min.gain for including transition in gain lists
98	6.00	max. # of Doppler widths included in spectra
99	0.00	max. ratio between temperature iterates for steady-state calculation if > 1.0
100	0.00	angle w.r.t. magnetic field for TOTAL line profiles
101	1.00e+10	time interval between entries in xfile (sec)
102	0.00	minimum electron temperature (eV)
103	0.00	minimum ion temperature (eV)
104	1.00	minimum temperature for collisional rate fits (eV)
105	1.00e+4	maximum temperature for collisional rate fits (eV)

## Export Controlled Information

106	0.10	fractional perturbation for calculating linear response
107	1.00e-10	threshold fractional elemental population for consideration in convergence test
108	1.00	maximum fractional intensity change for use with linear response
109	1.00e-12	convergence criterion for iterative kinetics solver
110	4.00	multiplier for TOTAL energy ranges used for spectrum
111	10.0	maximum allowed fractional change in opacity when using linear response tables
112	0.50	multiplier for generating configuration broadening in screened hydrogenic models
113	1.00	multiplier for continuum edge broadening from collisions and Stark
114	3.50	telescoping factor for projective integration
115	1.00e-3	minimum fractional intensity for laser ray tracing
116	1.00	multiplier on Ginzburg resonance absorption $\tau$
117	1.00	multiplier on resonance absorption fraction
118	1.00	multiplier on inverse brems absorption, or on quiver velocity (if switch(46)=1,2)
119	0.00	minimum (-) rate matrix diagonal for steady-state kinetics
120	1.00	multiplier on electrical conductivity
121	1.00	multiplier for doppler widths for spectral calculation
122	1.00	multiplier for natural widths for spectral calculation
123	1.00	multiplier for Stark widths for spectral calculation
124	0.00	minimum population for doing kinetics or for sourcing densities
125	1.00	multiplier for UTA and configuration broadening
126	1.00	multiplier for level component of transition configuration broadening
127	0.30	multiplier for energy component of transition configuration broadening
128	1.00	multiplier for Z-component of transition configuration broadening
129	1.00	multiplier for NLTE dE/dt term
130	1.00	maximum virtual population (relative to isosequence population)
131	0.00	maximum transition rate allowed without warning if $\neq 0$ . >0. : produces error <0. : produces warning
132	0.00	multiplier on electron-neutral scattering cross section
133	1.00	multiplier on boundary values for spectral calculations
134	1.00	multiplier for non-thermal collisional excitation / de-excitation
135	1.00	multiplier for non-thermal collisional ionization
136	0.00	multiplier for non-thermal collisional recombination
137	1.00	multiplier for non-thermal recombination (other than collisional)
138	0.00	multiplier on charge exchange rates
139	0.00	total column density or distance for 0D escape factor edits ( $\text{g/cm}^2$ or $1/\text{cm}^2$ or cm)
140	1.00	multiplier for electron-electron coupling
141	1.00	multiplier for ion-ion coupling
142	1.00	multiplier on corrections obtained from LRM tables
146	1.00	multiplier on level widths used in constructing autoionization rates
147	0.00	multiplier on level widths used in rate calculations
148	0.50	multiplier for FFM ion dynamics in TOTAL
149	1.00e-3	threshold for integrating line profiles for photoexcitation rates

# Export Controlled Information

## Atomic Physics Input

Atomic data must be included in an ascii datafile for each atomic model specified in the generator. The data defines processes in one direction only - from a lower state to an upper state. The inverse processes will be calculated to be consistent with detailed balance. Transitions which do not correspond to defined levels are ignored (with a warning). Blank lines can be freely interspersed within the datafile. Lines beginning with *c* are interpreted as comments. Arguments within brackets are optional. Additional fields on any command line are permitted.

### File organization:

**idatname** *name*

This command is ignored by CRETIN.

**data** *type* [**sublevel**]

The **data** command specifies the beginning of a section of levels or fitted atomic data. The type may be *model*, *coes*, *coll excite*, *coll ionize*, *phxs*, *phis*, *augxs*, *augis* or *cois*. There may be multiple data sections of each type except for *model*. The type *phxsI* is also acceptable.

The *model* section, which defines the level structure, must come before any of the sections describing transitions.

If **sublevel** is present after a section type of transition data, the transition is defined between two sublevels and the **d** command must include the sublevel indices.

There are alternate forms for some of the data sections which use the parameterizations output by the HULLAC and FAC codes. The following section types may be substituted.

<u>type</u>	<u>alternate</u>
<i>coll excite</i>	<i>colex</i> or <i>cxbbtc</i>
<i>coll ionize</i>	<i>colon</i>
<i>phxs</i>	<i>rad_rate</i>
<i>phis</i>	<i>phot_cs</i> or <i>phot_rr</i>
<i>augxs</i>	<i>aut_rate</i>

**end**

This command signifies the end of a section of levels or fitted data.

### Model specification:

*model*

**enot** *iso nameiso energy*

This command defines an isoelectronic sequence *iso* with name *nameiso* and ionization potential *energy* (eV). The isoelectronic sequences include 0 for a bare nucleus, 1 for a hydrogenic ion, 2 for a helium-like ion, etc.

# Export Controlled Information

**elev** *iso i namelev g energy [ ... n]*

This command defines a level within an isoelectronic sequence, where *iso* identifies the isoelectronic sequence, *i* is the level number within the sequence, *namelev* is an identifier for the level (which is usually ignored by CRETIN but must be present), *g* is the statistical weight, and *energy* is the energy (eV) of the level relative to the ground state of the sequence. The principal quantum number *n* must be the last field on the line.

If the identifier *namelev* includes the sequence “*nX*”, where *n* is the principal quantum number, *X* will be interpreted to yield the angular momentum quantum number *l*, e.g. *X*=“s” or “S” for *l*=1, *X*=“p” or “P” for *l*=2. The first occurrence of *n* in *namelev* must be in the sequence “*nX*”.

**elevw** *iso i namelev g energy ewidth [ ... n]*

This command is the same as the **elev** command with the addition of a width *ewidth* (eV), representing configuration broadening of the level.

**esublev** *iso i isub namelev g de [ewidth]*

This command defines a sublevel within an isoelectronic sequence, where *iso* identifies the isoelectronic sequence, *i* is the level number within the sequence, *isub* is the sublevel number within the level, *namelev* is an identifier for the sublevel (which is usually ignored by CRETIN but must be present), *g* is the statistical weight, and *de* is the energy (eV) of the level relative to the energy of the base level defined by (*iso,i*). The remaining fields are the same as for the **elev** or **elevw** commands.

Sublevels are ignored during the kinetics calculations but are used when calculating spectra or lineshapes using TOTAL. In these instances, the sublevels are populated statistically from the base level. Transitions between two sublevels are used rather than transitions between the corresponding base levels.

All **esublev** commands must occur after all **elev** commands and before the end of the **data model** section. It is not necessary to preface them with an **enot** command.

**c shell** *k-index l-index m-index ...*

**c n\_shells**

**c ls\_shells**

**c jj\_shells**

These commands tell CRETIN to interpret various fields on the **elev** command as shell occupations. The **shell** command identifies the fields from 1 through *k-index* as containing the occupations of the k-shell, the *k-index+1* through *l-index* fields as containing the occupation of the l-shell, etc. Multiple indices can contribute to the total occupation of shell *n*. The **n\_shells** command identifies one field as belonging to each shell *n*, starting with *n*=1, i.e. each index corresponds to a principal quantum number. The **ls\_shells** command identifies *n+1* fields as belonging to shell *n*, as if each index corresponds to an orbital angular quantum number. The **jj\_shells** command identifies *2n+1* fields as belonging to shell *n*, as if each index corresponds to an total angular quantum number. This interpretation remains in effect for all succeeding **elev** commands until another of these commands is encountered. Note that these commands must be preceded by a **c**, so that other codes will interpret this as a comment.

# Export Controlled Information

Electron collisional excitation/ionization:  
*coes, coll excite, coll ionize*

**d iso1 i1 iso2 i2 a0 a1 a2 a3 n de tevmin tevmax**

This command defines an electron collisional excitation or ionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*). *a0, a1, a2, a3, n*, and *de* are fitting parameters as defined below. *tevmin* and *tevmax* are the minimum and maximum electron temperatures (eV) over which the rate coefficients were fitted.

The rate coefficient sigma (cm<sup>3</sup>/sec) for a transition of energy  $\Delta e$  and an electron temperature *tev* is defined by

$$\text{sigma} = 1.58 \times 10^{-5} n (p(b)/b) e^{-b} / \text{tev}^{1.5}$$

where

$$\begin{aligned} b &= de / \text{tev} & \text{tevmin} < \text{tev} < \text{tevmax} \\ &de / \text{tevmin} & \text{tevmin} > \text{tev} \\ &de / \text{tevmax} & \text{tev} > \text{tevmax} \end{aligned}$$

and

$$\ln p(b) = a0 + a1 (\ln b) + a2 (\ln b)^2 + a3 (\ln b)^3.$$

For ionizations, *b* is evaluated using  $\Delta e$  rather than *de*.

If requested, the rate coefficient is then modified for Debye shielding in the following manner:

$$1/\text{sigma (shielded)} = 1/\text{sigma} + 1/(v_e D^2)$$

where *v<sub>e</sub>* is the electron thermal velocity and *D* is the Debye length.

*sampson excite, sampson ionize*

**d iso1 i1 iso2 i2 c0 c1 c2 c3 s0 [de]**

This command defines an electron collisional excitation or ionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*). *c0, c1, c2, c3* and *s0* are fitting parameters as defined below.

The collision strength  $\Omega$  for a transition of energy  $\Delta e$  and an incident energy  $\varepsilon$  is defined by

$$\Omega = s0 [c0 + c1 / \sqrt{x} + c2 / x + c3 (\ln x)], \quad x = \varepsilon / \Delta e$$

and the rate coefficient sigma (cm<sup>3</sup>/sec) for an electron temperature *tev* is defined by

$$\text{sigma} = 1.09 \times 10^{-6} q(b) e^{-b} / (\Delta e \text{tev}^{0.5})$$

where *b* =  $\Delta e / \text{tev}$  and

$$q(b) = s0 [c0 + c1 \sqrt{\pi b} \operatorname{Erfc}(b) e^{b^2} + (c2 b + c3) E_1(b) e^b]$$

*E<sub>1</sub>* is the first exponential integral and *Erfc* is the complementary error function.

If the optional argument *de* is present, it is used in place of the transition energy  $\Delta e$ .

# Export Controlled Information

colex, colon

**d iso1 i1 iso2 i2 c0 c1 c2 c3 a0 c4**

This command defines an electron collisional excitation or ionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*). *c0, c1, c2, c3, c4* and *a0* are fitting parameters as defined below.

The collision strength  $\Omega$  for a transition of energy  $\Delta e$  and an incident energy  $\varepsilon$  is defined by

$$\Omega = c0 (\ln x) + c1 + c2 / (x+a0) + c3 / (x+a0)^2 + c4 / (x+a0)^3, x = \varepsilon/\Delta e$$

and the rate coefficient sigma (cm<sup>3</sup>/sec) for an electron temperature *tev* is defined by

$$\text{sigma} = 8.01 \times 10^{-8} q(b) e^{-b} / tev^{0.5}$$

where  $b = \Delta e / tev$  and

$$q(b) = c0 E_1(b) e^b + c1 + c2 b / (a0+1) + c4 [b / (a0+1)^2 - b^2 / (a0+1)]/2 + (c2 - c3 b + c4 b^2 / 2) b E_1((a0+1)b) e^{(a0+1)b}$$

and  $E_1$  is the first exponential integral.

cxbbtc

**d iso1 i1 iso2 i2 n c0 c1 z0 z2 ... z8**

This command defines an electron collisional excitation process from lower level (*iso1,i1*) to upper level (*iso2,i2*). *n, c0, c1*, and *zi* are fitting parameters for the Busquet-Burgess-Tully-Chidichimo (BBTC) parameterization (using 9 fitting values and linear interpolation) as described in M. Busquet, HEDP **3**, 2007, 48-51.

The collision strength  $\Omega$  for a transition of energy  $\Delta e$  and an incident energy  $\varepsilon$  is defined by

$$\Omega = c0 (\ln u) + z / (u+c1)^n, u = \varepsilon/\Delta e$$

where

$$x = 1 - (1+c1) / (u+c1), z1 = z(x1), z2 = z(x2) \dots$$

with

$$\{x_i\} = \{0.0, 0.05, 0.15, 0.3, 0.5, 0.7, 0.85, 0.95, 1.0\}$$

and the rate coefficient sigma (cm<sup>3</sup>/sec) for an electron temperature *tev* is defined by

$$\text{sigma} = 8.01 \times 10^{-8} q(b) e^{-b} / tev^{0.5}$$

where  $b = \Delta e / tev$  and  $q(b)$  is obtained by integrating  $\Omega(u) \exp(-bu)$ .

cam

**d iso1 i1 iso2 i2 c0 c1 c2**

This command defines an electron collisional ionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*). *c0, c1*, and *c2* are fitting parameters as defined below. As described in R.E.H. Clark, J. Abdallah, Jr., and J.B. Mann, ApJ **381**, 1991, 597-600, these parameters can be

## Export Controlled Information

used to describe ionization from either a state specified by  $(n,l)$  or an average state specified by  $n$ .

The collision strength  $\Omega$  for a transition of energy  $\Delta e$  and an incident energy  $\varepsilon$  is defined by

$$\Omega = c0 (\ln x) + c1 (1 - 1/x) + c3 (1 - 1/x)^2, \quad x = \varepsilon/\Delta e$$

and the rate coefficient sigma ( $\text{cm}^3/\text{sec}$ ) for an electron temperature  $tev$  is defined by

$$\text{sigma} = 8.01 \times 10^{-8} q(b) e^{-b} / tev^{0.5}$$

where  $b = \Delta e / tev$  and

$$q(b) = c0 E_1(b) e^b + c1 [1 - b E_1(b) e^b] + c2 [1 - 2b E_1(b) e^b + 2b E_2(b) e^b]$$

and  $E_1$  ( $E_2$ ) is the first (second) exponential integral.

### Photoexcitation:

*phxs*

**d iso1 i1 iso2 i2 f [lambda] [width] [e-dipole] [m-dipole]**

This command defines a photoexcitation from lower level  $(iso1,i1)$  to upper level  $(iso2,i2)$ .  $f$  is the absorption oscillator strength (dimensionless). If  $lambda$  is present and non-zero, it is taken to be the wavelength of the transition (in Angstroms), otherwise the energy of the transition is calculated from the energy levels of the model. If  $width$  is present, it is taken to be an additional width (in eV) for the transition, e.g. representing a UTA, which will be added to the Doppler width. For use with TOTAL,  $e\text{-dipole}$  is the electric dipole matrix element and  $m\text{-dipole}$  is the magnetic dipole matrix element.

*rad\_rate*

**d iso1 i1 iso2 i2 gA [lambda] [width] [e-dipole] [m-dipole]**

This command defines a photoexcitation from lower level  $(iso1,i1)$  to upper level  $(iso2,i2)$ .  $gA$  is the spontaneous emission rate (1/sec) (i.e. the statistical weight times the Einstein A-coefficient). The optional arguments  $lambda$ ,  $width$ ,  $e\text{-dipole}$ , and  $m\text{-dipole}$  are the same as for *phxs*.

### Photoionization:

*phis*

**d iso1 i1 iso2 i2 a0 a1 a2 a3 n de emin emax**

This command defines a photoionization process from lower level  $(iso1,i1)$  to upper level  $(iso2,i2)$ .  $a0$ ,  $a1$ ,  $a2$ ,  $a3$ ,  $n$ , and  $de$  are fitting parameters as defined below.  $emin$  and  $emax$  are the minimum and maximum photon energies (eV) over which the fit is valid. The cross section is extrapolated above  $emax$  assuming a dependence of  $1/e^3$ .

The cross section sigma ( $\text{cm}^2$ ) for a photon energy  $e$  (eV) is defined by

$$\text{sigma} = 10^{-18} n p(b) (13.606/de)$$

where

# Export Controlled Information

$b = e / de$   
and  
 $\ln p(b) = a0 + a1 (\ln b) + a2 (\ln b)^2 + a3 (\ln b)^3.$

*phot\_cs*  
*phot\_ion*

**d iso1 i1 iso2 i2 a0 b0 c0 d0**

This command defines a photoionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*).

The cross section sigma (cm<sup>2</sup>) for a photon energy *e* (eV) is defined by

$$\text{sigma} = a0 (e + b0)^c0 [(e - \Delta e)/(e + b0)]^d0$$

where  $\Delta e$  is the threshold energy.

The *phot\_ion* data parameterization is the same as the *phot\_cs* parameterization, but a Gaunt factor is added when evaluating the cross sections.

*phot\_rr*

**d iso1 i1 iso2 i2 a0 b0 c1 c2 de**

This command defines a photoionization process from lower level (*iso1,i1*) to upper level (*iso2,i2*).

The cross section sigma (cm<sup>2</sup>) for a photon energy *e* (eV) is defined by

$$\text{sigma} = 10^{-18} p(b) (13.606/de)$$

where

$b = e / de$   
and  
 $p(b) = a0 [ c1 / (b + b0) + c2 / (b + b0)^2 ] / b^2.$

Auger Excitation:  
*augxs*

**d iso1 i1 iso2 i2 f lambda [a] [g]**

This command defines a photoexcitation from lower level (*iso1,i1*) to a doubly excited state with a transition energy corresponding to the wavelength *lambda* (Angstroms). The doubly excited state decays through an Auger process to level (*iso2,i2*). *f* is the absorption oscillator strength (dimensionless). If *a* is present, it is the Auger decay rate, otherwise the decay rate is assumed to be 10<sup>18</sup>/sec. If *a* is negative, an infinite decay rate is assumed. If *g* is present, it is the statistical weight of the doubly excited state, otherwise the statistical weight is 1.

**d iso1 i1 iso2 i2 a**

This command defines an Auger decay from upper level (*iso2,i2*) to lower level (*iso1,i1*) with decay rate *a* (1/sec).

# Export Controlled Information

*aut\_rate*

**d** *iso1 i1 iso2 i2 a*

This command defines an Auger decay from upper level (*iso2,i2*) to lower level (*iso1,i1*) with decay rate *a* (1/sec).

Auger Ionization:

*augis*

**d** *iso elev g*

This command begins the specification of a set of Auger processes by defining a doubly excited state in isoelectronic sequence *iso* with an energy *elev* above the ground state of that sequence and statistical weight *g*. The doubly excited state can participate in a number of processes connecting levels in isoelectronic sequences *iso-1*, *iso*, and *iso+1* as defined by any number of the following commands.

**d** *iso1 i1 a0 a1 a2 a3 n de emin emax [g]*

This is an alternative command which also begins the specification of a set of Auger processes where (*iso1,i1*) is the starting level which has an inner shell electron that can be photoionized to yield a doubly excited state an energy *emin* above the starting level and statistical weight *g* (or 1 if *g* is absent). The photoionization cross section is defined as in the *phis* section. The previous form of the **d** command is preferred for new models.

**rec** *iso1 i1 a0 a1 a2 a3 n de emin emax*  
**rec** *iso1 i1 iso2 i2 a0 b0 c1 c2 de*

This command defines a radiative ionization from (*iso1,i1*) to the doubly excited state with a cross section defined as in either the *phis* or *phot\_rr* sections, for the first and second forms, respectively. This cannot be used with the second form of the **d** command.

**rad** *iso2 i2 def [width]*

This command defines a radiative decay from the doubly excited state to (*iso2,i2*) with transition energy *de*, absorption oscillator strength *f* and optional transition width *width*.

**aug** *iso2 i2 rate*

This command defines an Auger decay from the doubly excited state to (*iso2,i2*) with decay rate *rate* (1/sec).

**cex | cex0** *iso1 i1 a0 a1 a2 a3 n de tevmin tevmax*  
**cex | cex1** *iso1 i1 iso2 i2 c0 c1 c2 c3 a0 c4*  
**cex | cex3** *iso1 i1 iso2 i2 c0 c1 c2 c3 s0 [de]*

This command defines a collisional excitation or ionization from (*iso1,i1*) to the doubly excited state with a rate defined as in the *coes*, *sampson excite*, *sampson ionize*, *colex* or *colon* section. If **cex** is specified, the command is interpreted as the same parameterization as that used in the collisional excitation / ionization sections. If **cex0**, **cex1**, or **cex3** is specified, the command is interpreted as the parameterization from the *coes*, *colex* / *colon*, or *sampson excite* / *ionize*

# Export Controlled Information

section, respectively. These different parameterizations may be mixed within this section or within a single set of processes.

**done**

This command finishes the specification of the set of processes begun by a **d** command.

Ion collisional excitation/ionization:  
*cois, icoll excite, icoll ionize*

**per z iso**

This command specifies that all processes until the next **per** command refer to interactions with ions of atomic number *z* and isoelectronic sequence *iso*. This command is the same for all ion collisional section.

**d isol i1 iso2 i2 a0 a1 a2 a3 n de tivmin tivmax**

This command defines an ion collisional excitation or ionization process from lower level (*isol,i1*) to upper level (*iso2,i2*). *a0, a1, a2, a3, n*, and *de* are fitting parameters as defined below. *tivmin* and *tivmax* are the minimum and maximum ion temperatures (eV) over which the rate coefficients were fitted.

The rate coefficient sigma (cm<sup>3</sup>/sec) for a transition of energy  $\Delta e$  and an ion temperature *tiv* is defined by

$$\text{sigma} = 1.58 \times 10^{-5} n (p(b)/b) e^{-b} / tiv^{1.5}$$

where

$$\begin{array}{ll} b = \Delta e / tiv & tivmin < tiv < tivmax \\ de / tivmin & tivmin > tiv \\ de / tivmax & tiv > tivmax \end{array}$$

and

$$\ln p(b) = a0 + a1 (\ln b) + a2 (\ln b)^2 + a3 (\ln b)^3.$$

*icolex, icolon*

**d isol i1 iso2 i2 c0 c1 c2 c3 a0 c4**

This command defines an ion collisional excitation or ionization process from lower level (*isol,i1*) to upper level (*iso2,i2*). *c0, c1, c2, c3, c4* and *a0* are fitting parameters as defined below.

The collision strength  $\Omega$  for a transition of energy  $\Delta e$  and an incident energy  $\varepsilon$  is defined by

$$\Omega = c0 (\ln x) + c1 + c2 / (x+a0) + c3 / (x+a0)^2 + c4 / (x+a0)^2, x = \varepsilon / \Delta e$$

and the rate coefficient sigma (cm<sup>3</sup>/sec) for an ion temperature *tiv* is defined by

$$\text{sigma} = 8.01 \times 10^{-8} q(b) e^{-b} / tiv^{0.5}$$

where

$$q(b) = c0 E_1(b) e^b + c1 + c3 b / (a0+1) + c4 [b / (a0+1)^2 - b^2 / (a0+1)]/2$$

## Export Controlled Information

$$+ (c2 - c3 b + c4 b^2 / 2) b E_1((a\theta+1)b) e^{(a\theta+1)b}$$

and  $E_1$  is the first exponential integral.

### Sublevel transitions:

type **sublevel**

**d** *iso1 i1 isub1 iso2 i2 isub2 ...*

This form of the **d** command defines a transition from lower sublevel (*iso1,i1,isub1*) to upper sublevel (*iso2,i2,isub2*). The remaining parameters are as described for transitions appropriate to data of type *type*.

# Export Controlled Information

## Xfile Input

Various quantities can be specified in supplementary data files (xfiles), as defined below. There may be any number of xfiles, but xfiles beyond the first one containing **pbins** or **pbcgs** sections are assumed to have the same group structure as the first xfile. Blank lines can be freely interspersed within the xfile. Lines beginning with *c* are interpreted as comments. All input on a line following an exclamation point (!) will be interpreted as a trailing comment. If only one problem time is present in the file, the data will be used for initialization at generation time only. Commands ending in “**3d**” are identical to those ending in “**2d**” – either form may be used for either dimensionality.

