CRETIN

Session 4

Howard Scott

April 9, 2018

LLNL-PRES-749762

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

Session topics

Atomic models

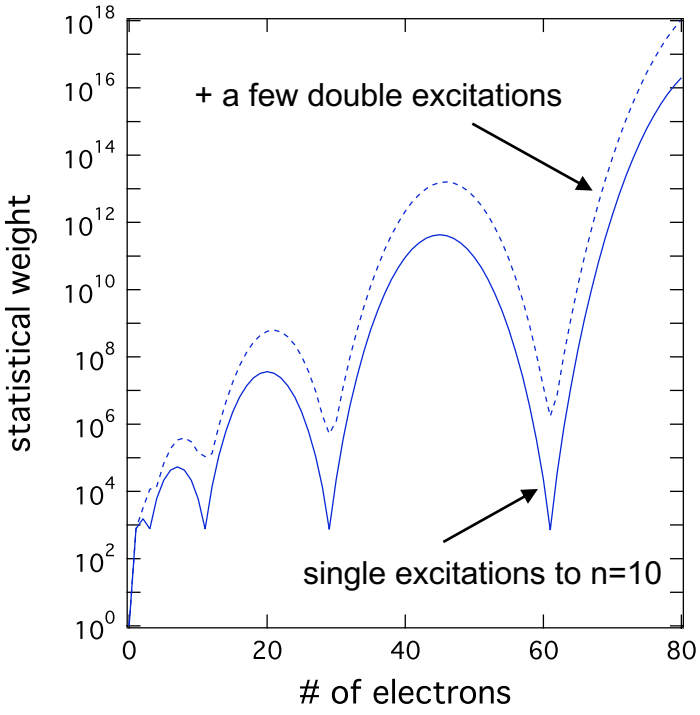
- Where to find them
- How to interpret them
  - Levels
  - Transitions
- Identifying transitions
- Constructing atomic models
- Useful command line options

General comments on atomic models

- Atomic models provide the data for everything simulated by Cretin – atomic kinetics, radiation transfer, EOS, transport coefficients, …
- Even simple atomic systems are way too complex to model in detail
  - see ”Advances in NLTE modeling for integrated simulations” (DCA_2010.pdf in /usr/apps/cretin/docs) for a discussion
- The requirements for an atomic model depend on the application

$\Longrightarrow$ Generally available atomic models are

1. Approximate and highly averaged
2. May not be suitable for your simulation
3. Often do a decent job
   - but don’t look too closely



Specifying atomic models

atoms dca_18k ar

"dca_18k": designate atomic datafile

Where does the code look for dca_18k?

1. Current working directory
2. Directory holding the generator
3. Directory identified by environment variable AFILEDIR
4. /usr/gdata/dca/Models/latest → /usr/gdata/dca/Models/2014_03_03

Recommendation:

/usr/gdata/dca/Models/2017_09_08 for standard models: dca_*xx*

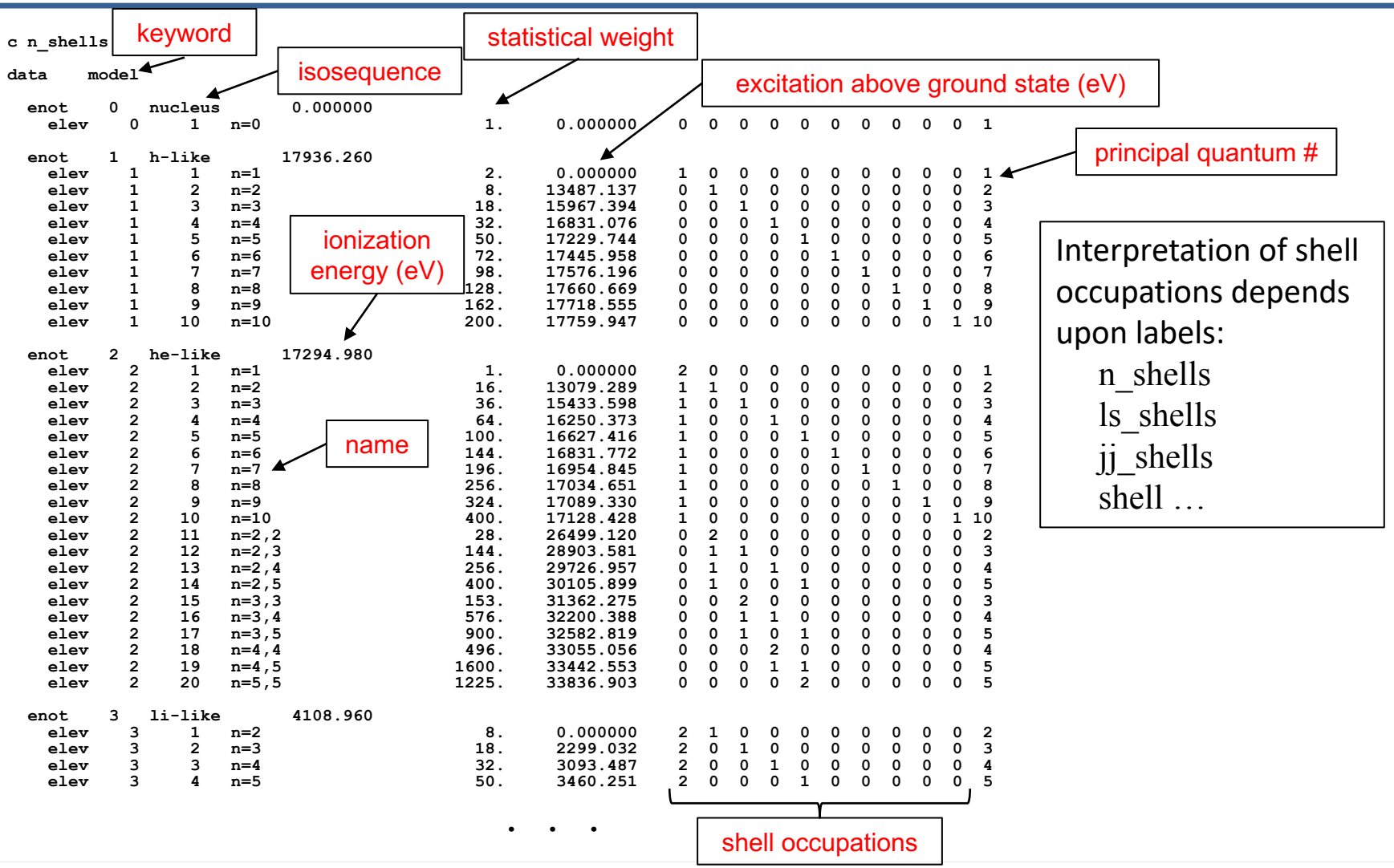
/usr/gdata/dca/Models/latest for K-shell models: dca_*xx*k, k5, ka

Atomic model organization

- Data sections describing
  1. atomic structure
  2. photoexcitation data – oscillator strength + …
  3. photoionization cross sections
  4. e^- collisional excitation cross sections / rate coefficients
  5. e^- collisional ionization cross sections / rate coefficients
  6. autoionization (Auger) rates
  7. Δn=0 transitions + virtual state transitions [screened-hydrogenic models only]
- Data sections are marked by "data *keyword*" and "end data"
- Cross sections are described by a functional form

  e.g. $\sigma(\nu) = \sigma(\nu_0)({}^{\nu_0}/_{\nu})^3$
- Data sections contain parameters used in the functional form
- Each section has multiple options for parameterizations distinguished by keywords, # of parameters and values

Atomic structure section – dca_36