**problem** *name time itime ir1 ir2*

Specify the time *time* (sec) for which the following data applies and the number of nodes ( $n = ir2 - ir1 + 1$ ) for which data will be supplied. The remaining fields are necessary but are ignored by CRETIN.

**e** *ng*

This command is followed by *ng* energies (eV) to define the group structure for continuum radiation. The energies appearing here are the group upper boundaries. The lower boundary of the first group is zero. This command should appear only in the first section of the xfile. The group structure need not be the same as the group structure specified in the generator, and each xfile may have a separate group structure. If no group structure is specified in the generator, the group structure from the first xfile will be used.

**pbins** *ir*

This command is followed by *ng* photon energy densities (jerks/keV/cm<sup>3</sup>) for node *ir*. If the energy densities are not specified for each node in the problem, the information from the first node will be broadcast to all other nodes or used as boundary information where appropriate. This command cannot coexist with the **pbcgs** command.

**pbcgs** *ir*

This command is followed by *ng* photon energy densities (erg/Hz/cm<sup>3</sup>) for node *ir*. If the energy densities are not specified for each node in the problem, the information from the first node will be broadcast to all other nodes or used as boundary information where appropriate. This command cannot coexist with the **pbins** command.

**ee** *ne*

This command is followed by *ne+1* energies (eV) to define the group structure for the non-thermal electron distribution. The energies appearing here are the group boundaries, starting with the lower boundary of the first group. This command should appear only in the first section of the xfile.

**edist** *ir*

This command is followed by *ne* electron densities (#/eV/cm<sup>3</sup>) for node *ir*.

**r**

This command is followed by *n* 1-d positions (cm).

# Export Controlled Information

## **x2d or x3d**

This command is followed by  $n$  2-d (x- or r-) or 3-d (x) positions (cm).

## **y2d or y3d**

This command is followed by  $n$  2-d (y- or z-) or 3-d (y) positions (cm).

## **z2d or z3d**

This command is followed by  $n$  2-d (y- or z-) or 3-d (z) positions (cm).

## **u**

This command is followed by  $n$  1-d velocities (cm/sec).

## **u2d or u3d**

This command is followed by  $n$  2-d (x- or r-) or 3-d (x) velocities (cm/sec).

## **v2d or v3d**

This command is followed by  $n$  2-d (y- or z-) or 3-d (y) velocities (cm/sec).

## **w2d or w3d**

This command is followed by  $n$  2-d (z- or f-) or 3-d (z) velocities (cm/sec).

## **durd<sub>r</sub>**

This command is followed by  $n$  1-d velocity gradients (1/sec).

## **d**

This command is followed by  $n$  densities (g/cm<sup>3</sup>).

## **te**

This command is followed by  $n$  electron temperatures (eV).

## **ti**

This command is followed by  $n$  ion temperatures (eV).

## **tr**

This command is followed by  $n$  radiation temperatures (eV).

## **tdilute**

This command is followed by  $n$  dilution factors, used for constructing intensities from radiation temperatures. If this section is absent, the dilution factors are assumed to be 1.0.

# Export Controlled Information

## **bx**

This command is followed by  $n$  magnetic field x- (1-d planar or 2-d xy or 3-d) or r- (1-d cylindrical, spherical or 2-d rz) components (G).

## **by**

This command is followed by  $n$  magnetic field field y- (1-d planar or 2-d xy or 3-d) or theta (1-d spherical) or z- (1-d cylindrical or 2-d rz) components (G).

## **bz**

This command is followed by  $n$  magnetic field field z-(1-d planar or 2-d xy or 3-d) or phi- (1-d cylindrical,spherical or 2-d rz) components (G).

## **bfield**

This command is followed by  $n$  magnetic field magnitudes (G).

## **done**

This command terminates the data associated with a given time or **problem** command.

The following commands contain time-independent data and need only appear once in the xfile. They do not need to be contained within a **problem** section.

## **regmap $kmax$ [ $lmax$ ] [ $mmax$ ]**

This command is followed by  $n$  integers identifying each node with a region. These values must be consistent with  $n = kmax$  or  $n = kmax \times lmax$  or  $n = kmax \times lmax \times mmax$ . If the integer for a node =0, that denotes a node in a null region. If the integer for a node is <0, that denotes material from multiple regions.

## **logmesh $kmax$ $lmax$**

This command is equivalent to the **regmap** command.

## **coeff $ic$**

This command is followed by  $n$  values. This set of values is associated with index  $ic$ .

# Export Controlled Information

The following 6 commands allow the user to define time-dependent element and electron number densities for each node in the problem. The **znuc** and (optionally) **anuc**, **isonuc** and **i** commands need only be present once in the xfile to identify the elements for which number densities will be specified. If the identifiers specified in the **znuc** (**anuc**, **isonuc**, **i**) sections match the specification in a **source** command, the number densities from the xfile will be used for the element or isosequence specified in the **source** command. The identifiers are used only as labels and do not have to match the specification in an **atoms** command. Elements are not restricted to the regions where they were specified in the generator. If an element appears at a node where it was previously absent, the populations are initialized to steady-state conditions appropriate to that node. If the **znuc** command is present, either the **ne**, **ni** and **zbar** commands must also be present, or the **isonuc** and **ni** commands must also be present.

## **znuc** *nnuc*

This command is followed by *nnuc* atomic numbers.

## **anuc** *nnuc*

This command is followed by *nnuc* atomic weights.

## **isonuc** *nnuc*

This command is followed by *nnuc* isosequence specifications.

## **inuc** *nnuc*

This command is followed by *nnuc* level specifications.

## **ne**

This command is followed by *n* electron densities (1/cm<sup>3</sup>).

## **ni** *inuc* [*ireg*]

This command is followed by *n* number densities for the element (or isosequence) *inuc* defined with the **znuc** (or **znuc** and **isonuc**) [and **anuc**] command[s] (1/cm<sup>3</sup>). If *ireg* is present and >0, it defines a region label to be (see **regmap**) and can be used as an identifier in a **source** command.

## **zbar**

This command is followed by *n* average ion charges.

The following 2 commands allow the user to define time-dependent electron number distributions whose temperatures can be different from that of the main electron distribution.

## **tehot** *ihot*

This command is followed by *n* temperatures for electron distribution *ihot* (eV).

## **nehot** *ihot*

This command is followed by *n* number densities for electron distribution *ihot* (eV).

# Export Controlled Information

## Edits

All edits in CRETIN are specified by an edit name and four indices, plus an optional multiplier, e.g.,  
*edit i1 i2 i3 i4 [mult]*.

The first index, *i1*, specifies either the model (*iz*), line (*iline*), transition (*itr*), spectral-group (*is*), or xfile (*ix*). The second index, *i2*, specifies the spatial node (*ir*), k-line (*k*) or l-line (*l*). The third index, *i3*, specifies either the continuum group (*igp*), spectral energy group (*isp*), line energy bin (*jfr*), electron distribution energy group (*ie*) or isoelectronic sequence (*iso*). The fourth index, *i4*, specifies either an atomic level (*i*) or directional ray (*iray*).

Trailing indices which are zero may be omitted. If *mult* is present, it is used as a multiplier on the edit value and can be used to change units. In this case, trailing indices may not be omitted.

If an edit runs over a particular index (e.g., editing a quantity for all nodes), that index should be zero. The exception is if an edit runs over a k-line or (l-line), that index should specify the k-line (or l-line). There are also three special cases for indices: *ir* = -1 specifies the last node, *jfr* = -1 specifies the bin at line center and *isp* = -*iline* specifies the spectral energy group corresponding to the center of line *iline*.

Units for the edits are mostly c.g.s. with a few exceptions. Some energies and temperatures are in eV, line intensities are in photons/mode (as well as c.g.s.). Continuum and spectral radiation quantities are in c.g.s., but these units can be modified separately for each edit with a **rad-units** command, as can line radiation quantities which do not use photons/mode.

All edits can be integrated or averaged over time with a **time-integrated** or **time-averaged** command. Most edits can also be integrated over space over volume- or mass-averaged with an **integrated**, **averaged**, or **mass-averaged** command.

CRETIN does NOT check edits for correctness or consistency. Undefined edits will return zero values. Non-zero values for unused indices will not affect the results -- this does not include the (usually-) zero value for the running index.

Independent variables for *xvar*:

1. cycle, irow, time
2. ir, r, cdens, x2d, y2d, z2d, x3d, y3d, z3d, xy, xzc, yzc, zzc, k, kr, kx, ky, kz, l, lr, lx, ly, lz, m, mr, mx, my, mz
3. ifr, energy, freq, wvl, ebins, fbins, wbins, ifrline, evline, isp, sp\_energy, sp\_freq, sp\_wvl, ie, iso
4. level, elev

### Time and timesteps:

#### **cycle**

Timestep number.

#### **iter**

Timestep number.

#### **itime**

Timestep number.

# Export Controlled Information

## **time**

Time (sec).

## **dtme**

Timestep (sec).

## **dtfrac**

Fractional timestep.

## **dtme\_te 0 ir**

Timestep due to electron temperature change at node *ir*.

## **dtme\_ti 0 ir**

Timestep due to ion temperature change at node *ir*.

## **dtme\_tr 0 ir**

Timestep due to radiation temperature / energy density change at node *ir*.

## **dtme\_rho 0 ir**

Timestep due to density change at node *ir*.

## **dtmek iz ir**

Timestep due to kinetics for element *iz* at node *ir*.

## **dtkin iz ir**

Current kinetics timestep during subcycling for element *iz* at node *ir*.

## **dtkmin iz ir**

Minimum kinetics timestep during subcycling for element *iz* at node *ir*.

## **dtkmax iz ir**

Maximum kinetics timestep during subcycling for element *iz* at node *ir*.

## **id\_dtime**

Type of timestep constraint responsible for current timestep, according to

- 3 : reduced timestep for retrying failed iteration
- 2 : timestep for spectral postprocessing
- 1 : xfile timestep
- 0 : constant timestep
- 1 : ionization state
- 2 : electron temperature

# Export Controlled Information

3 : ion temperature

4 : radiation temperature / energy density

## **ir\_dtime**

Node index with most restrictive timestep constraint.

## **ir\_dtimek**

Node index with most restrictive atomic kinetics timestep constraint.

## **iz\_dtime**

## **iz\_dtimek**

Atomic model index with most restrictive timestep constraint.

## Mesh properties:

### **ir 0 ir**

Node index.

### **-ir 0 ir**

Node index relative to the last node in the mesh.

### **ireg 0 ir**

Region index for material at node *ir*. A region index of 0 indicates no material is present. A negative number indicates mixed material from more than one region.

### **nr**

### **nt**

### **nnode**

Total number of nodes in mesh.

### **nzone**

Total number of zones in mesh.

### **kmax**

Number of nodes in k-direction.

### **lmax**

Number of nodes in l-direction.

### **mmax**

Number of nodes in m-direction.

## Export Controlled Information

**r** 0 *ir*  
**x** 0 *ir*  
**z** 0 *ir*  
**xy** 0 *m*

Position of node *ir* (cm). NOTE: For a 2-d problem, “**xvar r**” is expanded to “**xvar x2d**” and “**xvar y2d**”. For a 3-d problem, “**xvar r**” is expanded to “**xvar x3d**”, “**xvar y3d**” and “**xvar z3d**” and “**xvar xy**” is expanded to “**xvar x3d**” and “**xvar y3d**”.

**-r** 0 *ir*  
**-x** 0 *ir*  
**-z** 0 *ir*

Position of node *ir* (cm) with respect to the position of the last node in the mesh.

**x2d** 0 *ir*  
**x3d** 0 *ir*

Position of node *ir* in x- (or r-) direction (cm). When this is used as an **xvar**, it is treated the same as **kx**, i.e. the k-coordinate is varied and the others are held fixed.

**y2d** 0 *ir*  
**y3d** 0 *ir*

Position of node *ir* in y- (or z-) direction (cm). When this is used as an **xvar**, it is treated the same as **ly**, i.e. the l-coordinate is varied and the others are held fixed.

**z2d** 0 *ir*  
**z3d** 0 *ir*

Position of node *ir* in z-direction (cm). When this is used as an **xvar**, it is treated the same as **mz**, i.e. the m-coordinate is varied and the others are held fixed.

**r2d** 0 *ir*  
**r3d** 0 *ir*

Distance from origin of node *ir* (cm).

**xzc** 0 *ir*

Position of center of zone *ir* in x-direction (cm). When this is used as an **xvar**, it is treated the same as **kx**, i.e. the k-coordinate is varied and the others are held fixed, but the total number of values is **kx-1**.

**yzc** 0 *ir*

Position of center of zone *ir* in y-direction (cm). When this is used as an **xvar**, it is treated the same as **ly**, i.e. the l-coordinate is varied and the others are held fixed, but the total number of values is **ly-1**.

**zzc** 0 *ir*

Position of center of zone *ir* in z-direction (cm). When this is used as an **xvar**, it is treated the same as **mz**, i.e. the m-coordinate is varied and the others are held fixed, but the total number of values is **mz-1**.

# Export Controlled Information

**rhodr**  $0 \ ir$

Column density from node  $ir$  to node  $ir+1$  ( $\text{g}/\text{cm}^2$ ).

**cdens**  $0 \ ir$

Column density from node 1 to node  $ir$  ( $\text{g}/\text{cm}^2$ ).

**vel**  $0 \ ir$

Speed at node  $ir$  ( $\text{cm/sec}$ ). In 1-d problems, this is a signed quantity denoting speed and direction. In 2-d and 3-d problems, this is the magnitude of the velocity.

**dvdr**  $0 \ ir$

Magnitude of velocity gradient at node  $ir$  ( $1/\text{sec}$ ).

**u2d**  $0 \ ir$

**u3d**  $0 \ ir$

Velocity component in x- (or r-) direction for node  $ir$  ( $\text{cm/sec}$ ).

**v2d**  $0 \ ir$

**v3d**  $0 \ ir$

Velocity component in y- (or z-) direction for node  $ir$  ( $\text{cm/sec}$ ).

**w2d**  $0 \ ir$

**w3d**  $0 \ ir$

Velocity component in z-direction for node  $ir$  ( $\text{cm/sec}$ ).

For the following quantities, the second index should specify the k-line  $k$  or the l-line  $l$  if the quantity is an independent x-variable, and should specify the node  $ir$  if the quantity is a dependent y-variable. The third index should specify the m-line  $m$  if the quantity is an independent x-variable for a 3-d problem. If the quantity is a dependent y-variable, the k-, l-, and m-indices will be obtained from the x-variables if necessary.

**k**  $0 \ (l,ir) \ (m)$

k-line index along l-line  $l$  (and m-line  $m$ , if 3-d) or for node  $ir$ .

**kr**  $0 \ (l,ir) \ (m)$

Distance along l-line  $l$  (and m-line  $m$ , if 3-d) from node at  $(l,l,m)$  to node  $ir$  at  $(k,l,m)$  ( $\text{cm}$ ).

**kx**  $0 \ (l,ir) \ (m)$

Distance along x- (or r-) axis (for l-line  $l$ , and m-line  $m$ , if 3-d) from node at  $(l,l,m)$  to node  $ir$  at  $(k,l,m)$  ( $\text{cm}$ ).

**ky**  $0 \ (l,ir) \ (m)$

Distance along y- (or z-) axis (for l-line  $l$ , and m-line  $m$ , if 3-d) from node at  $(l,l,m)$  to node  $ir$  at  $(k,l,m)$  ( $\text{cm}$ ).

# Export Controlled Information

**kz 0 (l,ir) (m)**

Distance along z-axis (for l-line  $l$ , and m-line  $m$ , if 3-d) from node at  $(l,l,m)$  to node  $ir$  at  $(k,l,m)$  (cm).

**l 0 (k,ir) (m)**

l-line index for k-line  $k$  (and m-line  $m$ , if 3-d) or for node  $ir$ .

**lr 0 (k,ir) (m)**

Distance along k-line  $k$  from node at  $(k,l,m)$  to node  $ir$  at  $(k,l,m)$  (cm).

**lx 0 (k,ir) (m)**

Distance along x- (or r-) axis (for k-line  $k$ , and m-line  $m$ , if 3-d) from node at  $(k,l,m)$  to node  $ir$  at  $(k,l,m)$  (cm).

**ly 0 (k,ir) (m)**

Distance along y- (or z-) axis (for k-line  $k$ , and m-line  $m$ , if 3-d) from node at  $(k,l,m)$  to node  $ir$  at  $(k,l,m)$  (cm).

**lz 0 (k,ir) (m)**

Distance along z-axis (for k-line  $k$ , and m-line  $m$ , if 3-d) from node at  $(k,l,m)$  to node  $ir$  at  $(k,l,m)$  (cm).

**m 0 ir**

m-line index along k-line  $k$  and l-line  $l$  (specified by  $ir = kmax*l+k$ ) or for node  $ir$ .

**mr 0 ir**

Distance along m-line  $m$  from node at  $(k,l,I)$  to node  $ir$  at  $(k,l,m)$  (cm).

**mx 0 ir**

Distance along x- (or r-) axis (for k-line  $k$  and l-line  $l$ ) from node at  $(k,l,I)$  to node  $ir$  at  $(k,l,m)$  (cm).

**my 0 ir**

Distance along y- (or z-) axis (for k-line  $k$  and l-line  $l$ ) from node at  $(k,l,I)$  to node  $ir$  at  $(k,l,m)$  (cm).

**mz 0 ir**

Distance along z-axis (for k-line  $k$  and l-line  $l$ ) from node at  $(k,l,I)$  to node  $ir$  at  $(k,l,m)$  (cm).

## Plasma properties:

(\_lte) signifies the value that would apply in LTE

# Export Controlled Information

**lteflag 0 ir**

Flag denoted whether material at node  $ir$  is considered to be in LTE (0) or NLTE ( $\neq 0$ ).

**ne 0 ir**

**ne\_lte 0 ir**

Electron density at node  $ir$  (1/cm $^3$ ).

**neb 0 ir**

Background electron density at node  $ir$  (1/cm $^3$ ).

**dnedt 0 ir**

Derivative w.r.t. time of electron density at node  $ir$  (1/cm $^3$ ).

**ni 0 ir**

Ion density at node  $ir$  (1/cm $^3$ ).

**nib 0 ir**

Background ion density at node  $ir$  (1/cm $^3$ ).

**ntot 0 ir**

Particle density at node  $ir$  (1/cm $^3$ ).

**rho 0 ir**

Mass density at node  $ir$  (g/cm $^3$ ).

**te 0 ir**

Electron temperature at node  $ir$  (K).

**tev 0 ir**

Electron temperature at node  $ir$  (eV).

**tev\_b 0 ir**

Background electron temperature at node  $ir$  (eV).

**te\_lte 0 ir**

Equivalent temperature at node  $ir$  (K) from opacity table.

**tev\_lte 0 ir**

Equivalent temperature at node  $ir$  (eV) from opacity table.

# Export Controlled Information

**tfermi**  $0 \ ir$

Fermi temperature at node  $ir$  (eV).

**ti**  $0 \ ir$

Ion temperature at node  $ir$  (K).

**tiv**  $0 \ ir$

Ion temperature at node  $ir$  (eV).

**tivb**  $0 \ ir$

Background ion temperature at node  $ir$  (eV).

**tr**  $0 \ ir$

Radiation temperature (from input) at node  $ir$  (K).

**trv**  $0 \ ir$

Radiation temperature (from input) at node  $ir$  (eV).

**tdfac**  $0 \ ir$

Dilution factor used to construct intensities from radiation temperature at node  $ir$ .

**tehot**  $i \ ir$

Temperature of  $i^{\text{th}}$  electron distribution at node  $ir$  (eV), where  $i = 0$  corresponds to the basic thermal distribution.

**nehot**  $i \ ir$

Number density of  $i^{\text{th}}$  electron distribution at node  $ir$  ( $1/\text{cm}^3$ ), where  $i = 0$  corresponds to the basic thermal distribution.

**ne\_thermal**  $0 \ ir$

Number density of basic thermal electron distribution at node  $ir$  ( $1/\text{cm}^3$ ) - same as  $i = 0$  electron distribution.

**emat**  $0 \ ir$

Material energy density at node  $ir$  ( $\text{erg}/\text{cm}^3$ ).

**emat\_e**  $0 \ ir$

Electron energy density at node  $ir$  ( $\text{erg}/\text{cm}^3$ ).

**emat\_i**  $0 \ ir$

Ion energy density at node  $ir$  ( $\text{erg}/\text{cm}^3$ ).

# Export Controlled Information

**erad 0 ir**

Radiation energy density at node *ir* (erg/cm<sup>3</sup>).

**etot 0 ir**

Total energy density (material + radiation) at node *ir* (erg/cm<sup>3</sup>).

**chemu 0 ir**

Chemical potential at node *ir* (eV).

**fugacity 0 ir**

Fugacity [ = exp(chemu/tev) ] at node *ir*.

**zeta 0 ir**

Degeneracy parameter [ = ln (1+fugacity) ] at node *ir*.

**cv 0 ir**

Specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/eV).

**cvegs 0 ir**

Specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/K).

**cp 0 ir**

Specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/eV).

**cvpgs 0 ir**

Specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/K).

**cv\_e 0 ir**

Electron specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/eV).

**cvecgs 0 ir**

Electron specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/K).

**cvetot 0 ir**

Electron specific heat (constant volume) including contribution from variation of internal energy with radiation at node *ir* (erg/cm<sup>3</sup>/eV).

**cvtotcgs 0 ir**

Electron specific heat (constant volume) including contribution from variation of internal energy with radiation at node *ir* (erg/cm<sup>3</sup>/K).

# Export Controlled Information

**cv\_i 0 ir**

Ion specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/eV).

**cviegs 0 ir**

Ion specific heat (constant volume) at node *ir* (erg/cm<sup>3</sup>/K).

**cp\_e 0 ir**

Electron specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/eV).

**cpecgs 0 ir**

Electron specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/K).

**cp\_i 0 ir**

Ion specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/eV).

**cpiags 0 ir**

Ion specific heat (constant pressure) at node *ir* (erg/cm<sup>3</sup>/K).

**gamma1 0 ir**

First adiabatic index, given by the ratio of specific heats **cp/cv** at node *ir*.

**gamma1\_e 0 ir**

First adiabatic index for electrons, given by the ratio of specific heats **cp\_e/cv\_e** at node *ir*.

**gamma1\_i 0 ir**

First adiabatic index for ions, given by the ratio of specific heats **cp\_i/cv\_i** at node *ir*.

**dedte 0 ir**

Derivative with respect to electron temperature of total energy at node *ir* (erg/cm<sup>3</sup>/eV).

**detti 0 ir**

Derivative with respect to ion temperature of total energy at node *ir* (erg/cm<sup>3</sup>/eV).

**dedne 0 ir**

Derivative with respect to electron density of total energy at node *ir* (erg).

**dedni 0 ir**

Derivative with respect to ion density of total energy at node *ir* (erg).

# Export Controlled Information

**dedr** *0 ir*

Derivative with respect to mass density of total energy at node *ir* (erg/g).

**pmat** *0 ir*

Material pressure at node *ir* (erg/cm<sup>3</sup>).

**pmat\_e** *0 ir*

Electron pressure at node *ir* (erg/cm<sup>3</sup>).

**pmat\_i** *0 ir*

Ion pressure at node *ir* (erg/cm<sup>3</sup>).

**prad** *0 ir*

Radiation pressure at node *ir* (erg/cm<sup>3</sup>).

**ptot** *0 ir*

Total pressure (material + radiation) at node *ir* (erg/cm<sup>3</sup>).

**dptde** *0 ir*

Derivative with respect to electron temperature of total pressure at node *ir* (erg/cm<sup>3</sup>/eV).

**dptti** *0 ir*

Derivative with respect to ion temperature of total pressure at node *ir* (erg/cm<sup>3</sup>/eV).

**dpdne** *0 ir*

Derivative with respect to electron density of total pressure at node *ir* (erg).

**dpdni** *0 ir*

Derivative with respect to ion density of total pressure at node *ir* (erg).

**dpdtr** *0 ir*

Derivative with respect to mass density of total pressure at node *ir* (erg/g).

**abar** *0 ir*

Average ion atomic weight at node *ir*.

**zbar** *0 ir*

**zbar\_lte** *0 ir*

Average ion charge at node *ir*.

## Export Controlled Information

**z2abar**  $0 \ ir$

Average of square of ion charge divided by atomic weight at node  $ir$ .

**z2bar**  $0 \ ir$

Average of square of ion charge at node  $ir$ .

**qmoment\_n**  $iz \ ir$

The  $n^{th}$  moment of the charge distribution of element  $iz$  at node  $ir$ .

**wplasma**  $0 \ ir$

Plasma frequency at node  $ir$  (eV).

**tauee**  $0 \ ir$

Electron-electron collision time at node  $ir$  (sec).

**tauei**  $0 \ ir$

Electron-ion collision time at node  $ir$  (sec).

**tauii**  $0 \ ir$

Ion-ion collision time at node  $ir$  (sec).

**cee**  $0 \ ir$

Electron-electron coupling coefficient at node  $ir$  (cm<sup>3</sup>/sec).

**cei**  $0 \ ir$

Electron-ion coupling coefficient at node  $ir$  (cm<sup>3</sup>/sec).

**cii**  $0 \ ir$

Ion-ion coupling coefficient at node  $ir$  (cm<sup>3</sup>/sec) for atomic mass and number set to 1.

**coullog**  $0 \ ir$

Coulomb logarithm for electron-ion collisions at node  $ir$ .

**couloge**  $0 \ ir$

Coulomb logarithm for electron-electron collisions at node  $ir$ .

**coulogi**  $0 \ ir$

Coulomb logarithm for ion-ion collisions at node  $ir$ .

# Export Controlled Information

**gplasma**  $0 \ ir$

Plasma coupling constant at node  $ir$ .

**econd**  $0 \ ir$

Electron thermal conduction coefficient at node  $ir$  (cm $^2$ /s).

**icond**  $0 \ ir$

Ion thermal conduction coefficient at node  $ir$  (cm $^2$ /s).

**bcond**  $0 \ ir$

Electrical conductivity at node  $ir$  (cm $^3$ /s).

**eresist**  $0 \ ir$

Electrical resistivity at node  $ir$  (ohm cm).

**bx**  $0 \ ir$

Component of magnetic field in x- (or r-) direction at node  $ir$  (G).

**by**  $0 \ ir$

Component of magnetic field in y- (or z-) direction at node  $ir$  (G).

**bz**  $0 \ ir$

Component of magnetic field in z- (or phi-) direction at node  $ir$  (G).

**bfield**  $0 \ ir$

Magnetic field strength at node  $ir$  (G).

**wtaue**  $0 \ ir$

Omega-tau at node  $ir$ .

## Electron Distribution:

**nee**

Number of electron groups.

**eenergy**  $0 \ 0 \ ie$

Energy at center of electron group  $ie$  (eV).

# Export Controlled Information

**ebins 0 0 ie**

Energy at boundaries of electron group *ie* (eV). Using this as an x-variable results in a histogram, with y-variable values evaluated at group centers.

**edist 0 ir ie**

Number density of non-thermal electron distribution at node *ir* in energy group *ie* (1/cm<sup>3</sup>/eV).

**nedist 0 ir ie**

Number density of total electron distribution (thermal + non-thermal + hot) at node *ir* in energy group *ie* (1/cm<sup>3</sup>/eV).

## Populations:

(\_lte) signifies the value that would apply in LTE

**anuc iz**

Atomic weight of element *iz*.

**znuc iz**

Atomic number of element *iz*.

**niso iz**

Number of isosequences in atomic model for element *iz*. If *iz* is zero, total number of isosequences in all atomic models.

**nlev iz**

Number of levels in atomic model for element *iz*. If *iz* is zero, total number of levels in all atomic models.

**level iz 0 iso i**

Index of state (*iso,i*) of element *iz*. If *iso* is zero, the index is with respect to the entire model, otherwise, the index is with respect to isoelectronic sequence *iso*.

**iso iz 0 iso i**

Isoelectronic sequence of state (*iso,i*) of element *iz*. When used as an **xvar**, *iso* and *i* will be ignored.

**ziso iz 0 iso**

Charge of isoelectronic sequence *iso* of element *iz*.

**npq iz 0 iso i**

Principal quantum number of state (*iso,i*) of element *iz*.

# Export Controlled Information

**nlq** *iz 0 iso i*

Angular momentum quantum number of state (*iso,i*) of element *iz*.

**npqmax** *iz 0 iso i*

Maximum principal quantum number of state (*iso,i*) of element *iz* before lines merge (Inglis-Teller limit).

**shell\_xx** *iz 0 iso i*

Shell occupation for state (*iso,i*) of element *iz*, where **xx** is an index denoting the shell (1=k-shell, 2=l-shell, 3=m-shell, ...).

**y** *iz ir iso i*

**y\_lte** *iz ir iso i*

Ion density of state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**dydt** *iz ir iso i*

Derivative with respect to electron temperature of ion density of state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>/eV).

**dydn** *iz ir iso i*

Derivative with respect to electron density of ion density of state (*iso,i*) of element *iz* at node *ir*.

**dydr** *iz ir iso i*

Derivative with respect to mass density of ion density of state (*iso,i*) of element *iz* at node *ir* (1/g).

**yfrac** *iz ir iso i*

**yfrac\_lte** *iz ir iso i*

Fractional population of state (*iso,i*) of element *iz* at node *ir*.

**yfrac1** *iz ir iso i*

**yfraciso** *iz ir iso i*

**yfrac1\_lte** *iz ir iso i*

**yfraciso\_lte** *iz ir iso i*

Fractional population of state (*iso,i*) of element *iz* relative to total population of isosequence (*iso,i*) at node *ir*.

**yg** *iz ir iso i*

**yg\_lte** *iz ir iso i*

Ion density divided by statistical weight of state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**ygfrac** *iz ir iso i*

**ygfrac\_lte** *iz ir iso i*

Fractional population divided by statistical weight of state (*iso,i*) of element *iz* at node *ir*.

## Export Controlled Information

**yeff0** *iz ir iso i*

Neutral-coupled component of ion density of state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**yeff1** *iz ir iso i*

Ion-coupled component of ion density of state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**y1** *iz ir iso*  
**yiso** *iz ir iso*  
**y1\_lte** *iz ir iso*  
**yiso\_lte** *iz ir iso*

Ion density of isoelectronic sequence *iso* of element *iz* at node *ir* (1/cm<sup>3</sup>).

**y1frac** *iz ir iso*  
**yisofrac** *iz ir iso*  
**y1frac\_lte** *iz ir iso*  
**yisofrac\_lte** *iz ir iso*

Fractional population of isoelectronic sequence *iso* of element *iz* at node *ir*.

**y1x** *iz ir iso*  
**yisox** *iz ir iso*

Ion density of isoelectronic sequence *iso* of element *iz* at node *ir*, evaluated at an intermediate iteration (1/cm<sup>3</sup>).

**dy1dt** *iz ir iso*  
**dyisodt** *iz ir iso*

Derivative with respect to electron temperature of ion density of isoelectronic sequence *iso* of element *iz* at node *ir* (1/cm<sup>3</sup>/eV).

**dy1dn** *iz ir iso*  
**dyisodn** *iz ir iso*

Derivative with respect to electron density of ion density of isoelectronic sequence *iso* of element *iz* at node *ir*.

**dy1dr** *iz ir iso*  
**dyisodr** *iz ir iso*

Derivative with respect to mass density of ion density of isoelectronic sequence *iso* of element *iz* at node *ir* (1/g).

**ni** *iz ir*  
**ytot** *iz ir*  
**ytot\_lte** *iz ir*

Total ion density of element *iz* at node *ir* (1/cm<sup>3</sup>).

# Export Controlled Information

**yshell\_n iz ir [iso]**

Number density of bound electrons in shell  $n$  due to element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ). If  $iso$  is present, only electrons from ions in isosequence  $iso$  are included. The shell index  $n$  refers to the principal quantum number ( $n=1$ : K shell,  $n=2$ : L shell,  $n=3$ : M shell, ...)

**yfshell\_n iz ir [iso]**

Fractional population of element  $iz$  at node  $ir$  with at least one electron in shell  $n$  or higher. If  $iso$  is present, only electrons from ions in isosequence  $iso$  are included. The shell index  $n$  refers to the principal quantum number ( $n=1$ : K shell,  $n=2$ : L shell,  $n=3$ : M shell, ...)

**yn\_n1\_n2\_n3\_n4 iz ir [iso]**

Number density of ions matching the specified occupations of up to 4 shells  $n1, n2, n3, n4$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ). If  $iso$  is present, only ions in isosequence  $iso$  are included. For example,

yn\_1        all ions with 1 K-shell electron  
yn\_1\_3      all ions with 1 K-shell + 3 L-shell electrons  
yn\_1\_3\_0     all ions with 1 K-shell + 3 L-shell + 0 M-shell electrons

where each of the examples is a subset of the preceding example.

**ynf\_n1\_n2\_n3\_n4 iz ir [iso]**

Fractional number density of ions matching the specified occupations of up to 4 shells  $n1, n2, n3, n4$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ). If  $iso$  is present, only ions in isosequence  $iso$  are included.

**ym\_n1\_n2\_n3\_n4 iz ir [iso]**

Number density of ions with occupations no greater than the specified occupations of up to 4 shells  $n1, n2, n3, n4$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ). If  $iso$  is present, only ions in isosequence  $iso$  are included. For example,

ym\_1        all ions with 0 or 1 K-shell electron  
ym\_1\_3      all ions with 0 or 1 K-shell + <=3 L-shell electrons  
ym\_1\_3\_2     all ions with 0 or 1 K-shell + <=3 L-shell + <=2 M-shell electrons

where each of the examples is a subset of the preceding example.

**ymf\_n1\_n2\_n3\_n4 iz ir [iso]**

Fractional number density of ions with occupations no greater than the specified occupations of up to 4 shells  $n1, n2, n3, n4$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ). If  $iso$  is present, only ions in isosequence  $iso$  are included.

**yiso\_v iz ir iso**

Virtual ion density of isoelectronic sequence  $iso$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ).

**yisofrac\_v iz ir iso**

Fractional virtual ion density (relative to the total real ion density **ytot**) of isoelectronic sequence  $iso$  of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ).

**ytot\_v iz ir**

Total ion density of virtual states of element  $iz$  at node  $ir$  ( $1/\text{cm}^3$ ).

# Export Controlled Information

**ytfrac\_v** *iz ir*

Fractional ion density (relative to the total real ion density **ytot**) of virtual states of element *iz* at node *ir* (1/cm<sup>3</sup>).

**yisoc** *iz ir iso*

Areal density of isoelectronic sequence *iso* of element *iz* from node 1 to node *ir* (1/cm<sup>2</sup>).

**gammatot** *iz ir iso i*

Total destruction rate of state (*iso,i*) of element *iz* at node *ir* (1/s).

**gamma** *iz ir iso i*

Destruction rate of state (*iso,i*) of element *iz* at node *ir* used in calculation of line widths (1/s).

**nbound** *iz ir*

**nbound\_lte** *iz ir*

Average number of bound electrons for ions of element *iz* at node *ir*.

**zbar** *iz ir*

**zbar\_lte** *iz ir*

Average ion charge of element *iz* at node *ir*.

**zbnd** *iz ir*

**zbnd\_lte** *iz ir*

Average number of bound electron for element *iz* at node *ir*.

**ilevmin** *iz ir*

Minimum index for levels included for element *iz* at node *ir*.

**ilevmax** *iz ir*

Maximum index for levels included for element *iz* at node *ir*.

**isomin** *iz ir*

Minimum isoelectronic sequence included for ions of element *iz* at node *ir*.

**isomax** *iz ir*

Maximum isoelectronic sequence included for ions of element *iz* at node *ir*.

**eiso0** *iz 0 iso*

Energy of ground state of isoelectronic sequence *iso* of element *iz*, relative to zero energy for the ground state of the lowest isoelectronic sequence with no continuum lowering (eV).

# Export Controlled Information

**eiso** *iz ir iso*

Energy of ground state of isoelectronic sequence *iso* of element *iz* at node *ir*, relative to zero energy for the ground state of the lowest isoelectronic including effects of continuum lowering (eV).

**deiso0** *iz 0 iso*

Ionization energy of ground state of isoelectronic sequence *iso* of element *iz*, relative to zero energy for the ground state of the lowest isoelectronic sequence with no continuum lowering (eV).

**deiso** *iz ir iso*

Ionization energy of ground state of isoelectronic sequence *iso* of element *iz* at node *ir*, relative to zero energy for the ground state of the lowest isoelectronic including effects of continuum lowering (eV).

**elev0** *iz 0 iso i*

Energy of state (*iso,i*) of element *iz*, relative to zero energy for the ground state of the lowest isoelectronic sequence with no continuum lowering (eV).

**elev** *iz ir iso i*

Energy of state (*iso,i*) of element *iz* at node *ir*, relative to zero energy for the ground state of the lowest isoelectronic including effects of continuum lowering (eV).

**glev0** *iz ir iso i*

Statistical weight of state (*iso,I*) of element *iz* at node *ir* with no continuum lowering.

**glev** *iz ir iso i*

Statistical weight of state (*iso,i*) of element *iz* at node *ir* including effects of continuum lowering.

**gion0** *iz ir iso*

Total statistical weight of isoelectronic sequence *iso* of element *iz* at node *ir* with no continuum lowering.

**gion** *iz ir iso*

Total statistical weight of isoelectron sequence *iso* of element *iz* at node *ir* including effects of continuum lowering.

**wlev** *iz 0 iso i*

Width of state (*iso,i*) of element *iz* (eV).

**plev** *iz ir iso i*

Statistical weight multiplier (degeneracy lowering) of state (*iso,i*) of element *iz* at node *ir*.

**delev** *iz ir iso i*

Energy lowering required for state (*iso,i*) of element *iz* at node *ir* to vanish (eV).

# Export Controlled Information

**elower** *iz ir iso i*

Energy lowering of state (*iso,i*) of element *iz* at node *ir* (eV).

**elowerdt** *iz ir iso i*

Derivative w.r.t. electron temperature of energy lowering of state (*iso,i*) of element *iz* at node *ir*.

**elowerdn** *iz ir iso i*

Derivative w.r.t. log(electron density) of energy lowering of state (*iso,i*) of element *iz* at node *ir* (eV).

**eint0** *iz ir*

Internal energy of element *iz* at node *ir* with no continuum lowering (erg/cm<sup>3</sup>).

**deint0dt** *iz ir*

Derivative w.r.t. electron temperature of internal energy of element *iz* at node *ir* with no continuum lowering (erg/cm<sup>3</sup>/eV).

**eint** *iz ir*

Internal energy of element *iz* at node *ir* including effects of continuum lowering (erg/cm<sup>3</sup>).

**eintcl** *iz ir*

Change in internal energy of element *iz* at node *ir* due to continuum lowering (erg/cm<sup>3</sup>).

**deintdt** *iz ir*

Derivative w.r.t. electron temperature of internal energy of element *iz* at node *ir* including effects of continuum lowering (erg/cm<sup>3</sup>/eV).

**deintdn** *iz ir*

Derivative w.r.t. electron density of internal energy of element *iz* at node *ir* including effects of continuum lowering (erg).

**einttot** *0 ir*

Internal energy for all elements at node *ir* (erg/cm<sup>3</sup>).

**deinttdt** *0 ir*

Derivative w.r.t. electron temperature of internal energy of all elements at node *ir* (erg/cm<sup>3</sup>/eV).

**deinttdt** *0 ir*

Derivative w.r.t. electron density of internal energy of all elements at node *ir* (erg).

**deinttdr** *0 ir*

Derivative w.r.t. mass density of internal energy of all elements at node *ir* (erg/g).

# Export Controlled Information

**pfnc0** *iz ir*

Partition function of element *iz* at node *ir* with no continuum lowering.

**pfnc** *iz ir*

Partition function of element *iz* at node *ir* including effects of continuum lowering.

## Rates:

- (+) signifies the transition from lower to upper state(s)
- (-) signifies the transition from upper to lower state(s)

Edits specific to a particular transition *itr* and node *ir* will return a zero value if that transition has not been calculated for that node. This will occur if the populations involved are not included for that node.

The rate matrix edits, **rates**, **rates+**, and **rates-** behave differently than other edit quantities and should not be combined with other *yvars* in the same edit specification.

**acoeff** *itr ir*

Einstein A-value (spontaneous de-excitation rate) for photoexcitation transition *itr* (1/s). Although this quantity is independent of position and plasma conditions, a valid node index *ir* must be present

**avoigt0** *itr ir*

Voigt parameter for photoexcitation transition *itr* at node *ir*.

**uvoigt0** *itr ir*

Maximum value of Voigt function for photoexcitation transition *itr* at node *ir*.

**gain** *itr ir*

Gain for photoexcitation transition *itr* at node *ir* (1/cm).

**aemis** *itr ir*

Spontaneous emitted power for photoexcitation transition *itr* at node *ir* (erg/ cm<sup>3</sup>/sec).

**widthn** *itr ir*

Natural width for photoexcitation transition *itr* at node *ir* (eV).

**widths** *itr ir*

Stark width for photoexcitation transition *itr* at node *ir* (eV).

**widthd** *itr ir*

Doppler width for photoexcitation transition *itr* at node *ir* (eV).

# Export Controlled Information

**pump** *itr ir*

Photon intensity for photoexcitation transition *itr* at node *ir* (photons/mode).

**escprob** *itr ir*

Escape probability for photoexcitation transition *itr* at node *ir*.

**bsplit** *itr ir*

Zeeman splitting for photoexcitation transition *itr* at node *ir* (eV).

**c\_rate+** *itr ir*

**c\_rate-** *itr ir*

Rate for electron collisional transition *itr* at node *ir* (1/sec).

**p\_rate+** *itr ir*

**p\_rate-** *itr ir*

Rate for photo transition *itr* at node *ir* (1/sec).

**a\_rate+** *itr ir*

**a\_rate-** *itr ir*

Rate for auger transition *itr* at node *ir* (1/sec).

**d\_rate+** *itr ir*

**d\_rate-** *itr ir*

Rate for ion collisional transition *itr* at node *ir* (1/sec).

**t\_rate+** *itr ir*

**t\_rate-** *itr ir*

Rate for all processes for transition *itr* at node *ir* (1/sec).

**cn\_rate+** *itr ir*

**cn\_rate-** *itr ir*

Net rate for electron collisional transition *itr* at node *ir* (1/sec).

**pn\_rate+** *itr ir*

**pn\_rate-** *itr ir*

Net rate for photo transition *itr* at node *ir* (1/sec).

**an\_rate+** *itr ir*

**an\_rate-** *itr ir*

Net rate for auger transition *itr* at node *ir* (1/sec).

**dn\_rate+** *itr ir*

## Export Controlled Information

**dn\_rate- *itr ir***

Net rate for ion collisional transition *itr* at node *ir* (1/sec).

**tn\_rate+ *itr ir***

**tn\_rate- *itr ir***

Net rate for all processes for transition *itr* at node *ir* (1/sec).

**c\_tot+ *itr ir***

**c\_tot- *itr ir***

**c\_net *itr ir***

Total rate for electron collisional transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (1/sec).

**p\_tot+ *itr ir***

**p\_tot- *itr ir***

**p\_net *itr ir***

Total rate for photo transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (1/sec).

**a\_tot+ *itr ir***

**a\_tot- *itr ir***

**a\_net *itr ir***

Total rate for auger transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (1/sec).

**d\_tot+ *itr ir***

**d\_tot- *itr ir***

**d\_net *itr ir***

Total rate for ion collisional transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (1/sec).

**t\_tot+ *itr ir***

**t\_tot- *itr ir***

**t\_net *itr ir***

Total rate for all processes for transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (1/sec).

**e\_tot+ *itr ir***

**e\_tot- *itr ir***

Total effective rate for all processes for transition *itr* at node *ir*, (1/sec).

**ec\_rate+ *itr ir***

**ec\_rate- *itr ir***

Energy contribution to free electrons for electron collisional transition *itr* at node *ir* (eV).

## Export Controlled Information

**ep\_rate+ *itr ir***  
**ep\_rate- *itr ir***

Energy contribution to free electrons for photo transition *itr* at node *ir* (eV).

**ea\_rate+ *itr ir***  
**ea\_rate- *itr ir***

Energy contribution to free electrons for auger transition *itr* at node *ir* (eV).

**ed\_rate+ *itr ir***  
**ed\_rate- *itr ir***

Energy contribution to free electrons for ion collisional transition *itr* at node *ir* (eV).

**ce\_rate+ *itr ir***  
**ce\_rate- *itr ir***

Energy deposition rate for electron collisional transition *itr* at node *ir* (eV/sec).

**pe\_rate+ *itr ir***  
**pe\_rate- *itr ir***

Energy deposition rate for photo transition *itr* at node *ir* (eV/sec).

**ae\_rate+ *itr ir***  
**ae\_rate- *itr ir***

Energy deposition rate for auger transition *itr* at node *ir* (eV/sec).

**de\_rate+ *itr ir***  
**de\_rate- *itr ir***

Energy deposition rate for ion collisional transition *itr* at node *ir* (eV/sec).

**te\_rate+ *itr ir***  
**te\_rate- *itr ir***

Energy deposition rate for all processes for transition *itr* at node *ir* (eV/sec).

**ee\_rate+ *itr ir***  
**ee\_rate- *itr ir***

Effective energy deposition rate for all processes for transition *itr* at node *ir* (eV/sec).

**rp\_rate+ *itr ir***  
**rp\_rate- *itr ir***

Energy contribution to radiation field for photo transition *itr* at node *ir* (eV).

**ra\_rate+ *itr ir***  
**ra\_rate- *itr ir***

Energy contribution to radiation field for auger transition *itr* at node *ir* (eV).

## Export Controlled Information

**pr\_rate+ *itr ir***  
**pr\_rate- *itr ir***

Energy contribution rate to radiation field for photo transition *itr* at node *ir* (eV/sec).

**ar\_rate+ *itr ir***  
**ar\_rate- *itr ir***

Energy contribution rate to radiation field for auger transition *itr* at node *ir* (eV/sec).

**tr\_rate+ *itr ir***  
**tr\_rate- *itr ir***

Energy contribution rate to radiation field for all processes for transition *itr* at node *ir* (eV/sec).

**ce\_tot+ *itr ir***  
**ce\_tot- *itr ir***  
**ce\_net *itr ir***

Total energy deposition rate for electron collisional transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**pe\_tot+ *itr ir***  
**pe\_tot- *itr ir***  
**pe\_net *itr ir***

Total energy deposition rate for photo transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**ae\_tot+ *itr ir***  
**ae\_tot- *itr i***  
**ae\_net *itr ir***

Total energy deposition rate for auger transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**de\_tot+ *itr ir***  
**de\_tot- *itr ir***  
**de\_net *itr ir***

Total energy deposition rate for ion collisional transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**te\_tot+ *itr ir***  
**te\_tot- *itr ir***  
**te\_net *itr ir***

Total energy deposition rate for all processes for transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**pr\_tot+ *itr ir***  
**pr\_tot- *itr ir***

# Export Controlled Information

**pr\_net** *itr ir*

Total energy deposited into radiation field for photo transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**ar\_tot+** *itr ir*

**ar\_tot-** *itr ir*

**ar\_net** *itr ir*

Total energy deposited into radiation field for auger transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**tr\_tot+** *itr ir*

**tr\_tot-** *itr ir*

**tr\_net** *itr ir*

Total energy deposited into radiation field for all processes for transition *itr* at node *ir*, averaged over initial state(s) and summed over final state(s) (eV/sec).

**y\_flux+** *itr ir*

**y\_flux-** *itr ir*

**y\_flux** *itr ir*

Population flux (unidirectional or net) between levels specified by transition *itr* at node *ir* (cm<sup>3</sup>/sec).

**ebbtot** *iz ir iso*

Total emission due to photo de-excitations from states in isosequence *iso* of element *iz* at node *ir* (erg/cm<sup>3</sup>/sec/ster). If *iso* is 0, transitions from all isosequences will be included.

**ebbtot0** *iz ir iso*

Total emission per ion of element *iz* due to photo de-excitations from states in isosequence *iso* of element *iz* at node *ir* (erg/sec/ster). If *iso* is 0, transitions from all isosequences will be included.

**ebbtot1** *iz ir iso*

Total emission per ion of isosequence (*iz,iso*) due to photo de-excitations from states in isosequence *iso* of element *iz* at node *ir* (erg/sec/ster). If *iso* is 0, transitions from all isosequences will be included.

**isorate+** *iz ir iso*

Total ionization rate of isosequence *iso* to isosequence *iso+1* of element *iz* at node *ir* due to all processes (1/s).

**isorate-** *iz ir iso*

Total recombination rate from isosequence *iso* to isosequence *iso-1* of element *iz* at node *ir* due to all processes (1/s).

**isoraten** *iz ir iso*

Net ionization rate of isosequence *iso* to isosequence *iso+1* of element *iz* at node *ir* due to all processes (1/s).

# Export Controlled Information

**isorate+c iz ir iso**

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isorate-\_c iz ir iso**

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isoraten\_c iz ir iso**

Net ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isorate+\_p iz ir iso**

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to photoionization processes (1/s).

**isorate-\_p iz ir iso**

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to photo recombination processes (1/s).

**isoraten\_p iz ir iso**

Net ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to photoionization processes (1/s).

**isorate+\_a iz ir iso**

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**isorate-\_a iz ir iso**

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**isoraten\_a iz ir iso**

Net ionization rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**isorate+\_x iz ir iso**

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to charge exchange processes (1/s).

**isorate-\_x iz ir iso**

## Export Controlled Information

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to charge exchange processes (1/s).

**isoraten\_x**  $iz\ ir\ iso$

Net ionization rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to charge exchange processes (1/s).

**isoflux+**  $iz\ ir\ iso$

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to all processes (1/s).

**isoflux-**  $iz\ ir\ iso$

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to all processes (1/s).

**isofluxn**  $iz\ ir\ iso$

Net ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to all processes (1/s).

**isoflux+\_c**  $iz\ ir\ iso$

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isoflux-\_c**  $iz\ ir\ iso$

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isofluxn\_c**  $iz\ ir\ iso$

Net ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to electron collisional processes (1/s).

**isoflux+\_p**  $iz\ ir\ iso$

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to photoionization processes (1/s).

**isoflux-\_p**  $iz\ ir\ iso$

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to photo recombination processes (1/s).

**isofluxn\_p**  $iz\ ir\ iso$

Net ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to photoionization processes (1/s).

# Export Controlled Information

**isoflux+\_a iz ir iso**

Total ionization rate of isosequence  $iso$  to isosequence  $iso+1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**isoflux-\_a iz ir iso**

Total recombination rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**isofluxn\_a iz ir iso**

Net ionization rate from isosequence  $iso$  to isosequence  $iso-1$  of element  $iz$  at node  $ir$  due to auger processes (1/s).

**gamma\_cx iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to electron collisional excitation processes (1/s).

**gamma\_ci iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to electron collisional ionization processes (1/s).

**gamma\_c iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to all electron collisional processes (1/s).

**gamma\_px iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to photoexcitation processes (1/s).

**gamma\_pi iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to photoionization processes (1/s).

**gamma\_p iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to all photo processes (1/s).

**gamma\_a iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to all auger processes (1/s).

**gamma\_dx iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to ion collisional excitation processes (1/s).

**gamma\_di iz ir iso i**

Total destruction rate of state  $(iso,i)$  of element  $iz$  at node  $ir$  due to ion collisional ionization processes (1/s).

# Export Controlled Information

**gamma\_d iz ir iso i**

Total destruction rate of state (*iso,i*) of element *iz* at node *ir* due to all ion collisional processes (1/s).

**eigenval iz ir 0 i**

*i*<sup>th</sup> eigenvalue of the rate matrix for element *iz* at node *ir*. Requesting this edit is the same as requesting both **eigenv\_r** and **eigenv\_i**. The eigenvalues are sorted in descending order of the real part of the eigenvalue if *i* is positive, and in ascending order if *i* is negative.

**eigenv\_r iz ir 0 i**

Real part of the *i*<sup>th</sup> eigenvalue of the rate matrix for element *iz* at node *ir*. The eigenvalues are sorted in descending order of the real part of the eigenvalue if *i* is positive, and in ascending order if *i* is negative.

**eigenv\_i iz ir 0 i**

Imaginary part of the *i*<sup>th</sup> eigenvalue of the rate matrix for element *iz* at node *ir*. The eigenvalues are sorted in descending order of the real part of the eigenvalue if *i* is positive, and in ascending order if *i* is negative.

**eigenvec iz ir i j**

*j*<sup>th</sup> component of *i*<sup>th</sup> eigenvalue of the rate matrix for element *iz* at node *ir*. Requesting this edit is the same as requesting both **eigenvec\_r** and **eigenvec\_i**.

**eigenvec\_r iz ir i j**

Real part of the *j*<sup>th</sup> component of *i*<sup>th</sup> eigenvector of the rate matrix for element *iz* at node *ir*.

**eigenvec\_i iz ir i j**

Imaginary part of the *j*<sup>th</sup> component of *i*<sup>th</sup> eigenvector of the rate matrix for element *iz* at node *ir*.

**fillpct iz ir iso**

**fillpct- iz ir iso**

**fillpct+ iz ir iso**

Sparsity measure of the rate matrix of element *iz* at node *ir*. If *iso* = 0, this applies to the entire rate matrix. If *iso* ≠ 0, this applies to the block connecting isoelectronic states (*iso,iso*) for **fillpct**, (*iso-1,iso*) for **fillpct-**, and (*iso+1,iso*) for **fillpct+**.

**rates iz ir iso**

**rates- iz ir iso**

**rates+ iz ir iso**

The rate matrix of element *iz* at node *ir*. If *iso* = 0, this applies to the entire rate matrix. If *iso* ≠ 0, this applies to the block connecting isoelectronic states (*iso,iso*) for **rates**, (*iso-1,iso*) for **rates-**, and (*iso+1,iso*) for **rates+**.

For each *xvar*, put an image of the rate matrix in the binary plot file (PDB). The edit quantity is the logarithm (base 10) of the non-zero rates (with the rate matrix diagonals made positive). Zero entries are adjusted to fall below the scale of all non-zero entries. The images may be viewed with PDBVIEW. These edits do not appear in the ascii output files.

# Export Controlled Information

## Heating rates:

- $x = "a"$  signifies auger processes
- $x = "c"$  signifies electron collisional processes
- $x = "p"$  signifies photo processes

**heatb**  $0 \ ir$

Free electron heating rate due to bremsstrahlung absorption at node  $ir$  (eV/sec).

**heate**  $0 \ ir$

Rate of increase in internal energy due to atomic transitions at node  $ir$  (eV/sec).

**heatx**  $iz \ ir$

Free electron heating rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (eV/sec).

**heati**  $iz \ ir$

Ion heating rate due to ion collisional processes from element  $iz$  at node  $ir$  (eV/sec).

**heatj**  $0 \ ir$

Energy gain rate due to radiation absorption at node  $ir$  (eV/sec).

**heatl**  $0 \ ir$

Energy gain rate due to laser absorption at node  $ir$  (eV/sec).

**coolb**  $0 \ ir$

Free electron cooling rate due to bremsstrahlung emission at node  $ir$  (eV/sec).

**coole**  $0 \ ir$

Rate of decrease in internal energy due to atomic transitions at node  $ir$  (eV/sec).

**coolx**  $iz \ ir$

Free electron cooling rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (eV/sec).

**cooli**  $iz \ ir$

Ion cooling rate due to ion collisional processes from element  $iz$  at node  $ir$  (eV/sec).

**coolj**  $0 \ ir$

Energy loss rate due to radiation emission at node  $ir$  (eV/sec).

**pgain**  $0 \ ir$

Energy gain due to radiation absorption at node  $ir$  per ion (ergs cm<sup>3</sup>/sec).

# Export Controlled Information

**ploss**  $0 \ ir$

Energy loss rate due to radiation emission at node  $ir$  per ion (ergs cm<sup>3</sup>/sec).

**pnet**  $0 \ ir$

Net energy loss rate due to radiation absorption / emission at node  $ir$  per ion per free electron (ergs cm<sup>6</sup>/sec).

**heatbt**  $0 \ ir$

Net free electron heating rate due to bremsstrahlung absorption and emission at node  $ir$  (eV/sec).

**heatet**  $0 \ ir$

Net rate of increase in internal energy due to atomic transitions at node  $ir$  (eV/sec).

**heatxt**  $iz \ ir$

Net free electron heating rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (eV/sec).

**heatit**  $iz \ ir$

Net ion heating rate due to ion collisional processes from element  $iz$  at node  $ir$  (eV/sec).

**heatjt**  $0 \ ir$

Net energy gain rate due to radiation absorption and emission at node  $ir$  (eV/sec).

**heatt**  $0 \ ir$

Net free electron heating rate due to all processes at node  $ir$  (eV/sec).

**heatti**  $0 \ ir$

Net ion heating rate due to all processes at node  $ir$  (eV/sec).

**heatchk**  $0 \ ir$

Net energy imbalance (free electron heating rate + rate of increase in internal energy – net energy gain due to radiation absorption and emission) at node  $ir$  (eV/sec).

**dhtbdt**  $0 \ ir$

Derivative w.r.t. electron temperature of free electron heating rate due to bremsstrahlung absorption at node  $ir$  (1/sec).

**dhtbdn**  $0 \ ir$

Derivative w.r.t. electron density of free electron heating rate due to bremsstrahlung absorption at node  $ir$  (eVcm<sup>3</sup>/sec).

# Export Controlled Information

## **dhtedt 0 ir**

Derivative w.r.t. electron temperature of rate of increase of internal energy due to atomic transitions at node *ir* (1/sec).

## **dhtedn 0 ir**

Derivative w.r.t. electron density of rate of increase of internal energy due to atomic transitions at node *ir* (eVcm<sup>3</sup>/sec).

## **dhtxdt iz ir**

Derivative w.r.t. electron temperature of free electron heating rate due to atomic processes of type *x* from element *iz* at node *ir* (1/sec).

## **dhtxdn iz ir**

Derivative w.r.t. electron density of free electron heating rate due to atomic processes of type *x* from element *iz* at node *ir* (eVcm<sup>3</sup>/sec).

## **dhtidt iz ir**

Derivative w.r.t. ion temperature of ion heating rate due to ion collisional processes from element *iz* at node *ir* (1/sec).

## **dhtjdt 0 ir**

Derivative w.r.t. electron temperature of energy gain rate due to radiation absorption at node *ir* (1/sec).

## **dhtjdn 0 ir**

Derivative w.r.t. electron density of energy gain rate due to radiation absorption at node *ir* (eVcm<sup>3</sup>/sec).

## **dhtldt 0 ir**

Derivative w.r.t. electron temperature of energy gain rate due to laser absorption at node *ir* (1/sec).

## **dhtldn 0 ir**

Derivative w.r.t. electron density of energy gain rate due to laser absorption at node *ir* (eVcm<sup>3</sup>/sec).

## **dclbdt 0 ir**

Derivative w.r.t. electron temperature of free electron cooling rate due to bremsstrahlung emission at node *ir* (1/sec).

## **dclbdn 0 ir**

Derivative w.r.t. electron density of free electron cooling rate due to bremsstrahlung emission at node *ir* (eVcm<sup>3</sup>/sec).

## **dcledt 0 ir**

## Export Controlled Information

Derivative w.r.t. electron temperature of rate of decrease of internal energy due to atomic transitions at node  $ir$  (1/sec).

**dclledn**  $0\ ir$

Derivative w.r.t. electron density of rate of decrease of internal energy due to atomic transitions at node  $ir$  (eVcm $^3$ /sec).

**dclxdt**  $iz\ ir$

Derivative w.r.t. electron temperature of free electron cooling rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (1/sec).

**dclxdn**  $iz\ ir$

Derivative w.r.t. electron density of free electron cooling rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (eVcm $^3$ /sec).

**dclidt**  $iz\ ir$

Derivative w.r.t. ion temperature of ion cooling rate due to ion collisional processes from element  $iz$  at node  $ir$  (1/sec).

**dcljdt**  $0\ ir$

Derivative w.r.t. electron temperature of energy loss rate due to radiation emission at node  $ir$  (1/sec).

**dcljdn**  $0\ ir$

Derivative w.r.t. electron density of energy loss rate due to radiation emission at node  $ir$  (eVcm $^3$ /sec).

**dhtxtdt**  $iz\ ir$

Derivative w.r.t. electron temperature of net free electron heating rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (1/sec).

**dhtxtdn**  $iz\ ir$

Derivative w.r.t. electron density of net free electron heating rate due to atomic processes of type  $x$  from element  $iz$  at node  $ir$  (eVcm $^3$ /sec).

**dhtitdt**  $iz\ ir$

Derivative w.r.t. ion temperature of net ion heating rate due to ion collisional processes from element  $iz$  at node  $ir$  (1/sec).

**dhtttdt**  $0\ ir$

Derivative w.r.t. electron temperature of net free electron heating rate due to all processes at node  $ir$  (1/sec).

**dhtttdn**  $0\ ir$

Derivative w.r.t. electron density of net free electron heating rate due to all processes at node  $ir$  (eVcm $^3$ /sec).

# Export Controlled Information

**dhtetdt** 0 *ir*

Derivative w.r.t. electron temperature of net rate of increase of internal energy due to atomic transitions at node *ir* (1/sec).

**dhtetdn** 0 *ir*

Derivative w.r.t. electron density of net rate of increase of internal energy due to atomic transitions at node *ir* (eVcm<sup>3</sup>/sec).

**dhtjtdt** 0 *ir*

Derivative w.r.t. electron temperature of net energy gain rate due to radiation absorption and emission at node *ir* (1/sec).

**dhtjtdn** 0 *ir*

Derivative w.r.t. electron density of net energy gain rate due to radiation absorption and emission at node *ir* (eVcm<sup>3</sup>/sec).

**dhtchkdt** 0 *ir*

Derivative w.r.t. electron temperature of net energy imbalance (free electron heating rate + rate of increase in internal energy – net energy gain due to radiation absorption and emission) at node *ir* (1/sec).

**dhtchkdn** 0 *ir*

Derivative w.r.t. electron density of net energy imbalance (free electron heating rate + rate of increase in internal energy – net energy gain due to radiation absorption and emission) at node *ir* (eVcm<sup>3</sup>/sec).

**heff0** *iz ir iso i*

Neutral-coupled component of heating rate for state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**heff1** *iz ir iso i*

Ion-coupled component of heating rate for state (*iso,i*) of element *iz* at node *ir* (1/cm<sup>3</sup>).

**emistot** 0 *ir*

Energy loss rate due to radiation emission at node *ir* (ergs/cm<sup>3</sup>/sec).

**emisnet** 0 *ir*

Net energy loss rate due to radiation absorption / emission at node *ir* (ergs/cm<sup>3</sup>/sec).

**econd** 0 *ir*

Electron thermal conduction coefficient at node *ir* (cm<sup>2</sup>/s).

**icond** 0 *ir*

Ion thermal conduction coefficient at node *ir* (cm<sup>2</sup>/s).

# Export Controlled Information

**esrce 0 ir**

Energy sources into free electrons at node *ir* (erg/cm<sup>3</sup>/s).

**desrcedt 0 ir**

Derivative w.r.t. electron temperature of energy sources into free electrons at node *ir* (erg/cm<sup>3</sup>/s).

**esrcev 0 ir**

Free electron heating rate due to energy sources at node *ir* (eV/s).

**esrci 0 ir**

Energy sources into ions at node *ir* (erg/cm<sup>3</sup>/s).

**desrcidt 0 ir**

Derivative w.r.t. ion temperature of energy sources into ions at node *ir* (erg/cm<sup>3</sup>/s).

**esrciv 0 ir**

Ion heating rate due to energy sources at node *ir* (eV/s).

**desrce 0 ir**

Derivative of free electron heating rate at node *ir* w.r.t. temperature (1/s).

**desrci 0 ir**

Derivative of ion heating rate at node *ir* w.r.t. temperature (1/s).

## Continuum radiation:

- (+) signifies the direction of increasing *ir* in 1-d geometries
- (-) signifies the direction of decreasing *ir* in 1-d geometries
- (+,-) are equivalent in 2-d geometries

**ifr 0 0 igp**

Continuum group index.

**nf**

Number of continuum groups.

**energy 0 0 igp**

Energy at center of continuum group *igp* (eV).

**ebins 0 0 igp**

## Export Controlled Information

Energy at boundaries of continuum group *igp* (eV). Using this as an x-variable results in a histogram, with y-variable values evaluated at group centers.

**freq** 0 0 *igp*

Frequency at center of continuum group *igp* (Hz).

**fbins** 0 0 *igp*

Frequency at boundaries of continuum group *igp* (Hz). Using this as an x-variable results in a histogram, with y-variable values evaluated at group centers.

**nu** 0 0 *igp*

Frequency at center of continuum group *igp* (Hz).

**wvl** 0 0 *igp*

Wavelength at center of continuum group *igp* (cm).

**wbins** 0 0 *igp*

Wavelength at boundaries of continuum group *igp* (cm). Using this as an x-variable results in a histogram, with y-variable values evaluated at group centers.

**nray**

Number of directions used in transfer calculation.

**neray**

Number of additional directions used in transfer calculation to satisfy edit requests.

**ndtct**

Number of detectors currently defined for transfer calculation.

**rabsn** 0 *ir*

**rkap** 0 *ir*

Rosseland mean absorption coefficient, including scattering, at node *ir* (1/cm).

**rabsn0** 0 *ir*

Rosseland mean absorption coefficient, without scattering, at node *ir* (1/cm).

**rabsn\_nl** 0 *ir*

**rkap\_nl** 0 *ir*

Rosseland mean absorption coefficient, not including contributions of photoexcitations, at node *ir* (1/cm).

**ropac** 0 *ir*

Rosseland mean opacity, at node *ir* (cm<sup>2</sup>/g).

# Export Controlled Information

**ropac\_nl 0 ir**

Rosseland mean opacity, not including contributions of photoexcitations, at node *ir* (cm<sup>2</sup>/g).

**pabsn 0 ir**  
**pkap 0 ir**

Planck mean absorption coefficient at node *ir* (1/cm).

**pabsn\_nl 0 ir**  
**pkap\_nl 0 ir**

Planck mean absorption coefficient, not including contributions of photoexcitations, at node *ir* (1/cm).

**popac 0 ir**

Planck mean opacity, at node *ir* (cm<sup>2</sup>/g).

**popac\_nl 0 ir**

Planck mean opacity, not including contributions of photoexcitations, at node *ir* (cm<sup>2</sup>/g).

**eabsn 0 ir**  
**ekap 0 ir**

Emission mean absorption coefficient at node *ir* (1/cm).

**eabsn\_nl 0 ir**  
**ekap\_nl 0 ir**

Emission mean absorption coefficient, not including contributions of photoexcitations, at node *ir* (1/cm).

**eopac 0 ir**

Emission mean opacity, at node *ir* (cm<sup>2</sup>/g).

**eopac\_nl 0 ir**

Emission mean opacity, not including contributions of photoexcitations, at node *ir* (cm<sup>2</sup>/g).

**absn 0 ir igr**  
**ckap 0 ir igr**  
**ckappa 0 ir igr**

Absorption coefficient at node *ir* for group *igr* (1/cm).

**absn\_nl 0 ir igr**  
**ckap\_nl 0 ir igr**  
**ckappa\_nl 0 ir igr**

Absorption coefficient, not including contributions of photoexcitations, at node *ir* for group *igr* (1/cm).

# Export Controlled Information

**absn\_hk** *iz ir igp*

Absorption coefficient from Henke tables for element *iz* at node *ir* for group *igp* (1/cm). If *iz*=0, the absorption coefficient will be for the combination of elements at node *ir*. It is not necessary to specify Henke opacities for the material to edit this quantity.

**dadt** *0 ir igp*

Derivative w.r.t. electron temperature of absorption coefficient at node *ir* for group *igp* (1/cm/eV).

**dadn** *0 ir igp*

Derivative w.r.t. log(electron density) of absorption coefficient at node *ir* for group *igp* (1/cm).

**opac** *0 ir igp*

**copac** *0 ir igp*

Opacity at node *ir* for group *igp* (cm<sup>2</sup>/g).

**opac\_nl** *0 ir igp*

**copac\_nl** *0 ir igp*

Opacity, not including contributions of photoexcitations, at node *ir* for group *igp* (cm<sup>2</sup>/g).

**opac\_hk** *iz ir igp*

Opacity from Henke tables for element *iz* at node *ir* for group *igp* (cm<sup>2</sup>/g). If *iz*=0, the opacity will be for the combination of elements at node *ir*. It is not necessary to specify Henke opacities for the material to edit this quantity.

**ckapb** *0 ir igp*

**ckappab** *0 ir igp*

Effective absorption coefficient at node *ir* for group *igp* (1/cm) obtained by dividing the emission coefficient by the Planck function.

**opacb** *0 ir igp*

Effective opacity at node *ir* for group *igp* (cm<sup>2</sup>/g) obtained by dividing the emission coefficient by the Planck function and the mass density.

**ckapbb** *iz ir igp [iso]*

**ckapbf** *iz ir igp [iso]*

**ckapbt** *iz ir igp [iso]*

**copacbb** *iz ir igp [iso]*

**copacbf** *iz ir igp [iso]*

**copactb** *iz ir igp [iso]*

Absorption coefficient or opacity at node *ir* for group *igp* due to bound-bound (**bb**) or bound-free (**bf**) or bound-bound plus bound-free (**bt**) transitions element *iz* (1/cm or cm<sup>2</sup>/g). If *iz*=0, the absorption is due to all elements. If *iz* is specified and *iso*≥0, the absorption is that due to transitions within isosequence *iso* (for bound-bound) or from isosequence *iso* to isosequence *iso-1* (for bound-free). If *iz* is specified and *iso*<0, the absorption is that due to all transitions within element *iz*.

## Export Controlled Information

**ckape** *0 ir igp*  
**ckappae** *0 ir igp*

Effective absorption coefficient at node *ir* for group *igp* (1/cm) obtained by dividing the emission coefficient by the Planck function.

**copace** *0 ir igp*  
**opace** *0 ir igp*

Effective opacity at node *ir* for group *igp* (cm<sup>2</sup>/g) obtained by dividing the emission coefficient by the Planck function and the mass density.

**ckapff** *iz ir igp [iso]*

Free-free absorption coefficient or opacity at node *ir* for group *igp* (1/cm or cm<sup>2</sup>/g). If *iz*=0, the coefficient is that due to free-free absorption from all elements. If *iz* is specified and *iso*≥0, the coefficient is that from isosequence *iso*. If *iz* is specified and *iso*<0, the coefficient is that from element *iz*.

**ckapj** *0 ir igp*  
**ckappaj** *0 ir igp*

Effective absorption coefficient at node *ir* for group *igp* (1/cm) obtained by dividing the emission coefficient by the radiation intensity.

**ckapjtot** *0 i*

Total absorbed radiative energy at node *ir* (erg/cm<sup>3</sup>/sec).

**copacj** *0 ir igp*

Effective opacity at node *ir* for group *igp* (cm<sup>2</sup>/g) obtained by dividing the emission coefficient by the radiation intensity and the mass density.

**copacjtot** *0 i*

Total absorbed radiative energy at node *ir* (erg/g/sec).

**emis** *0 ir igp*  
**cemis** *0 ir igp*

Emission coefficient at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster). The value reported includes all requested contributions and multipliers.

**emis\_nl** *0 ir igp*  
**cemis\_nl** *0 ir igp*

Emission coefficient, not including contributions of photoexcitations, at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster).

**dedit** *0 ir igp*

Derivative w.r.t. electron temperature of emission coefficient at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster/eV)

# Export Controlled Information

**dedn** *0 ir igp*

Derivative w.r.t. log(electron density) of emission coefficient at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster)

**cemisb** *0 ir igp*

Effective emission coefficient at node *ir* for group *igp* obtained by multiplying the absorption coefficient by the Planck function (erg/cm<sup>3</sup>/sec/Hz/ster).

**cemisbb** *iz ir igp iso*

**cemisbf** *iz ir igp iso*

**cemisbt** *iz ir igp iso*

Emission coefficient at node *ir* for group *igp* due to bound-bound (**bb**) or bound-free (**bf**) or bound-bound plus bound-free (**bt**) transitions at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster). If *iz*=0, the emission coefficient is due to all elements. If *iz* is specified and *iso*≥0, the emission coefficient is that due to transitions within isosequence *iso* (for bound-bound) or from isosequence *iso* to isosequence *iso*+1 (for bound-free) for element *iz*. If *iz* is specified and *iso*<0, the emission coefficient is that due to all transitions within element *iz*.

**cemisbbtot** *iz ir [iso]*

**cemisbftot** *iz ir [iso]*

**cemisbttot** *iz ir [iso]*

Integrated bound-bound (**bb**), bound-free (**bf**), or bound-bound plus bound-free (**bt**) emission coefficient at node *ir* (erg/cm<sup>3</sup>/sec). If *iz*=0, the emission coefficient is due to all elements. If *iz* is specified and *iso*≥0, the emission coefficient is that due to transitions in isosequence *iso* within isosequence *iso* (for bound-bound) or from isosequence *iso* to isosequence *iso*+1 (for bound-free) for element *iz*. If *iz* is specified and *iso*<0, the emission coefficient is that due to all transitions within element *iz*.

**cemisff** *iz ir igp [iso]*

Free-free emission coefficient at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec/Hz/ster). If *iz*=0, the emission coefficient is that due to free-free emission from all elements. If *iz* is specified and *iso*≥0, the emission coefficient is that from isosequence *iso*. If *iz* is specified and *iso*<0, the emission coefficient is that from element *iz*.

**cemisfftot** *iz ir igp [iso]*

Integrated free-free emission coefficient at node *ir* for group *igp* (erg/cm<sup>3</sup>/sec). If *iz*=0, the emission coefficient is that due to free-free emission from all elements. If *iz* is specified and *iso*≥0, the emission coefficient is that from isosequence *iso*. If *iz* is specified and *iso*<0, the emission coefficient is that from element *iz*.

**cemisj** *0 ir igp*

Effective emission coefficient at node *ir* for group *igp* obtained by multiplying the absorption coefficient by the radiation intensity (erg/cm<sup>3</sup>/sec/Hz/ster).

**cemisnet** *0 ir igp*

Net emission coefficient at node *ir* for group *igp* obtained by subtracting the absorption coefficient times by the radiation intensity from the emission coefficient (erg/cm<sup>3</sup>/sec/Hz/ster).

# Export Controlled Information

**cemisk** *0 ir igp*

Effective emission coefficient at node *ir* for group *igp* obtained by multiplying the absorption coefficient by the Planck function (erg/cm<sup>3</sup>/sec/Hz/ster).

**cemisnet** *0 ir*

Net integrated emission less absorption at node *ir* (erg/cm<sup>3</sup>/sec).

**cemistot** *0 ir*

Integrated emission coefficient at node *ir* (erg/cm<sup>3</sup>/sec). The value reported includes all requested contributions and multipliers.

**gff** *0 ir*

Free-free Gaunt factor at node *ir*.

**scat** *0 ir igp*

Scattering coefficient at node *ir* for group *igp* (1/cm).

**nescat** *0 ir*

Effective electron density for scattering at node *ir* (1/cm<sup>3</sup>).

**btot** *0*

**bratetot** *0*

Black-body radiation energy density appropriate to the electron temperature at node *ir* (erg/cm<sup>3</sup>).

**bnu** *0 ir igp*

**brate** *0 ir igp*

Black-body radiation intensity appropriate to the electron temperature at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**bnur** *0 ir igp*

Black-body radiation intensity appropriate to the radiation temperature (**tr**) at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**bnurad** *0 ir igp*

Black-body radiation intensity appropriate to the effective radiation temperature (**trad**) at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jbndry** *ix 0 igp*

Boundary radiation intensity from xfile *ix* for group *igp* (erg/cm<sup>2</sup>/sec/Hz).

**jbnd2** *ix 0 igp*

Boundary radiation intensity for first-flight radiation transport from xfile *ix* for group *igp* (erg/cm<sup>2</sup>/sec/Hz).

# Export Controlled Information

**jnu** 0 *ir igp*

Radiation intensity (angle-averaged) at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jdnu** 0 *ir igp*

Radiation intensity (angle-averaged) at node *ir* integrated over group *igp* (erg/cm<sup>2</sup>/sec/ster).

**jnu2** 0 *ir igp*

Radiation intensity (angle-averaged) from first-flight radiation transport at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jnul** 0 *ir igp*

Radiation intensity (angle-averaged) from lines at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jnutot** 0 *ir igp*

Radiation intensity (angle-averaged) from continuum and lines at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jrate** 0 *ir igp*

Radiation intensity used for atomic kinetics at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jratei** 0 *ir igp*

Radiation intensity used for photoionizations at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jratex** 0 *ir igp*

Radiation intensity used for photoexcitations at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jemis** 0 *ir igp*

Spontaneous emission radiation intensity used for emissivities at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jspon** 0 *ir igp*

Spontaneous emission radiation intensity used for transition rates at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

**jratio** 0 *ir igp*

Ratio of radiation intensity used for atomic kinetics to average radiation intensity at node *ir* for group *igp*.

**snu** 0 *ir igp*

**csrc** 0 *ir igp*

Source function obtained by dividing emission by absorption at node *ir* for group *igp* (erg/cm<sup>2</sup>/sec/Hz/ster).

# Export Controlled Information

**eradc** *0 ir*

Continuum radiation energy density at node *ir* (erg/cm<sup>3</sup>).

**eradl** *0 ir*

Line radiation energy density at node *ir* (erg/cm<sup>3</sup>).

**erad** *0 ir*

Total radiation energy density (from continuum and lines) at node *ir* (erg/cm<sup>3</sup>).

**erad2** *0 ir*

Total radiation energy density from first-flight radiation transport at node *ir* (erg/cm<sup>3</sup>).

**pradc** *0 ir*

Continuum radiation pressure at node *ir* (erg/cm<sup>3</sup>).

**pradl** *0 ir*

Line radiation pressure at node *ir* (erg/cm<sup>3</sup>).

**prad** *0 ir*

Total radiation pressure (from continuum and lines) at node *ir* (erg/cm<sup>3</sup>).

**trad** *0 ir*

Radiation temperature equivalent of **erad** at node *ir* (K).

**tradv** *0 ir*

Radiation temperature equivalent of **erad** at node *ir* (eV).

The following three sets of edits specify optical depths. In 1-d geometries, if *ir* is positive the extent is from node 1 to node *ir*, if *ir* is negative the extent is from node *-ir* to the last node, and if *ir*=0 the extent covers all nodes. In 2-d and 3-d geometries, the extent is to node *ir* along the appropriate k-line (for **tau\_k**) or l-line (for **tau\_l**) or m-line (for **tau\_m**).

**taukap** *0 ir igp*

**taukapk** *0 ir igp*

**taukapl** *0 ir igp*

**taukapm** *0 ir igp*

Absorption optical depth for group *igp*.

**tauscat** *0 ir igp*

**tauscatk** *0 ir igp*

**tauscatl** *0 ir igp*

**tauscatm** *0 ir igp*

Scattering optical depth for group *igp*.

## Export Controlled Information

**tautot** 0 *ir igp*  
**tautotk** 0 *ir igp*  
**tautotl** 0 *ir igp*  
**tautotm** 0 *ir igp*

Total optical depth for group *igp*.

**inu+** 0 *ir igp idir*  
**inu-** 0 *ir igp idir*

Specific intensity at node *ir* for group *igp* in direction *idir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec/Hz/ster).

**cflux+** 0 *ir igp*  
**cflux-** 0 *ir igp*

Flux at node *ir* for group *igp* in (+) or (-) direction (erg/cm<sup>2</sup>/sec/Hz) (for 1-d geometries only).

**cfluxx+** 0 *ir igp*  
**cfluxx-** 0 *ir igp*

Flux at node *ir* for group *igp* in (+) or (-) x-direction or r-direction (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

**cfluxy+** 0 *ir igp*  
**cfluxy-** 0 *ir igp*

Flux at node *ir* for group *igp* in (+) or (-) y-direction or z-direction (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

**eflux+** 0 *ir*  
**eflux-** 0 *ir*

Integrated flux at node *ir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec) (for 1-d geometries only).

**efluxx+** 0 *ir*  
**efluxx-** 0 *ir*

Integrated flux at node *ir* in (+) or (-) x-direction or r-direction (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**efluxy+** 0 *ir*  
**efluxy-** 0 *ir*

Integrated flux at node *ir* in (+) or (-) y-direction or z-direction (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**cdetect** 0 *idtct igp*

Flux for group *igp* seen by detector *idtct* (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

**edetect** 0 *idtct igp*

Integrated flux seen by detector *idtct* (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

# Export Controlled Information

**tauray** *0 ir igp idir*

In 1-d geometries, total optical depth for group *igp*, in direction *idir*. The extent is defined the same as for **tautot**.

In 2-d and 3-d geometries, total optical depth from node *ir* to boundary for group *igp* in direction *idir*.

**jnuarea** *0 ir igp*

For spherical geometry, the area-integrated intensity for group *igp* emergent from a circular mask centered on  $r=0$  and extending to  $r=r(ir)$  (erg/sec/Hz/ster).

For cylindrical geometry, the area-integrated intensity for group *igp* emergent from a mask of length param(84) centered on  $r=0$  and extending to  $r=r(ir)$ , viewed from an angle  $\theta$  with respect to the z-axis given by  $\theta = \cos^{-1}(\text{param}(83))$  (erg/sec/Hz/ster).

**jnuline** *0 ir igp*

For spherical geometry, the integrated intensity for group *igp* emergent from a linear mask centered on  $r=0$  and extending to  $r=r(ir)$  (erg/cm/sec/Hz/ster).

## Spectral radiation:

- (+) signifies the direction of increasing *ir* in 1-d geometries
- (-) signifies the direction of decreasing *ir* in 1-d geometries
- (+,-) are equivalent in 2-d geometries

Two indices are required to specify a particular energy. The spectral range index *is* specifies the energy range, corresponding to a **spectral-group** command. The energy index *isp* specifies the energy within that range. The value *is*=0 corresponds to the full spectrum. For those edits requiring an element index *iz*, the spectral index *is* is obtained from a spectral x-variable if present, or is assumed to be 0.

**isp** *is 0 isp*

Spectral group index within the range specified by the index *is*.

**nsp** *is*

Number of spectral groups within the range specified by the index *is*.

**sp\_energy** *is 0 isp*

Energy for spectral group (*is,isp*) (eV).

**sp\_freq** *is 0 isp*

Frequency for spectral group (*is,isp*) (Hz).

**sp\_nu** *is 0 isp*

Frequency for spectral group (*is,isp*) (Hz).

**sp\_wvl** *is 0 isp*

# Export Controlled Information

Wavelength for spectral group ( $is,isp$ ) (cm).

**emis\_sp** *is ir isp*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) without doppler shifts (erg/cm<sup>3</sup>/sec/Hz/ster).

**emis\_sp+** *is ir isp idir*

**emis\_sp-** *is ir isp idir*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) in direction  $idir$  in (+) or (-) direction (erg/cm<sup>3</sup>/sec/Hz/ster).

**ebb\_sp** *is ir isp*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to bound-bound transitions (erg/cm<sup>3</sup>/sec/Hz/ster).

**ebf\_sp** *is ir isp*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to bound-free transitions (erg/cm<sup>3</sup>/sec/Hz/ster).

**ebt\_sp** *is ir isp*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to bound-bound and bound-free transitions (erg/cm<sup>3</sup>/sec/Hz/ster).

**eff\_sp** *is ir isp*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to free-free emission (erg/cm<sup>3</sup>/sec/Hz/ster).

**ebbi\_sp** *iz ir isp iso*

**ebfi\_sp** *iz ir isp iso*

**ebti\_sp** *iz ir isp iso*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to bound-bound (**bb**) or bound-free (**bf**) or bound-bound plus bound-free (**bt**) transitions of element  $iz$  (erg/cm<sup>3</sup>/sec/Hz/ster). If  $iso \geq 0$  is specified, the emission coefficient is that due to transitions within isosequence  $iso$  (for bound-bound) or from isosequence  $iso$  to isosequence  $iso+1$  (for bound-free). If  $iso < 0$  is specified, the emission coefficient is that due to all transitions within element  $iz$ .

**effi\_sp** *iz ir isp iso*

Emission coefficient at node  $ir$  for spectral group ( $is,isp$ ) due to free-free emission from element  $iz$  (erg/cm<sup>3</sup>/sec/Hz/ster). If  $iso \geq 0$  is specified, the emission coefficient is that due to isosequence  $iso$ . If  $iso < 0$  is specified, the emission coefficient is that due to element  $iz$ .

**kappa\_sp** *is ir isp*

Absorption coefficient at node  $ir$  for spectral group ( $is,isp$ ) without doppler shifts (1/cm).

**kappa\_sp+** *is ir isp idir*

**kappa\_sp-** *is ir isp idir*

# Export Controlled Information

Absorption coefficient at node  $ir$  for spectral group  $(is,isp)$  in direction  $idir$  in (+) or (-) direction (1/cm).

**kappa\_sp\_hk**  $iz\ ir\ igr$

Absorption coefficient from Henke tables for element  $iz$  at node  $ir$  for spectral group  $isp$  (1/cm). If  $iz=0$ , the absorption coefficient will be for the combination of elements at node  $ir$ . It is not necessary to specify Henke opacities for the material to edit this quantity.

**opac\_sp**  $is\ ir\ isp$

Opacity at node  $ir$  for spectral group  $(is,isp)$  without doppler shifts (cm<sup>2</sup>/g).

**opac\_sp+**  $is\ ir\ isp\ idir$

**opac\_sp-**  $is\ ir\ isp\ idir$

Opacity at node  $ir$  for spectral group  $(is,isp)$  in direction  $idir$  in (+) or (-) direction (cm<sup>2</sup>/g).

**opac\_sp\_hk**  $iz\ ir\ igr$

Opacity from Henke tables for element  $iz$  at node  $ir$  for spectral group  $isp$  (cm<sup>2</sup>/g). If  $iz=0$ , the absorption coefficient will be for the combination of elements at node  $ir$ . It is not necessary to specify Henke opacities for the material to edit this quantity.

**kbb\_sp**  $is\ ir\ isp$

**opbb\_sp**  $is\ ir\ isp$

Absorption coefficient or opacity at node  $ir$  for spectral group  $(is,isp)$  due to bound-bound transitions (1/cm or cm<sup>2</sup>/g).

**kbf\_sp**  $is\ ir\ isp$

**opbf\_sp**  $is\ ir\ isp$

Absorption coefficient or opacity at node  $ir$  for spectral group  $(is,isp)$  due to bound-free transitions (1/cm or cm<sup>2</sup>/g).

**kbt\_sp**  $is\ ir\ isp$

**opbt\_sp**  $is\ ir\ isp$

Absorption coefficient or opacity at node  $ir$  for spectral group  $(is,isp)$  due to bound-bound and bound-free transitions (1/cm or cm<sup>2</sup>/g).

**kff\_sp**  $is\ ir\ isp$

**opff\_sp**  $is\ ir\ isp$

Absorption coefficient or opacity at node  $ir$  for spectral group  $(is,isp)$  due to free-free absorption (1/cm or cm<sup>2</sup>/g).

**kbhi\_sp**  $iz\ ir\ isp\ iso$

**kbfi\_sp**  $iz\ ir\ isp\ iso$

**kbtii\_sp**  $iz\ ir\ isp\ iso$

**opbbi\_sp**  $iz\ ir\ isp\ iso$

**opbfi\_sp**  $iz\ ir\ isp\ iso$

**opbti\_sp**  $iz\ ir\ isp\ iso$

## Export Controlled Information

Absorption coefficient or opacity at node *ir* for spectral group (*is,isp*) due to bound-bound (**bb**) or bound-free (**bf**) or bound-bound plus bound-free (**bt**) transitions element *iz* (1/cm or cm<sup>2</sup>/g). If *iso*≥0 is specified, the absorption is that due to transitions within isosequence *iso* (for bound-bound) or from isosequence *iso* to isosequence *iso-1* (for bound-free). If *iso*<0 is specified, the coefficient is that due to free-free absorption from element *iz*.

**kffi\_sp** *iz ir isp iso*  
**opffi\_sp** *iz ir isp iso*

Absorption coefficient or opacity for element *iz* at node *ir* for spectral group (*is,isp*) due to free-free absorption (1/cm or cm<sup>2</sup>/g). If *iso*≥0 is specified, the coefficient is that due to free-free absorption from isosequence *iso*. If *iso*<0 is specified, the coefficient is that due to free-free absorption from element *iz*.

**src\_sp** *is ir isp*

Source function at node *ir* for spectral group (*is,isp*) (erg/cm<sup>2</sup>/sec/Hz/ster).

**jsp** *is ir isp*

Spectral radiation intensity at node *ir* for spectral group (*is,isp*) (erg/cm<sup>2</sup>/sec/Hz/ster).

**tr\_sp** *0 ir isp*

Equivalent temperature at node *ir* for spectral radiation (eV).

**isp+ is ir isp idir**  
**isp- is ir isp idir**

Spectral specific intensity at node *ir* for spectral group (*is,isp*) in direction *idir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec/Hz/ster).

**bnu\_sp** *is ir isp*

Black-body radiation intensity appropriate to the electron temperature at node *ir* for spectral group (*is,isp*) (erg/cm<sup>2</sup>/sec/Hz/ster).

**spflux+ is ir isp**  
**spflux- is ir isp**

Flux at node *ir* for spectral group (*is,isp*) in (+) or (-) direction (erg/cm<sup>2</sup>/sec/Hz) (for 1-d geometries only).

**spfluxx+ is ir isp**  
**spfluxx- is ir isp**

Flux at node *ir* for spectral group (*is,isp*) in (+) or (-) x-direction or r-direction (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

**spfluxy+ is ir isp**  
**spfluxy- is ir isp**

Flux at node *ir* for spectral group (*is,isp*) in (+) or (-) y-direction or z-direction (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

## Export Controlled Information

**speflux+ is ir**  
**speflux- is ir**

Integrated flux at node *ir* in (+) or (-) direction for spectral range *is* (erg/cm<sup>2</sup>/sec) (for 1-d geometries only).

**spefluxx+ is ir**  
**spefluxx- is ir**

Integrated flux at node *ir* in (+) or (-) x-direction or r-direction for spectral range *is* (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**spefluxy+ is ir**  
**spefluxy- is ir**

Integrated flux at node *ir* in (+) or (-) y-direction or z-direction for spectral range *is* (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**spdetect is idtct isp**

Flux for spectral group *isp* seen by detector *idtct* (erg/cm<sup>2</sup>/sec/Hz) (for 2-d geometries only).

**tausp is ir isp**  
**tauspk is ir isp**  
**tauspl is ir isp**  
**tauspm is ir isp**

Optical depth for spectral group (*is,isp*). In 1-d geometries, if *ir* is positive the extent is from node 1 to node *ir*, if *ir* is negative the extent is from node *-ir* to the last node, and if *ir*=0 the extent covers all nodes. In 2-d and 3-d geometries, the extent is to node *ir* along the appropriate k-line (for **tauspk**) or l-line (for **tauspl**) or m-line (for **tauspm**).

**tausp+ is ir isp idir**  
**tausp- is ir isp idir**

In 1-d geometries, the optical depth for spectral group (*is,isp*) in direction *idir* in (+) or (-) direction. The extent is defined the same as for **tausp**.

In 2-d and 3-d geometries, optical depth from node *ir* to boundary for spectral group (*is,isp*) in direction *idir*.

**jsparea is ir isp**

For spherical geometry, the area-integrated intensity for spectral group (*is,isp*) emergent from a circular mask centered on *r*=0 and extending to *r*=*r(ir)* (erg/sec/Hz/ster).

For cylindrical geometry, the area-integrated intensity for spectral group (*is,isp*) emergent from a mask of length param(84) centered on *r*=0 and extending to *r*=*r(ir)*, viewed from an angle  $\theta$  with respect to the z-axis given by  $\theta = \cos^{-1}(\text{param}(83))$  (erg/sec/Hz/ster).

**jspline is ir isp**

For spherical geometry, the integrated intensity for spectral group (*is,isp*) emergent from a linear mask centered on *r*=0 and extending to *r*=*r(ir)* (erg/cm/sec/Hz/ster).

# Export Controlled Information

**isp\_0d** *is ir isp*

Spectral specific intensity from node *ir* for spectral group (*is,isp*) attenuated by material of depth specified by param(53) (erg/cm<sup>2</sup>/sec/Hz/ster).

**esp\_0d** *is ir isp*

Spatially-averaged spectral emission coefficient from node *ir* for spectral group (*is,isp*) attenuated by material of depth specified by param(53) (erg/cm<sup>3</sup>/sec/Hz/ster).

## Line radiation:

- (+) signifies the direction of increasing ir in 1-d geometries
- (-) signifies the direction of decreasing ir in 1-d geometries
- (+,-) are equivalent in 2-d geometries

**avoigt** *iline ir*

Voigt parameter for line *iline* at node *ir*.

**uvoigt** *iline ir*

Maximum value of Voigt function for line *iline* at node *ir*.

**dopwidth** *iline ir*

Doppler width of line *iline* at node *ir* (eV).

**deline** *iline ir*

Transition energy of line *iline* (eV).

**deline** *iline ir*

Energy offset of line *iline* from central line in resonant group (eV).

**eline** *iline ir*

Energy density in radiation of line *iline* at node *ir* (erg/cm<sup>3</sup>).

**jmode** *iline*

Conversion factor from (photons/mode) to (erg/cm<sup>2</sup>/sec/Hz/ster) for line *iline*.

**jbar** *iline ir*

Line strength of line *iline* at node *ir* (photons/mode).

**jbaregs** *iline ir*

Line strength of line *iline* at node *ir* (erg/cm<sup>2</sup>/sec/Hz).

# Export Controlled Information

**jbarx** *iline ir*

Intermediate value of line strength of line *iline* at node *ir* (photons/mode). This quantity may only be edited as a snapshot over the entire mesh and one edit will appear for each iteration during a timestep.

**lineiter** *iline*

Number of iterations taken by the approximate operator formalism for line *iline* during a timestep. This quantity may only be edited vs. time.

**jsat** *iline ir*

Saturation intensity for line *iline* at node *ir* (photons/mode).

**jsatcgs** *iline ir*

Saturation intensity for line *iline* at node *ir* (erg/cm<sup>2</sup>/sec/Hz).

**nrb** *iline ir*

Net radiative bracket for line *iline* at node *ir*.

**sigma** *iline ir*

Integrated absorption cross-section for line *iline* at node *ir* (cm<sup>2</sup>).

**ylinel** *iline ir*

Effective lower level population for line *iline* at node *ir*, i.e.  $y_l - (g_l/g_u)y_u$  (1/cm<sup>3</sup>).

**ylineu** *iline ir*

Effective upper level population for line *iline* at node *ir*, i.e.  $(g_l/g_u)y_u$  (1/cm<sup>3</sup>).

**dyldj** *iline ir*

Derivative of logarithm of effective lower level population w.r.t. jbar for line *iline* at node *ir* (modes/photon).

**dyudj** *iline ir*

Derivative of logarithm of effective upper level population w.r.t. jbar for line *iline* at node *ir* (modes/photon).

**ylinec** *iline ir*

Areal density of lower level population of line *iline* from node 1 to node *ir* (1/cm<sup>2</sup>).

**prdfrac** *iline ir*

Fraction of emissivity which is redistributed for prd line *iline* at node *ir*.

# Export Controlled Information

**ifrline** *iline 0 ifr*

Line group index for line *iline*.

**evline** *iline 0 ifr*

Energy of bin *ifr* in line *iline* w.r.t. center of line *iline* (eV).

**nuline** *iline 0 ifr*

Frequency of bin *ifr* in line *iline* w.r.t. center of line *iline* (Hz).

**jline** *iline ir ifr*

Average intensity (in fluid frame) in line *iline* at node *ir* for bin *ifr* (photons/mode).

**jline0** *iline ir ifr*

Average intensity (in lab frame) in line *iline* at node *ir* for bin *ifr* (photons/mode).

**jlinecgs** *iline ir ifr*

Average intensity (in fluid frame) in line *iline* at node *ir* for bin *ifr* (erg/cm<sup>2</sup>/sec/Hz).

**jline0cgs** *iline ir ifr*

Average intensity (in lab frame) in line *iline* at node *ir* for bin *ifr* (erg/cm<sup>2</sup>/sec/Hz).

**lemis** *iline ir ifr*

Emission coefficient for line *iline* at node *ir* for bin *ifr* (photons/mode/cm).

**lemiscgs** *iline ir ifr*

Emission coefficient for line *iline* at node *ir* for bin *ifr* (erg/cm<sup>3</sup>/sec/Hz/ster).

**lemiss** *iline ir ifr*

**lemistot** *iline ir ifr*

Total emission coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* (photons/mode/cm).

**lemistcgs** *iline ir ifr*

Total emission coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* (erg/cm<sup>3</sup>/sec/Hz/ster).

**lkap** *iline ir ifr*

Absorption coefficient for line *iline* at node *ir* for bin *ifr* (1/cm).

**lkapt** *iline ir ifr*

**lkaptot** *iline ir ifr*

Total absorption coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* (1/cm).

# Export Controlled Information

**srcline** *iline ir ifr*

Source function in line *iline* at node *ir* for bin *ifr* (photons/mode).

**srlinects** *iline ir ifr*

Source function in line *iline* at node *ir* for bin *ifr* (erg/cm<sup>2</sup>/sec/Hz/ster).

**tauline** *iline ir ifr*

**taulinek** *iline ir ifr*

**taulinel** *iline ir ifr*

**taulinem** *iline ir ifr*

Optical depth in line *iline* for bin *ifr*. In 1-d geometries, if *ir* is positive the extent is from node 1 to node *ir*, if *ir* is negative the extent is from node *-ir* to the last node, and if *ir*=0 the extent covers all nodes. In 2-d and 3-d geometries, the extent is to node *ir* along the appropriate k-line (for **taulinek**) or l-line (for **taulinel**) or m-line (for **taulinem**).

**tauline+** *iline ir ifr idir*

**tauline-** *iline ir ifr idir*

In 1-d geometries, optical depth in direction *idir* in (+) or (-) direction. The extent is defined the same as for **tauline**.

In 2-d and 3-d geometries, optical depth from node *ir* to boundary for bin *ifr* in direction *idir*.

**iline+** *iline ir ifr idir*

**iline-** *iline ir ifr idir*

Specific intensity in line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (photons/mode).

**lline+** *iline ir 0 idir*

**lline-** *iline ir 0 idir*

Integrated intensity in line *iline* at node *ir* in direction *idir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec/ster).

**iflux+** *iline ir ifr*

**iflux-** *iline ir ifr*

Flux in line *iline* at node *ir* for bin *ifr* in (+) or (-) direction (photons/mode) (for 1-d geometries only).

**iflux+** *iline ir*

**iflux-** *iline ir*

Integrated flux in line *iline* at node *ir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec) (for 1-d geometries only).

**ifluxx+** *iline ir ifr*

**ifluxx-** *iline ir ifr*

Flux in line *iline* at node *ir* for bin *ifr* in (+) or (-) x-direction or r-direction (photons/mode) (for 2-d geometries only).

**ifluxy+** *iline ir ifr*

**ifluxy-** *iline ir ifr*

# Export Controlled Information

Flux in line *iline* at node *ir* for bin *ifr* in (+) or (-) y-direction or z-direction (photons/mode) (for 2-d geometries only).

**lfluxx+** *iline ir*  
**lfluxx-** *iline ir*

Integrated flux in line *iline* at node *ir* in (+) or (-) x-direction or r-direction (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**lfluxy+** *iline ir*  
**lfluxy-** *iline ir*

Integrated flux in line *iline* at node *ir* in (+) or (-) y-direction or z-direction (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**idetect** *iline idtct ifr*

Flux in line *iline* for bin *ifr* seen by detector *idtct* (photons/mode) (for 2-d geometries only).

**ldetect** *iline idtct*

Integrated flux in line *iline* seen by detector *idtct* (erg/cm<sup>2</sup>/sec) (for 2-d geometries only).

**lemis+** *iline ir ifr idiry*  
**lemis-** *iline ir ifr idir*

Emission coefficient for line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (photons/mode/cm).

**lemiscgs+** *iline ir ifr idiry*  
**lemiscgs-** *iline ir ifr idir*

Emission coefficient for line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (erg/cm<sup>3</sup>/sec/Hz/ster).

**lemist+** *iline ir ifr idir*  
**lemist-** *iline ir ifr idir*

Total emission coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (photons/mode/cm)

**lemistcgs+** *iline ir ifr idir*  
**lemistcgs-** *iline ir ifr idir*

Total emission coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (erg/cm<sup>3</sup>/sec/Hz/ster)

**lkap+** *iline ir ifr idir*  
**lkap-** *iline ir ifr idir*

Absorption coefficient for line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (1/cm).

**lkapt+** *iline ir ifr idir*

# Export Controlled Information

## **Ikapt-** *iline ir ifr idir*

Total absorption coefficient (lines + continuum) in line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (1/cm).

## **srlcline+** *iline ir ifr idir* **srlcline-** *iline ir ifr idir*

Source function for line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (photons/mode).

## **srlclinecgs+** *iline ir ifr idir* **srlclinecgs-** *iline ir ifr idir*

Source function for line *iline* at node *ir* for bin *ifr* in direction *idir* in (+) or (-) direction (erg/cm<sup>2</sup>/sec/Hz/ster).

## **Inuarea** *iline ir ifr*

For spherical geometry, the area-integrated intensity for bin *ifr* of line *iline* emergent from a circular mask centered on r=0 and extending to r=r(*ir*) (erg/sec/Hz/ster).

For cylindrical geometry, the area-integrated intensity for bin *ifr* of line *iline* emergent from a mask of length param(84) centered on r=0 and extending to r=r(*ir*), viewed from an angle  $\theta$  with respect to the z-axis given by  $\theta = \cos^{-1}(\text{param}(83))$  (erg/sec/Hz/ster).

## **Inuline** *iline ir ifr*

For spherical geometry, the integrated intensity for bin *ifr* of line *iline* emergent from a linear mask centered on r=0 and extending to r=r(*ir*) (erg/cm/sec/Hz/ster).

## **Itotarea** *iline ir*

Frequency-integrated version of **Inuarea** (erg/sec/ster).

## **Itotline** *iline ir*

Frequency-integrated version of **Inuline** (erg/cm/sec/ster).

## Hydrodynamics:

### **cspd** *0 ir*

Sound speed in zone *ir* (cm/sec).

### **pmass** *0 ir*

Nodal mass associated with node *ir* (g).

# Export Controlled Information

**qsflg** *0 ir*

Flag denoting whether a quiet start is (=1) or is not (=0) affecting zone *ir*.

**qsd** *0 ir*

Reference density for quiet start in zone *ir* (g/cm<sup>3</sup>).

**qst** *0 ir*

Reference temperature for quiet start in zone *ir* (eV).

**zmass** *0 ir*

Mass in zone *ir* (g).

**zrho** *0 ir*

Mass density in zone *ir* (g/cm<sup>3</sup>).

**zp** *0 ir*

Pressure in zone *ir* (erg/cm<sup>3</sup>).

**zpv** *0 ir*

Pressure times volume in zone *ir* (erg).

**zq** *0 ir*

Artificial viscosity in zone *ir* (erg/cm<sup>3</sup>).

**zqv** *0 ir*

Artificial viscosity times volume in zone *ir* (erg).

**ze** *0 ir*

Internal energy in zone *ir* (erg).

**zekin** *0 ir*

Kinetics energy associated with node *ir* (erg).

**zetot**

Total internal energy summed over all zones (erg).

**zktot**

Total kinetic energy summed over all nodes (erg).

# Export Controlled Information

## zektot

Total internal + kinetic energy summed over all nodes / zones (erg).

## zgam *0 ir*

Effective gamma for EOS in zone *ir*.

## zv *0 ir*

Volume of zone *ir* (cm<sup>3</sup>).

## Laser Raytrace:

### **necrit** *ilas*

Critical electron density for laser *ilas* (cm<sup>-3</sup>).

### **plaser+** *ilas ir 0 ilray*

Laser power for laser *ilas* at node *ir* along ray *ilray* in the direction of increasing r (erg/sec).

### **plaser-** *ilas ir 0 ilray*

Laser power for laser *ilas* at node *ir* along ray *ilray* in the direction of decreasing r (erg/sec).

## Convergence:

### **ntry**

Number of iterations for current timestep.

### **ntryz** *0 ir*

Number of kinetics iterations for node *ir* for current timestep.

### **diffy** *iz ir iso i*

Fractional change in population of state (*iso,i*) of element *iz* at node *ir* for current timestep.

### **diffy1** *iz ir iso*

Fractional change in population of isoelectronic sequence *iso* of element *iz* at node *ir* for current timestep.

### **diffte** *0 ir*

Fractional change in electron temperature at node *ir* for current timestep.

### **diffti** *0 ir*

Fractional change in ion temperature at node *ir* for current timestep.

# Export Controlled Information

## **diffj** *iline ir*

Fractional change in line strength (jbar) for line *iline* at node *ir* for current timestep.

## **diffymax**

Maximum fractional change in isosequence populations for current timestep.

## **diffmax**

Maximum fractional change in electron or ion temperature for current timestep.

## **diffjmax**

Maximum fractional change in line strengths for current timestep.

## **ndiff**

Zone with maximum fractional change in electron or ion temperature for current timestep.

## **ndiffy**

Zone with maximum fractional change in isosequence populations for current timestep.

## **ndiff\_iz**

Model index with maximum fractional change in isosequence populations for current timestep.

## **ndiff\_iso**

Isosequence with maximum fractional change in isosequence populations for current timestep.

## **ndiffj**

Zone with maximum fractional change in line strengths for current timestep.

## **ndiff\_line**

Line index with maximum fractional change in line strengths for current timestep.

## **iter\_temp**

Number of iterations for 2d/3d thermal conduction linear solver.

## **err\_temp**

Residual magnitude for 2d/3d thermal conduction linear solver.

## Sizes, memory usage, cpu time:

## **nr**

# Export Controlled Information

**nt**

Number of computational points.

**nf**

**ng**

Number of continuum frequency groups.

**nyz**

Total number of stored populations.

**nyvz**

Total number of stored virtual populations.

**ny1z**

Total number of isosequence populations.

**myz**

Allocated number of stored populations.

**myvz**

Allocated number of stored virtual populations.

**my1z**

Allocated number of isosequence populations.

**nxtndy**

Number of times allocated space for populations has been increased.

**nxtndyv**

Number of times allocated space for virtual populations has been increased.

**nxtndy1**

Number of times allocated space for isosequence populations has been increased.

**cputime**

Cputime used by the current computational cycle (sec).

**cputot**

Total cputime used up through the current computational cycle (sec).

## Export Controlled Information

### **cpukin *iz ir***

Cputime used by kinetics calculations for element *iz* of zone *ir* (sec). If *iz*=0, this includes all elements.

### **cpukin *iline***

Cputime used by line transfer calculations for line *iline* (sec).

### **memory**

Memory currently allocated (bytes).

### **memmax**

Maximum memory allocated up through the current computational cycle (bytes).

# Export Controlled Information

## Sample Generator Files

### 0-d kinetics

```
c **** 0-d kinetics/temperature evolution ****
c -----
c   Aliases
c -----
alias Te    1000.          ! initial temperature
alias N_fe  1.e20           ! Fe number density
alias N_c   3.e21           ! C   number density
c -----
c   Materials
c -----
atoms dn106ha.dat c
atoms dnl26h1.dat fe

region 1 1 Te
element 2 N_fe 20 26
element 1 N_c

c -----
c   Sources
c -----
xfile x0dty.dat

xmult 1 pbins 0.1          ! decrease intensities from xfile
c -----
c   Controls
c -----
tstart 0.
tquit 1.0e-8

restart

c -----
c   Switches and Parameters
c -----
switch 11 1                ! make .plt file
switch 25 1                ! time-dependent
switch 29 1                ! use fixed timesteps
switch 30 10               ! dump every n timesteps
switch 31 1                ! do temperature calculation
switch 44 3                 ! maximum # of iterations

param 41 1.e-10            ! initial timestep
param 44 2.e-11            ! minimum timestep
param 45 2.e-10            ! maximum timestep
param 48 0.1               ! electron-ion coupling multiplier
```

## Export Controlled Information

```
param 61 1.e-4           ! iso-sequence population threshold

c -----
c   Edits
c -----

plot "Y1FRAC vs ISO (Fe)"
  xvar iso    2
  yvar ylfrac 2 1

plot "TEV, TIV vs TIME"
  xvar time
  yvar tev    0 1
  yvar tiv    0 1

plot "NE vs TIME"
  xvar time
  yvar ne     0 1

plot "Y1FRAC (Fe) vs TIME"
  xvar time
  yvar ylfrac 2 1 26:0:-1

plot "ISOMIN, ISOMAX (Fe) vs TIME"
  xvar time
  yvar isomin 2 1
  yvar isomax 2 1

plot "HEATx vs TIME"
  xvar time
  yvar heatct 2 1
  yvar heatpt 2 1
  yvar heatat 2 1
  yvar heatbt 0 1
  yvar heatet 2 1
  yvar heatjt 0 1

plot "DHEATx vs TIME"
  xvar time
  yvar dhtcdt 2 1
  yvar dhtpdt 2 1
  yvar dhtadt 2 1
  yvar dhtbdt 0 1
  yvar dhtedt 2 1
  yvar dhtjdt 0 1

plot "EINT, DEINTDT vs TIME"
  xvar time
  yvar eint    1:2 1
  yvar deintdt 1:2 1

plot "EIGENV_R (IRON) vs TIME"
  xvar time
  yvar eigenv_r 2 1 0 1:5

plot "NTRY vs TIME"
  xvar time
  yvar ntry
```

# Export Controlled Information

## 1-d radiation transfer

```
c      **** 1D: 2-region angular scattering ****
c ... free-free absorption dominates at low frequencies
c ... scattering dominates at high frequencies
c ... region 1 is optically thin to scattering
c ... region 2 is optically thick to scattering
c -----
c   Aliases
c -----
alias Te    1.e4
alias Rho1 1.e-4
alias Rho2 10.

alias EMIN 1.
alias EMAX 1.e5

alias NG    6
alias N1    1
alias N2    11
alias N3    N2+
alias N4    22

alias N  N4
alias DN  N4-
c -----
c   Materials
c -----
region N1 N2  Te
  material Rho1 1. 1. 1.
  regmult absn 1.e5

region N3 N4  Te
  material Rho2 1. 1. 1.

c -----
c   Geometry
c -----
geometry slab
rlin N1 N2 0. 1.
rlin N3 N4 1. 2.

c -----
c   Sources and Boundary Conditions
c -----
source jbndry 1 0. EMAX value constant 1.e10 1.
```

# Export Controlled Information

```
boundary radiation streaming -1. 0. N1 xfile 1 1.

c -----
c   Radiation
c -----

ebins NG  EMIN EMAX

angles 5

c -----
c   Controls
c -----

tstart 0.
tquit  2.0e-10

restart

c -----
c   Switches and Parameters
c -----

switch 11 1          ! make .plt file
switch 29 1          ! use constant timesteps
switch 30 10         ! dump every n timesteps
switch 36 2          ! do continuum transfer (integral formalism)
switch 39 2          ! # iterations for angular scattering

param  41 1.e-10      ! timestep

c -----
c   Edits
c -----


alias 4PI  4. * PI
alias IG   4

editray 1  0.05  0.  0
editray 2  0.23  0.  0
editray 3  -.50  0.  0
editray 4  0.77  0.  0
editray 5  0.95  0.  0
editray 6  1.00  0.  1
editray 7  0.01  0.  1
editray 8  0.33  0.  0
editray 9  0.67  0.  0

plot "CKAP, SCAT vs ENERGY"
xvar  energy
yvar  ckap    0  1:N:DN
yvar  scat    0  1:N:DN

plot "TAU (region 1) vs ENERGY"
xvar  energy
yvar  taukap  0  N2
yvar  tauscat 0  N2
```

## Export Controlled Information

```
yvar tautot 0 N2

plot "TAU (region 2) vs ENERGY"
xvar energy
yvar taukap 0 -N3
yvar tauscat 0 -N3
yvar tautot 0 -N3

plot "NION, NE vs R"
xvar r
yvar nion
yvar ne

plot "JNU vs R"
xvar r
yvar jnu 0 0 1:NG 0 4PI

plot "JNU vs R, TIME"
xvar r
xvar time
yvar jnu 0 0 1:NG 0 4PI

plot "JNU vs TIME, R"
xvar time
xvar r
yvar jnu 0 0 1:NG 0 4PI

plot "JBNDRY vs ENERGY"
xvar energy
yvar jbndry 1 0 0 0 4PI

plot "JNU vs ENERGY"
xvar energy 0 0
yvar jnu 0 N1 0 0 4PI
yvar jnu 0 N2 0 0 4PI
yvar jnu 0 N3 0 0 4PI

plot "CFLUX+,- vs R"
xvar r
yvar cflux+ 0 0 1:NG
yvar cflux- 0 0 1:NG

plot "INU+ vs R"
xvar r
yvar inu+ 0 0 IG 1:9

plot "INU- vs R"
xvar r
yvar inu- 0 0 IG 1:9

plot "TAURAY vs ENERGY"
xvar energy
yvar tauray 0 0 0 1:3
```

# Export Controlled Information

## 2-d line transfer

```
c           **** overlapping line transfer / prd ****
c   "An Ne-Like Fe Laser Resonantly Photo-Pumped by Ne X Ly-Alpha Radiation"
c   J. Nilsen, JQSRT 46, 547-556 (1991)
c -----
c   Aliases
c -----
alias K0      1
alias K1      8
alias K2     17
alias K3     25

alias L0      1
alias L1      8
alias L2     17
alias L3     25

alias X0    0.00
alias X1    0.01
alias X2    0.03
alias X3    0.05

alias Y0    0.00
alias Y1    0.01
alias Y2    0.03
alias Y3    0.05

alias Te1    95.
alias Te2   271.

alias N_Fe  1.e18
alias N_Ne  1.e19

c ... node numbers for edits

alias n1    K0
alias n2    K1
alias n3  K3 * L1- + K1

alias n4    K1+
alias n5    K2
alias n6    K3
alias n7  K3 * L1  + K1+
alias n8  K3 * L2- + K2
alias n9  K3 * L3

c -----
c   Materials
c -----
atoms d12lu1.dat ne
atoms d17lu1.dat fe

regionkl ireg=2 K0 K3 L0 L3
regionkl ireg=1 K0 K1 L0 L1
```

# Export Controlled Information

```
reg 1 Te1
  level 2 10 1 N_Fe

reg 2 Te2
  level 1 1 1 N_Ne

c -----
c   Geometry
c -----

geometry xy

xlog K0 K1 X0 X1 0.80
xlog K1 K2 X1 X2 1.25
xlog K2 K3 X2 X3 0.80

ylog L0 L1 Y0 Y1 0.80
ylog L1 L2 Y1 Y2 1.25
ylog L2 L3 Y2 Y3 0.80

product_mesh

c -----
c   Radiation
c -----
```

```
angles 3

ebins 11 50. 8.e4

linedefault prd

line 1 1 1 1 1 2
  lbins 25 5. 1.05

line 2 2 10 1 10 2
  lbins 25 1. 1.02

line 3 2 10 1 10 6
  lbins 25 5. 1.05

resonance 1 3 0.2

r2file pumpr2a generate

aprd 0.000 0.004
aprd 0.008 0.010
aprd 0.020 0.022
aprd 0.038 0.040
```

```
c -----
c   Controls
c -----

tstart 0.
tquit 10.e-09

restart
```

# Export Controlled Information

```
c -----
c   Switches and Parameters
c -----

switch 11 1           ! make .plt file
switch 25 0           ! steady-state
switch 28 1           ! steady-state initialization
switch 29 1           ! use constant timesteps
switch 30 10          ! dump every n timesteps
switch 34 11          ! symmetric about k=1 & l=1
switch 35 1           ! don't do bremsstrahlung
switch 36 0           ! don't do continuum transfer
switch 37 1           ! do line transfer
switch 38 1           ! use symmetric line profiles
switch 44 2           ! iterate timesteps for convergence

param 41 1.e-9         ! timestep

c -----
c   Edits
c -----

transition 1 2 10 1 10 2
transition 2 2 10 1 10 6
transition 3 2 10 2 10 3
transition 4 2 10 2 10 4
transition 5 2 10 2 10 5

c ... snapshots

plot "GAIN vs R"
  xvar r
  yvar gain 1:5

plot "JBAR vs R"
  xvar r
  yvar jbar 1:3

plot "JBAR (line 1) vs KX"
  xvar kx 0 L0
  yvar jbar 1 L0
  yvar jbar 1 L1+
  yvar jbar 1 L2
  yvar jbar 1 L3

plot "JBAR (line 2) vs KX"
  xvar kx 0 L0
  yvar jbar 2 L0
  yvar jbar 2 L1

plot "JBAR (line 3) vs KX"
  xvar kx 0 L0
  yvar jbar 3 L0
  yvar jbar 3 L1

plot "JBAR (line 1) vs LY"
  xvar ly 0 K0
  yvar jbar 1 K0
  yvar jbar 1 K1+
  yvar jbar 1 K2
  yvar jbar 1 K3
```

## Export Controlled Information

```
plot "JBAR (line 2) vs LY"
xvar ly      0 K0
yvar jbar    2 K0
yvar jbar    2 K1

plot "JBAR (line 3) vs LY"
xvar ly      0 K0
yvar jbar    3 K0
yvar jbar    3 K1

c ... time plots

plot "GAIN vs TIME"
xvar time
yvar gain   1:5  1

plot "JBAR (line 1) vs TIME"
xvar time
yvar jbar   1 n4
yvar jbar   1 n5
yvar jbar   1 n6
yvar jbar   1 n7
yvar jbar   1 n8
yvar jbar   1 n9

plot "JBAR (line 2) vs TIME"
xvar time
yvar jbar   2 n1
yvar jbar   2 n2
yvar jbar   2 n3

plot "JBAR (line 3) vs TIME"
xvar time
yvar jbar   3 n1
yvar jbar   3 n2
yvar jbar   3 n3

plot "LINEITER vs TIME"
xvar time
yvar lineiter 1:3

plot "NTRY vs TIME"
xvar time
yvar ntry
```

# Export Controlled Information

## 3-d radiation transfer

```
c **** 3D Argon sphere with Hydrogen shell (1 octant) ****
c ****      line transfer, spectrum, continuum-lowering ****
```

```
c -----
c   Aliases
c -----
```

```
alias K0    1
alias K1    13
```

```
alias K2    5
alias KMID  7
alias K3    9
```

```
alias L0    1
alias LMID  4
alias L1    7
```

```
alias M0    1
alias M1    5
```

```
alias Te    1000.
alias Ni    6.e22
```

```
alias R     0.005
```

```
c ... node numbers for edits
```

```
alias ir1  KMID
alias ir2  KMID + (L1      - 1) * K1
alias ir3  KMID + (LMID    - 1) * K1
alias ir4  ir2  + (M1 - 1) * K1 * L1
```

```
alias is1  K1
alias is2  K1 * L1
alias is3  K1 * LMID
alias is4  K1 * L1 * M1
```

```
c -----
c   Materials
c -----
```

```
atoms hydrogenic(0:5) ar
modeltype fly
```

```
atoms hydrogenic h
```

```
regionklm  K0 K1 L0 L1 M0 M1  Te
element 2 Ni
```

```
regionklm  K2 K3 L0 L1 M0 M1  Te
element 1 Ni
```

```
c -----
c   Geometry
c -----
```

# Export Controlled Information

```
geometry xyz

cone K0 K1 L0 L1 M0 M1 0. R 90.0 -90.0 90.0 0.0 1.0 1.0 1.0

c -----
c   Radiation
c -----

angles 4

ebins 61 1. 1.e4

spectrum 40      5. 1000. 1.
spectrum 20    1000. 2800. 1.
spectrum 40    2800. 4200. 1.
spectrum 20    4200. 10000. 1.

alias ISP1 60
alias ISP2 100

line 1 1 1 1 1 2
lbins 15 50.0 1.1

line 2 1 1 1 1 3
lbins 15 50.0 1.1

c -----
c   Controls
c -----

tstart 0.
tquit 4.0e-10

restart

c -----
c   Switches and Parameters
c -----


switch 8 0          ! Carlson angles
switch 11 1         ! make .plt file
switch 29 2         ! use variable timesteps
switch 30 10        ! dump every n timesteps
switch 33 1         ! static escape factors
switch 34 133       ! symmetric about k=1, l=1, l=lmax, m=1, m=mmax
switch 36 0         ! no continuum transfer
switch 37 1         ! do line transfer
switch 38 1         ! symmetric line profiles
switch 52 -1        ! do Stark broadening for everything
switch 55 1         ! do continuum lowering
switch 154 1        ! use k-axis for escape factors

param 41 1.0e-10    ! initial timestep
param 44 1.0e-10    ! minimum timestep
param 45 1.0e-8     ! maximum timestep
param 46 1.5         ! maximum timestep fractional increase
param 61 1.e-4       ! frac. population for inclusion of iso-sequence
```

# Export Controlled Information

```
c -----
c   Edits
c -----
dump mesh temperatures kappa_sp emis_sp jbar

editray 1  0.01  1.570 0
editray 2  0.01 -1.570 0
editray 3  0.01  0.001 0
editray 4  1.00  0.001 0

plot "ISO POPULATIONS vs R"
  xvar kr      0 LMID M0
  yvar yiso    1 LMID 0:5

plot "H-, He-like POPULATIONS vs R"
  xvar kr      0 LMID M0
  yvar y      1 LMID 1 1:4
  yvar y      1 LMID 2 1:4

plot "TAULINEK vs EVLINE"
  xvar evline  1
  yvar taulinek 1 is1
  yvar taulinek 1 is2
  yvar taulinek 1 is3
  yvar taulinek 1 is4

plot "TAULINEK vs EVLINE"
  xvar evline  2
  yvar taulinek 2 is1
  yvar taulinek 2 is2
  yvar taulinek 2 is3
  yvar taulinek 2 is4

plot "TAULINE+ vs EVLINE"
  xvar evline  1
  yvar tauline+ 1 is1 0 1
  yvar tauline+ 1 is2 0 2
  yvar tauline+ 1 is3 0 3
  yvar tauline+ 1 is4 0 4

plot "TAULINE+ vs EVLINE"
  xvar evline  2
  yvar tauline+ 2 is1 0 1
  yvar tauline+ 2 is2 0 2
  yvar tauline+ 2 is3 0 3
  yvar tauline+ 2 is4 0 4

plot "LKAP vs EVLINE"
  xvar evline  1
  yvar lkap     1 1
  yvar lkap     2 1

plot "LKAP vs R"
  xvar kr      0 LMID M0
  yvar lkap     1 LMID -1
  yvar lkap     2 LMID -1

plot "UVOIGT vs R"
  xvar kr      0 LMID M0
  yvar uvoigt  1 LMID
  yvar uvoigt  2 LMID
```

# Export Controlled Information

```
plot "NRB vs R"
xvar kr          0 LMID M0
yvar nrb         1 LMID
yvar nrb         2 LMID

plot "JBAR vs R"
xvar kr          0 LMID M0
yvar jbar        1 LMID
yvar jbar        2 LMID

plot "JLINE vs R"
xvar kr          0 LMID M0
yvar jline       1 LMID -1

plot "JLINE vs EVLINE"
xvar evline      1
yvar jline        1 ir1
yvar jline        1 ir2
yvar jline        1 ir3
yvar jline        1 ir4

plot "JLINE vs R"
xvar kr          0 LMID M0
yvar jline       2 LMID -1

plot "JLINE vs EVLINE"
xvar evline      2
yvar jline        2 ir1
yvar jline        2 ir2
yvar jline        2 ir3
yvar jline        2 ir4

plot "NE, TEV, TIV, VEL, ZBAR, RHO vs R"
xvar kr          0 LMID M0
yvar ne          0 LMID M0
yvar tev         0 LMID M0
yvar tiv         0 LMID M0
yvar vel          0 LMID M0
yvar zbar         0 LMID M0
yvar rho          0 LMID M0

plot "OPTICAL DEPTH vs ENERGY"
xvar sp_energy
yvar tausp+      0 is1 0 1
yvar tausp+      0 is2 0 2
yvar tausp+      0 is3 0 3
yvar tausp+      0 is4 0 4

plot "SPECTRUM vs R"
xvar kr          0 LMID M0
yvar jsp          0 LMID -1
yvar jsp          0 LMID -2
yvar jsp          0 LMID ISP1
yvar jsp          0 LMID ISP2

plot "SPECTRUM vs ENERGY"
xvar sp_energy
yvar jsp          0 ir1
yvar jsp          0 ir2
yvar jsp          0 ir3
yvar jsp          0 ir4
```

# Export Controlled Information

```
plot "ISP vs ENERGY"
send-to-dump
  xvar  sp_energy
  yvar  isp+      0  is1 0 1
  yvar  isp+      0  is2 0 2
  yvar  isp+      0  is3 0 3
  yvar  isp+      0  is4 0 4

plot "ISP vs YZ"
slice k k1
send-to-dump
  xvar  y3d
  xvar  z3d
  yvar  isp+      0  0 -1 1
  yvar  isp+      0  0 -1 2

plot "ISP vs XZ"
slice k k1
send-to-dump
  xvar  x3d
  xvar  z3d
  yvar  isp+      0  0 -1 3

plot "ISP vs XY"
slice k k1
send-to-dump
  xvar  x3d
  xvar  y3d
  yvar  isp+      0  0 -1 4

plot "ISP vs R"
send-to-dump
  xvar  r
  yvar  isp+      0  0 -1 1
  yvar  isp+      0  0 -1 2
  yvar  isp+      0  0 -1 3
  yvar  isp+      0  0 -1 4

c
c -- time edits --
c

plot "RHO, NE, TEV, TIV, NION vs TIME"
  xvar  time
  yvar  rho       0  ir1
  yvar  ne        0  ir1
  yvar  tev       0  ir1
  yvar  tiv       0  ir1
  yvar  nion      0  ir1

plot "Y1 vs TIME"
  xvar  time
  yvar  yiso      1  ir1 0:5

plot "JBAR vs TIME"
  xvar  time
  yvar  jbar       1  ir1
  yvar  jbar       1  ir2
  yvar  jbar       1  ir3
  yvar  jbar       1  ir4

plot "JBAR vs TIME"
  xvar  time
```

## Export Controlled Information

```
yvar jbar      2 ir1
yvar jbar      2 ir2
yvar jbar      2 ir3
yvar jbar      2 ir4

plot "LINEITER vs TIME"
xvar time
yvar lineiter  1
yvar lineiter  2
```

# Export Controlled Information

## 1-d Marshak wave

```
c          **** 1-d grey radiation diffusion ****

c This is a thermal wave produced by an instantaneous plane source;
c it is a self similar problem discussed in Zel'dovich and Raiser,
c "Physics of Shock Waves and High-Temperature Hydrodynamic
c Phenomena", Vol. II, Chap. XI, Section 4, page 663.

c -----
c   Aliases
c -----

alias N      101
alias DN     10

alias TE     .001
alias TR      0.
alias TB    1.e3

alias RMAX   1000.
alias TMAX   1.e-4

alias DTMIN  TMAX / 1.e2
alias DTMAX  TMAX / 10
alias DTEDIT TMAX / 10

alias ITER  100

c -----
c   Geometry
c -----

geometry slab

rlin 1 N 0.0 RMAX

c -----
c   Materials
c -----

region 1 N  TE TE TR
  background 1.e24 1.e24 1. 1. 1.
  opacity analytic constant 1.

c -----
c   Radiation
c -----

energy 1.e5

c -----
c   Sources and Boundary Conditions
c -----


boundary diffusion value 1 TB
boundary diffusion milne N 0.
```

# Export Controlled Information

```
c -----
c   Controls
c -----
tstart 0.
tquit  TMAX

restart

c -----
c   Switches and Parameters
c -----

switch 11  1          ! make .plt file
switch 20  0          ! LTE
switch 29  2          ! variable timesteps

switch 100 1          ! do radiation

param  5 0.            ! scattering multiplier
param 40 DTEDIT       ! time between edits
param 41 DTMIN        ! initial timestep
param 44 DTMIN        ! minimum timestep
param 45 DTMAX        ! maximum timestep
param 46 1.0           ! max fractional change in Te
param 48 0.0           ! e-i coupling multiplier

rswitch 1 1           ! do radiation diffusion
rswitch 3 ITER         ! number of iterations
rswitch 4 0           ! direct solution for 1 group
rswitch 5 1           ! time-dependent
rswitch 6 0           ! no flux limiting

rparam  3 1.e-4        ! convergence criterion
rparam  5 1.0           ! max fractional change in Erad
rparam  6 0.1           ! min Emate/Erad to apply rparam(5)

c -----
c   Edits
c -----

plot "TEMPERATURE vs R"
  xvar  r
  yvar  tev
  yvar  trv

plot "ITERATIONS vs TIME"
  xvar  time
  yvar  iter_radd

plot "ENERGY vs R"
  xvar  r
  yvar  emat_e
  yvar  erad

plot "TEV vs TIME"
  xvar  time
  yvar  tev      0    1:N:DN
```

## Export Controlled Information

```
plot "TRV vs TIME"
  xvar time
  yvar trv    0   1:N:DN

plot "EMAT vs TIME"
  xvar time
  yvar emat_e 0   1:N:DN

plot "ERAD vs TIME"
  xvar time
  yvar erad   0   1:N:DN

plot "INTEGRATED ENERGY vs TIME"
integrated
  xvar time
  yvar emat_e
  yvar erad
```

# Export Controlled Information

## 1-d laser-driven foam

```
c      **** laser-driven foam, 4 drive options ****
c          1: sourced laser intensity
c          2: sourced photon intensity
c          3: radiation transfer
c          4: laser ray trace

c -----
c   Aliases
c -----

alias N0      1
alias N1      21
alias DN     10
alias DR     0.1

alias TE      1.0
alias RHO    0.003
alias ABAR    1.0

alias NI      RHO * AVGD / ABAR

alias ELASER  1000.           ! laser energy (J)
alias TLASER   2.e-9          ! laser pulse length
alias RSPOT    0.1            ! laser focal spot radius

alias ILASER   ELASER * 1.e7 / (TLASER * PI * RSPOT ^ 2)

c -----
c   Materials
c -----

atoms hydrogenic h

region N0 N1 TE
element 1 NI

rlin N0 N1 0. DR

c -----
c   Radiation
c -----

ebins 61 0.1 1.e5
angles 3

c -----
c   Sources
c -----


#ifndef SOURCE1
source laser 4x value history 1 1. N0 N1
#endif

#ifndef SOURCE2
```

## Export Controlled Information

```
alias E1 4.68
alias E2 E1 * 1.001
alias MULT HPEV / (E2 - E1)
source jnu E1 E2 value history 1 MULT NO N1
#endiff

#ifndef SOURCE3
alias E1 4.68
alias E2 E1 * 1.001
alias MULT HPEV / (E2 - E1)

source jbndry 1 E1 E2 value history 1 MULT

switch 36 2                      ! turn on continuum transfer

boundary radiation streaming 1. 0. N1 xfile 1 2.
                                         ! x2 to match energy density
#endiff

#ifndef SOURCE4
switch 45 1                      ! turn on laser ray trace
laser 1 4x value history 1 1.
  lasray 0.11 1. 0. 1.
#endiff

history 1 ILASER TLASER          ! flat-topped pulse
  tv 0. 1.
  tv 1.0 1.

c -----
c   Controls
c -----
tstart 0.
tquit 3.e-9

restart

c -----
c   Switches and Parameters
c -----
switch 11 1                      ! make .plt file
switch 30 100                     ! dump every n timesteps
switch 31 1                      ! temperature calculation
switch 44 5                       ! maximum # of iterations

param 40 1.e-9                    ! time between edits
param 41 1.e-12                   ! initial timestep
param 44 1.e-12                   ! minimum timestep
param 45 1.e-10                   ! maximum timestep
param 102 TE                      ! minimum temperature for temperature calc.

c -----
c   Edits
c -----
alias IG 17                      ! group containing radiation source
editray 1 1. 0. 1
```

## Export Controlled Information

```
plot "TEV vs R"
  xvar  r
  yvar  tev

plot "ZBAR vs R"
  xvar  r
  yvar  zbar

plot "NE vs R"
  xvar  r
  yvar  ne

plot "JNU vs R"
  xvar  r
  yvar  jnu      0 0 IG

plot "TAU vs time"
  xvar  time
  yvar  tautot   0 0 IG

plot "TEV vs TIME"
  xvar  time
  yvar  tev      0 N0:N1:DN

plot "ZBAR vs TIME"
  xvar  time
  yvar  zbar     0 N0:N1:DN

plot "NI, NE vs TIME"
  xvar  time
  yvar  nion     0 N0:N1:DN
  yvar  ne       0 N0:N1:DN

plot "HEATING vs TIME"
  xvar  time
  yvar  heatl    0 N0:N1:DN
  yvar  heatt    0 N0:N1:DN

plot "NTRY vs TIME"
  xvar  time
  yvar  ntry

plot "DTIME vs TIME"
  xvar  time
  yvar  dtime
```

# Export Controlled Information

## Xfile reader

```
c           **** 1-d xfile reader ****  
c -----  
c   Materials  
c -----  
reg 1 1.  
  
c -----  
c   Geometry  
c -----  
geometry plane  
  
c -----  
c   Sources  
c -----  
  
xfile data.xfa  
  
c -----  
c   Controls  
c -----  
tstart 0.  
tquit 1.  
  
c -----  
c   Switches and Parameters  
c -----  
switch 11 1           ! make .plt file  
switch 29 0           ! use xfile timesteps  
switch 30 1           ! dump every n timesteps  
  
c -----  
c   Edits  
c -----  
  
plot "DENSITY, TEMPERATURE vs R"  
  xvar r  
  yvar rho  
  yvar tev  
  yvar tiv
```

# Export Controlled Information

## Index of Commands and Edits

—#—

#define, 49  
#elif, 21  
#else, 21  
#elseif, 21  
#endif, 21  
#ifdef, 21  
#include, 21

—A—

a\_net, 150  
a\_rate-, 149  
a\_rate+, 149  
a\_tot-, 150  
a\_tot+, 150  
abar, 138  
absn, 165  
absn\_hk, 166  
absn\_nl, 165  
aceoff, 148  
ae\_net, 152  
ae\_rate-, 151  
ae\_rate+, 151  
ae\_tot-, 152  
ae\_tot+, 152  
aemis, 148  
alias, 49  
an\_rate-, 149  
an\_rate+, 149  
angles, 33  
anuc, 141  
aprd, 38  
ar\_net, 153  
ar\_rate-, 152  
ar\_rate+, 152  
ar\_tot-, 153  
ar\_tot+, 153  
atoms, 21  
atoms hydrogenic, 21  
atoms+, 23  
averaged, 41  
avoigt, 178  
avoigt0, 148

—B—

background, 25  
bcond, 140  
bfield, 25, 140  
bnu, 169  
bnu\_sp, 176  
bnur, 169  
bnurad, 169  
boundary, 43  
brate, 169  
bratetot, 169

broadened, 41  
bsplit, 149  
btot, 169  
bx, 140  
by, 140  
bz, 140

—C—

c\_net, 150  
c\_rate-, 149  
c\_rate+, 149  
c\_tot-, 150  
c\_tot+, 150  
cdens, 132  
cdetect, 172  
ce\_net, 152  
ce\_rate-, 151  
ce\_rate+, 151  
ce\_tot-, 152  
ce\_tot+, 152  
cee, 139  
cei, 139  
cemis, 167  
cemis\_nl, 167  
cemish, 168  
cemisbb, 168  
cemisbbtot, 168  
cemisbf, 168  
cemisbftot, 168  
cemisbt, 168  
cemisbttot, 168  
cemisff, 168  
cemisfftot, 168  
cemisj, 168  
cemisjnet, 168  
cemisk, 169  
cemisnet, 169  
cemistot, 169  
cflux-, 172  
cflux+, 172  
cfluxx-, 172  
cfluxx+, 172  
cfluxy-, 172  
cfluxy+, 172  
chemu, 136  
cii, 139  
ckap, 165  
ckap\_nl, 165  
ckapb, 166  
ckapbb, 166  
ckapbf, 166  
ckapbt, 166  
ckape, 167  
ckapff, 167  
ckapj, 167  
ckapjtot, 167  
ckappa, 165

# Export Controlled Information

**ckappa\_nl**, 165  
**ckappab**, 166  
**ckappae**, 167  
**ckappaj**, 167  
**cn\_rate-**, 149  
**cn\_rate+**, 149  
**cone**, 31  
**coolb**, 158  
**coole**, 158  
**cooli**, 158  
**coolj**, 158  
**coolx**, 158  
**copac**, 166  
**copac\_nl**, 166  
**copacbb**, 166  
**copacbf**, 166  
**copacbt**, 166  
**copace**, 167  
**copacj**, 167  
**copacjtot**, 167  
**coulog**, 139  
**couloge**, 139  
**coulogi**, 139  
**cp**, 136  
**cp\_e**, 137  
**cp\_i**, 137  
**cpcgss**, 136  
**cpecgss**, 137  
**cpcgss**, 137  
**cpukin**, 188  
**cpertime**, 187  
**cputot**, 187  
**cspd**, 183  
**csrc**, 170  
**cv**, 136  
**cv\_e**, 136  
**cv\_i**, 137  
**cvecgss**, 136  
**evecgss**, 136  
**evetot**, 136  
**evetotcgs**, 136  
**cviwegs**, 137  
**cycle**, 128

**de\_net**, 152  
**de\_rate-**, 151  
**de\_rate+**, 151  
**de\_tot-**, 152  
**de\_tot+**, 152  
**dedin**, 168  
**dedne**, 137  
**dedni**, 137  
**dedr**, 138  
**dedit**, 167  
**dedte**, 137  
**dedti**, 137  
**deint0dt**, 147  
**deintdn**, 147  
**deintdt**, 147  
**deinttddr**, 147  
**deinttdt**, 147  
**deiso**, 146  
**deiso0**, 146  
**delev**, 146  
**deline**, 178  
**deline2**, 178  
**desrc**, 163  
**desrcedt**, 163  
**desrci**, 163  
**descridt**, 163  
**detector**, 39  
**dhtbdn**, 159  
**dhtbdt**, 159  
**dhtchkdn**, 162  
**dhtchkdt**, 162  
**dhtedn**, 160  
**dhtedt**, 160  
**dhtetdn**, 162  
**dhtetdt**, 162  
**dhtidt**, 160  
**dhtitdt**, 161  
**dhtjdn**, 160  
**dhtjdt**, 160  
**dhtjtdn**, 162  
**dhtjtdt**, 162  
**dhtldn**, 160  
**dhtldt**, 160  
**dhttbdn**, 161  
**dhttbdt**, 161  
**dhtxdn**, 160  
**dhtxdt**, 160  
**dhtxdn**, 161  
**dhtxtdt**, 161  
**diffj**, 186  
**diffjmax**, 186  
**diffte**, 185  
**diffti**, 185  
**difftmax**, 186  
**diffy**, 185  
**diffy1**, 185  
**diffymax**, 186  
**display**, 40  
**dn\_rate-**, 150  
**dn\_rate+**, 149  
**dnedt**, 134  
**dopwidth**, 178

## —D—

**d\_net**, 150  
**d\_rate-**, 149  
**d\_rate+**, 149  
**d\_tot-**, 150  
**d\_tot+**, 150  
**dadn**, 166  
**dadt**, 166  
**dclbdn**, 160  
**dclbdt**, 160  
**dcledn**, 161  
**dcledt**, 160  
**dclidt**, 161  
**dcljdn**, 161  
**dcljdt**, 161  
**dclxdn**, 161  
**dclxdt**, 161

# Export Controlled Information

**dpdne**, 138  
**dpdni**, 138  
**dpdr**, 138  
**dpdte**, 138  
**dpdti**, 138  
**drat**, 39  
**dtfrac**, 129  
**dtime**, 129  
**dtime\_rho**, 129  
**dtime\_te**, 129  
**dtime\_ti**, 129  
**dtime\_tr**, 129  
**dtimek**, 129  
**dtkin**, 129  
**dtkmax**, 129  
**dtkmin**, 129  
**dump**, 46  
**dumpedit**, 40  
**dumpfile**, 47  
**dvdr**, 31, 132  
**dyldn**, 143  
**dyldr**, 143  
**dyldt**, 143  
**dydn**, 142  
**dydr**, 142  
**dydt**, 142  
**dyisodn**, 143  
**dyisodr**, 143  
**dyisodt**, 143  
**dyldj**, 179  
**dyudj**, 179

**ee\_rate+**, 151  
**eebins**, 38, 141  
**eenergy**, 38, 140  
**eff\_sp**, 174  
**effi\_sp**, 174  
**eflux-**, 172  
**eflux+**, 172  
**efluxx-**, 172  
**efluxx+**, 172  
**efluxy-**, 172  
**efluxy+**, 172  
**eigenv\_i**, 157  
**eigenv\_r**, 157  
**eigenval**, 157  
**eigenvec**, 157  
**eigenvec\_i**, 157  
**eigenvec\_r**, 157  
**eint**, 147  
**eint0**, 147  
**eintcl**, 147  
**einttot**, 147  
**eiso**, 146  
**eiso0**, 145  
**ekap**, 165  
**ekap\_nl**, 165  
**element**, 24  
**elev**, 146  
**elev0**, 146  
**eline**, 178  
**elower**, 147  
**elowerdn**, 147  
**elowerdt**, 147  
**emat**, 135  
**emat\_e**, 135  
**emat\_i**, 135  
**emis**, 167  
**emis\_nl**, 167  
**emis\_sp**, 174  
**emis\_sp-**, 174  
**emis\_sp+**, 174  
**emislist**, 48  
**emisnet**, 162  
**emistot**, 162  
**end**, 48  
**energy**, 32, 163  
**energy0**, 32  
**energy-averaged**, 41  
**energy-integrated**, 41  
**eopac**, 165  
**eopac\_nl**, 165  
**eos**, 27  
**eos leos**, 27  
**eofile**, 27  
**ep\_rate-**, 151  
**ep\_rate+**, 151  
**erad**, 136, 171  
**erad2**, 171  
**eradc**, 171  
**eradl**, 171  
**eresist**, 140  
**err\_temp**, 186  
**escprob**, 149

## —E—

**e\_tot-**, 150  
**e\_tot+**, 150  
**ea\_rate-**, 151  
**ea\_rate+**, 151  
**eabsn**, 165  
**eabsn\_nl**, 165  
**ebb\_sp**, 174  
**ebbi\_sp**, 174  
**ebbtot**, 153  
**ebbtot0**, 153  
**ebbtot1**, 153  
**ebf\_sp**, 174  
**ebfi\_sp**, 174  
**ebins**, 32, 163  
**ebins2**, 33  
**ebt\_sp**, 174  
**ebti\_sp**, 174  
**ec\_rate-**, 150  
**ec\_rate+**, 150  
**echo**, 49  
**econd**, 25, 140, 162  
**ed\_rate-**, 151  
**ed\_rate+**, 151  
**edetect**, 172  
**edist**, 141  
**edit**, 40  
**editray**, 39  
**ee\_rate-**, 151

# Export Controlled Information

**esource**, 27  
**esourci**, 27  
**esp\_0d**, 178  
**esrce**, 163  
**esrcev**, 163  
**esrci**, 163  
**esrciv**, 163  
**etot**, 136  
**evline**, 180

## —F—

**fbins**, 32, 164  
**fillpct**, 157  
**fillpct-**, 157  
**fillpct+**, 157  
**freq**, 164  
**frequency**, 32  
**fugacity**, 136

## —G—

**gain**, 148  
**gainlist**, 48  
**gamma**, 145  
**gamma\_a**, 156  
**gamma\_c**, 156  
**gamma\_ci**, 156  
**gamma\_cx**, 156  
**gamma\_d**, 157  
**gamma\_di**, 156  
**gamma\_dx**, 156  
**gamma\_p**, 156  
**gamma\_pi**, 156  
**gamma\_px**, 156  
**gamma1**, 137  
**gamma1\_e**, 137  
**gamma1\_i**, 137  
**gammatot**, 145  
**geometry**, 30  
**gff**, 169  
**gion**, 146  
**gion0**, 146  
**glev**, 146  
**glev0**, 146  
**gplasma**, 140

## —H—

**heatb**, 158  
**heatbt**, 159  
**heatchk**, 159  
**heate**, 158  
**heatet**, 159  
**heati**, 158  
**heatit**, 159  
**heatj**, 158  
**heatjt**, 159  
**heatl**, 158  
**heatt**, 159  
**heatti**, 159  
**heatx**, 158

**heatxt**, 159  
**heff0**, 162  
**heff1**, 162  
**hex**, 31  
**history**, 45

## —I—

**icond**, 140, 162  
**id\_dtime**, 129  
**idetect**, 182  
**iflux-**, 181  
**iflux+**, 181  
**ifluxx-**, 181  
**ifluxx+**, 181  
**ifluxy-**, 181  
**ifluxy+**, 181  
**ifr**, 163  
**ifrline**, 180  
**ilevmax**, 145  
**ilevmin**, 145  
**iline-**, 181  
**iline+**, 181  
**integrated**, 41  
**inu**, 172  
**inu-**, 172  
**-ir**, 130  
**ir**, 130  
**ir\_dtime**, 130  
**ir\_dtimek**, 130  
**ireg**, 130  
**iso**, 141  
**isofilter**, 35  
**isoflux-**, 155  
**isoflux\_a**, 156  
**isoflux\_c**, 155  
**isoflux\_p**, 155  
**isoflux+**, 155  
**isoflux+\_a**, 156  
**isoflux+\_c**, 155  
**isoflux+\_p**, 155  
**isofluxn**, 155  
**isofluxn\_a**, 156  
**isofluxn\_c**, 155  
**isofluxn\_p**, 155  
**isomax**, 145  
**isomin**, 145  
**isorange**, 22  
**isorate-**, 153  
**isorate\_a**, 154  
**isorate\_c**, 154  
**isorate\_p**, 154  
**isorate\_x**, 154  
**isorate+**, 153  
**isorate+\_a**, 154  
**isorate+\_c**, 154  
**isorate+\_p**, 154  
**isorate+\_x**, 154  
**isoraten**, 153  
**isoraten\_a**, 154  
**isoraten\_c**, 154  
**isoraten\_p**, 154

# Export Controlled Information

**isoraten\_x**, 155  
**isp**, 173  
**isp-**, 176  
**isp\_0d**, 178  
**isp+**, 176  
**iter**, 128  
**iter\_temp**, 186  
**itime**, 128  
**iz\_dtime**, 130  
**iz\_dtimek**, 130

## —J—

**jbar**, 178  
**jbarcgs**, 178  
**jbarx**, 179  
**jbnd2**, 169  
**jbndry**, 169  
**jdnu**, 170  
**jemis**, 170  
**jline**, 180  
**jline0**, 180  
**jline0cgs**, 180  
**jlinecgs**, 180  
**jmode**, 178  
**jnu**, 170  
**jnu2**, 170  
**jnurea**, 173  
**jnul**, 170  
**jnuline**, 173  
**jnutot**, 170  
**joinline**, 37  
**jrate**, 170  
**jratei**, 170  
**jratex**, 170  
**jratio**, 170  
**jsat**, 179  
**jsatcgs**, 179  
**jsp**, 176  
**jsparea**, 177  
**jspline**, 177  
**jspon**, 170

## —K—

**k**, 132  
**kappa\_sp**, 174  
**kappa\_sp-**, 174  
**kappa\_sp\_hk**, 175  
**kappa\_sp+**, 174  
**kb\_sp**, 175  
**kbpi\_sp**, 175  
**kbfp\_sp**, 175  
**kbfi\_sp**, 175  
**kbt\_sp**, 175  
**kbtsp**, 175  
**kff\_sp**, 175  
**kffi\_sp**, 176  
**kmax**, 130  
**kr**, 132  
**kx**, 132  
**ky**, 132

**kz**, 133

## —L—

**l**, 133  
**laser**, 38  
**lasray**, 38  
**lbins**, 37  
**ldetect**, 182  
**lemis**, 180  
**lemis-**, 182  
**lemis+**, 182  
**lemiscgs**, 180  
**lemiscgs-**, 182  
**lemiscgs+**, 182  
**lemist**, 180  
**lemist-**, 182  
**lemist+**, 182  
**lemistcgs**, 180  
**lemistcgs-**, 182  
**lemistcgs+**, 182  
**lemisttot**, 180  
**level**, 24, 141  
**lflux-**, 181  
**lflux+**, 181  
**lfluxx-**, 182  
**lfluxx+**, 182  
**lfluxy-**, 182  
**lfluxy+**, 182  
**line**, 36  
**linedefault**, 37  
**lineiter**, 179  
**linetype**, 36  
**lkap**, 180  
**lkap-**, 182  
**lkap+**, 182  
**lkapt**, 180  
**lkapt-**, 183  
**lkapt+**, 182  
**lkapttot**, 180  
**lline-**, 181  
**lline+**, 181  
**lmax**, 130  
**lnuarea**, 183  
**lnuline**, 183  
**lpump**, 48  
**lr**, 133  
**lteflag**, 134  
**ltotarea**, 183  
**ltotline**, 183  
**lx**, 133  
**ly**, 133  
**lz**, 133

## —M—

**m**, 133  
**mass-averaged**, 41  
**material**, 25  
**maximum**, 41  
**memmax**, 188  
**memory**, 188

# Export Controlled Information

**minimum**, 41  
**mmax**, 130  
**modeltype**, 22  
**mr**, 133  
**mx**, 133  
**my**, 133  
**my1z**, 187  
**myvz**, 187  
**myz**, 187  
**mz**, 133

## —N—

**nbound**, 145  
**nbound\_lte**, 145  
**ndiff\_iso**, 186  
**ndiff\_iz**, 186  
**ndiff\_line**, 186  
**ndiffj**, 186  
**ndifft**, 186  
**ndiffy**, 186  
**ndtct**, 164  
**ne**, 25, 134  
**ne\_lte**, 134  
**ne\_thermal**, 135  
**neb**, 134  
**necrit**, 185  
**nedist**, 141  
**nee**, 140  
**nehot**, 25, 135  
**neray**, 164  
**nescat**, 169  
**newxfile**, 43  
**nf**, 163, 187  
**ng**, 187  
**ni**, 134, 143  
**nib**, 134  
**niso**, 141  
**nlev**, 141  
**nlq**, 142  
**nltdedump**, 47  
**nnode**, 130  
**npq**, 141  
**npqmax**, 142  
**nr**, 130, 186  
**nray**, 164  
**nrb**, 179  
**nsp**, 173  
**nt**, 130, 187  
**ntot**, 134  
**ntry**, 185  
**ntryz**, 185  
**nu**, 164  
**nuline**, 180  
**nxtndy**, 187  
**nxtndy1**, 187  
**nxtndyv**, 187  
**ny1z**, 187  
**nyvz**, 187  
**nyz**, 187  
**nzone**, 130

**opac**, 166  
**opac\_hk**, 166  
**opac\_nl**, 166  
**opac\_sp**, 175  
**opac\_sp-**, 175  
**opac\_sp\_hk**, 175  
**opac\_sp+**, 175  
**opacb**, 166  
**opacdum**, 47  
**opace**, 167  
**opacfile**, 33  
**opacity**, 26  
**opacity cold**, 26  
**opacity exro**, 26  
**opacity file**, 25  
**opacity henke**, 26  
**opacity server**, 25  
**opacity spectral**, 26  
**opbb\_sp**, 175  
**opbbi\_sp**, 175  
**opbf\_sp**, 175  
**opbfi\_sp**, 175  
**opbt\_sp**, 175  
**opbti\_sp**, 175  
**opff\_sp**, 175  
**opffi\_sp**, 176

## —P—

**p\_net**, 150  
**p\_rate-**, 149  
**p\_rate+**, 149  
**p\_tot-**, 150  
**p\_tot+**, 150  
**pabsn**, 165  
**pabsn\_nl**, 165  
**param**, 48  
**pbinlin**, 33  
**pbins**, 33  
**pbins2**, 33  
**pe\_net**, 152  
**pe\_rate-**, 151  
**pe\_rate+**, 151  
**pe\_tot-**, 152  
**pe\_tot+**, 152  
**pfnc**, 148  
**pfnc0**, 148  
**pgain**, 158  
**pkap**, 165  
**pkap\_nl**, 165  
**plaser-**, 185  
**plaser+**, 185  
**plev**, 146  
**ploss**, 159  
**plot**, 40  
**pmass**, 183  
**pmat**, 138  
**pmat\_e**, 138  
**pmat\_i**, 138  
**pn\_rate-**, 149

# Export Controlled Information

**pn\_rate+**, 149  
**pnet**, 159  
**popac**, 165  
**popac\_nl**, 165  
**pr\_net**, 153  
**pr\_rate-**, 152  
**pr\_rate+**, 152  
**pr\_tot-**, 152  
**pr\_tot+**, 152  
**prad**, 138, 171  
**pradc**, 171  
**pradl**, 171  
**prdfrac**, 179  
**product\_mesh**, 31  
**ptot**, 138  
**pump**, 149

**ropac**, 164  
**ropac\_nl**, 165  
**rp\_rate-**, 151  
**rp\_rate+**, 151  
**rprd**, 37

## —S—

**scale**, 28  
**scalekl**, 28  
**scaleklm**, 29  
**scaleklmr**, 29  
**scaleklr**, 28  
**scat**, 169  
**sdump**, 46  
**send-to-dump**, 40  
**sense**, 47  
**sgauss**, 28  
**shell**, 142  
**sigma**, 179  
**slice**, 41  
**snu**, 170  
**source**, 43  
**sp\_energy**, 173  
**sp\_freq**, 173  
**sp\_nu**, 173  
**sp\_wvl**, 173  
**sp-averaged**, 41  
**spdetect**, 177  
**spectral-group**, 35  
**spectrum**, 34  
**spectrum\_ebins**, 35  
**spectrum\_lines**, 35  
**spectrum\_resolution**, 34  
**speflux-**, 177  
**speflux+**, 177  
**spefluxx-**, 177  
**spefluxx+**, 177  
**spefluxy-**, 177  
**spefluxy+**, 177  
**spflux-**, 176  
**spflux+**, 176  
**spfluxx-**, 176  
**spfluxx+**, 176  
**spfluxy-**, 176  
**spfluxy+**, 176  
**sp-integrated**, 41  
**sprd**, 38  
**src\_sp**, 176  
**srcline**, 181  
**srcline-**, 183  
**srcline+**, 183  
**srlinecgs**, 181  
**srlinecgs-**, 183  
**srlinecgs+**, 183  
**stanh**, 28  
**stark**, 36  
**subatoms**, 23  
**sv**, 46  
**svlist**, 45  
**switch**, 48

## —Q—

**qmoment**, 139  
**qsd**, 184  
**qsflg**, 184  
**qst**, 184  
**qstart**, 27  
**quad**, 31  
**quadb**, 31

## —R—

**-r**, 131  
**r**, 30, 131  
**r2d**, 131  
**r2file**, 37  
**r3d**, 131  
**rabsn**, 164  
**rabsn\_nl**, 164  
**rabsn0**, 164  
**rad-units**, 41  
**ratedump**, 47  
**rates**, 157  
**rates-**, 157  
**rates+**, 157  
**rbinlin**, 33  
**rbins**, 33  
**rbins2**, 33  
**reg**, 24  
**region**, 23  
**regionkl**, 24  
**regionklm**, 24  
**regmap**, 24  
**regmult**, 24  
**resonance**, 37  
**resonant**, 37  
**restart**, 46  
**rexp**, 31  
**rgeom**, 30  
**rho**, 24, 134  
**rhodr**, 132  
**rkap**, 164  
**rkap\_nl**, 164  
**rlin**, 30  
**rlog**, 30

# Export Controlled Information

## —T—

**t\_net**, 150  
**t\_rate-**, 149  
**t\_rate+**, 149  
**t\_tot-**, 150  
**t\_tot+**, 150  
**table**, 40  
**tauee**, 139  
**tauei**, 139  
**tauui**, 139  
**taukap**, 171  
**taukapk**, 171  
**taukapl**, 171  
**taukapm**, 171  
**tauline**, 181  
**tauline-**, 181  
**tauline+**, 181  
**taulinek**, 181  
**taulinel**, 181  
**taulinem**, 181  
**tauray**, 173  
**tauscat**, 171  
**tauscatk**, 171  
**tauscatl**, 171  
**tauscatm**, 171  
**tausp**, 177  
**tausp-**, 177  
**tausp+**, 177  
**tauspk**, 177  
**tauspl**, 177  
**tauspm**, 177  
**tautot**, 172  
**tautotk**, 172  
**tautotl**, 172  
**tautotm**, 172  
**tdfac**, 135  
**te**, 134  
**te\_lte**, 134  
**te\_net**, 152  
**te\_rate-**, 151  
**te\_rate+**, 151  
**te\_tot-**, 152  
**te\_tot+**, 152  
**tehot**, 25, 135  
**tev**, 134  
**tev\_lte**, 134  
**tevb**, 134  
**tfermi**, 135  
**ti**, 135  
**time**, 129  
**time-averaged**, 41  
**time-integrated**, 40  
**time-maximum**, 41  
**time-minimum**, 41  
**tiv**, 135  
**tivb**, 135  
**tn\_rate-**, 150  
**tn\_rate+**, 150  
**tp**, 45  
**tprofile**, 29  
**tprofile from**, 29

**tprofile zonal**, 29  
**tquit**, 42  
**tr**, 135  
**tr\_net**, 153  
**tr\_rate-**, 152  
**tr\_rate+**, 152  
**tr\_sp**, 176  
**tr\_tot-**, 153  
**tr\_tot+**, 153  
**trad**, 171  
**tradv**, 171  
**transition**, 39  
**trv**, 135  
**tstart**, 42  
**tv**, 45

## —U—

**u2d**, 32, 132  
**u3d**, 32, 132  
**uvogt**, 178  
**uvogt0**, 148

## —V—

**v2d**, 32, 132  
**v3d**, 32, 132  
**vacuum**, 27  
**vel**, 132  
**velocity**, 31

## —W—

**w2d**, 32, 132  
**w3d**, 32, 132  
**wavelength**, 32  
**wbins**, 33, 164  
**wedgek**, 31  
**wedgel**, 31  
**when**, 48  
**widthd**, 148  
**widthn**, 148  
**widths**, 148  
**wlev**, 146  
**wplasma**, 139  
**wtau**, 140  
**wvl**, 164

## —X—

**-x**, 131  
**x**, 30, 131  
**x2d**, 131  
**x3d**, 131  
**xexp**, 31  
**xfile**, 42  
**xfilebc**, 34  
**xgeom**, 30  
**xlin**, 30  
**xlog**, 30  
**xmult**, 42

# Export Controlled Information

**xvar**, 42  
**xy**, 131  
**xzc**, 131

## —Y—

**y**, 30, 142  
**y\_flux**, 153  
**y\_flux-**, 153  
**y\_flux+**, 153  
**y\_lte**, 142  
**y1**, 143  
**y1\_lte**, 143  
**y1frac**, 143  
**y1frac\_lte**, 143  
**y1x**, 143  
**y2d**, 131  
**y3d**, 131  
**yeff0**, 143  
**yeff1**, 143  
**yexp**, 31  
**yfrac**, 142  
**yfrac\_lte**, 142  
**yfrac\_v**, 145  
**yfrac1**, 142  
**yfrac1\_lte**, 142  
**yfraciso**, 142  
**yfraciso\_lte**, 142  
**yfshell**, 144  
**yg**, 142  
**yg\_lte**, 142  
**ygeom**, 30  
**ygfrac**, 142  
**ygfrac\_lte**, 142  
**yiso**, 143  
**yiso\_lte**, 143  
**yiso\_v**, 144  
**yisoc**, 145  
**yisofrac**, 143  
**yisofrac\_lte**, 143  
**yisofrac\_v**, 144  
**yisox**, 143  
**ylin**, 30  
**ylinec**, 179  
**ylinel**, 179  
**ylineu**, 179

**ylog**, 30  
**ym**, 144  
**ymf**, 144  
**yn**, 144  
**ynf**, 144  
**yshell**, 144  
**ytot**, 143  
**ytot\_lte**, 143  
**ytot\_v**, 144  
**yvar**, 42  
**yzc**, 131

## —Z—

**-z**, 131  
**z**, 30, 131  
**z2abar**, 139  
**z2bar**, 139  
**z2d**, 131  
**z3d**, 131  
**zbar**, 138, 145  
**zbar\_lte**, 138, 145  
**zbnd**, 145  
**zbnd\_lte**, 145  
**ze**, 184  
**zekin**, 184  
**zektot**, 185  
**zeta**, 136  
**zetot**, 184  
**zexp**, 31  
**zgam**, 185  
**zgeom**, 30  
**ziso**, 141  
**zktot**, 184  
**zlin**, 30  
**zlog**, 30  
**zmass**, 184  
**znuc**, 141  
**zp**, 184  
**zpv**, 184  
**zq**, 184  
**zqv**, 184  
**zrho**, 184  
**zv**, 185  
**zwindow**, 23  
**zzc**, 131

# Export Controlled Information

## Appendix: ULTRA and PDBVIEW basics

ULTRA is a program for the presentation, manipulation and analysis of 1-d data sets, i.e. curves. It can read and write ASCII data files or PDB files with ULTRA curves in them, such as the *.ult* files produced by CRETIN.

Before using ULTRA on a Unix system, you must define an ULTRA environment variable, specifying the directory for various configuration and extension files. The default behavior of ULTRA may be modified with a *.ultrarc* file. The following entries in *.ultrarc* are recommended for use with CRETIN (Note: the parentheses are required and the comments must be placed on lines beginning with semicolons):

(label-length 60)	[allow viewing the entire curve label created by CRETIN]
(label-space 0.2)	[allow space for labels under the graphic display]
(label-type-size 10)	[font size for labels]

In ULTRA commands, spaces delimit items on the input line. Semicolons may be used to stack multiple commands on a single input line. Ranges of curve numbers or letters may be indicated using colon notation, e.g. 3:6 or a:d.

To start ULTRA and open file *data.ult*, type

**ultra** *data.ult*

or to open file *data.ult* from within ULTRA, type

**rd** *data.ult*

More than one file may be open simultaneously.

To list the curves in *data.ult*, type

**menu**

or to list all curves with labels starting with *abc*, type

**menu** *abc\**

or to list all curves with labels containing the string *abc*, type

**menu** “\**abc*\*”

or to send the list of curves to a file, type

**print-menu** [*filename*]

where *filename* is optional and defaults to “ultra.menu”.

To display curves numbered 8 and 12, type

**cur** 8 12

To list the curves currently displayed, type

**lst**

To delete curves labeled *e* and *g* (where the labels are those on screen or returned by **lst**), type

**del** *e g*

To erase all curves currently displayed, type

**era**

To send the current graphics display to a postscript file (defaulted to “plots.ps”), type

**print**

To make a color postscript file, type

**ps-type** “color”

To set the name of postscript file to *filename*, type

# Export Controlled Information

**ps-name "filename"**

To exit ULTRA, type  
**end**

ULTRA has many capabilities for manipulating and combining curves. Descriptions of most commands can be obtained from ULTRA's help capabilities. To list available commands, type

**help**

To obtain information about a given command, type  
**help command**

If you don't know the name of a command, choose an appropriate keyword and type  
**apropos keyword**

PDBVIEW is a program for working with PDB files, such as the various output files (*.ult*, *.rxx*, *.sxx*, *.dxx*) produced by CRETIN. It allows browsing PDB files graphically or textually and also allows editing PDB files.

Before using PDBVIEW, you must define a SCHEME environment variable, specifying the directory for various configuration and extension files.

To start PDBVIEW and open file *data.pdb*, type

**pdbview data.pdb**

or to open file *data.pdb* from within PDBVIEW, type  
**cf data.pdb**

More than one file may be open simultaneously, but commands apply to only a single file. To list the files currently open, type

**if**

To change the focus to the file *data.pdb* which has already been opened, type  
**cf data.pdb** or **cf fx**, where *fx* is file label returned by the **if** command.

When dealing with a *.ult* file, PDBVIEW can graphically display both 1-d data sets (curves), 2-d data sets (mappings) and images. To list all curves, mappings and images, type

**menu**

or to list only mappings, type

**menu "\*/%\*"**

(This takes advantage of the titles assigned to 2-d and 3-d data sets by CRETIN).

To display curve/mapping/image number 5, type

**pl 5**

The default rendering for a 2-d mapping is a contour plot. To change the rendering, type

**vr mode**

where *mode* is one of the following: "contour", "image", "fill-poly", "shaded", "wire-frame", or "mesh".

To list the curves/mappings currently displayed, type

**lsv**

To delete the mapping labeled 2 (where the label is that returned by **lsv**), type

**dl 2**

To delete all mappings from the window, type

**clv**

# Export Controlled Information

To send the current graphics display to a postscript file, type

**hc [color]**

where **color** must be specified on the first **hc** command for a color postscript file.

PDBVIEW can display multiple windows simultaneously, and multiple viewports in a given window. See the PDBVIEW documentation or help package for more information. The above commands are appropriate for the current window with a single viewport.

To create a new window, type

**cw "title"**

where *title* will appear at the top of the new window. The quotes are necessary if *title* is ascii, but not if *title* is numeric. A **cw** command with no arguments will cycle through the existing windows, making each one active in turn.

When dealing with a general PDB file, such as a dump file (.rxx, .sxx, .dxx) from CRETIN, PDBVIEW can list and edit data from the file. To list the contents of a file, type

**ls**

The file may contain a directory structure which may be navigated with Unix commands.

To describe the structure and format of variable *var*, type

**desc var**

To list the data contained in variable *var*, type

**print var**

or simply

*var*

To list the subset ( $i=2-4, j=7-10$ ) of a 2-d array *var(i,j)*, type

**[print] var[2:4,7:10]**

To change the dimensions of a variable (for the current PDBVIEW session only), type

**chdim var dim1 dim2 dim3 ...**

To set element 9 of a 1-d array *var(i)* to have *value*, type

**change var[9] value**

or simply

*var[9] value*

To plot a variable *yvar* against a 1-d variable *xvar*, type

**plot yvar xvar**

To plot a variable *zvar* on a 2-d mesh consisting variables *xvar* and *yvar*, type

**plot zvar '(xvar yvar)**

To exit PDBVIEW, type

**end**

PDBVIEW has many more capabilities for viewing and manipulating data. To list the available commands, type

**help**

or to receive a description of *command*, type

**help command**

## Export Controlled Information

ULTRA, PDBVIEW and PDB are part of the Portable Application Code Toolkit (PACT), available at  
<http://pact.llnl.gov>.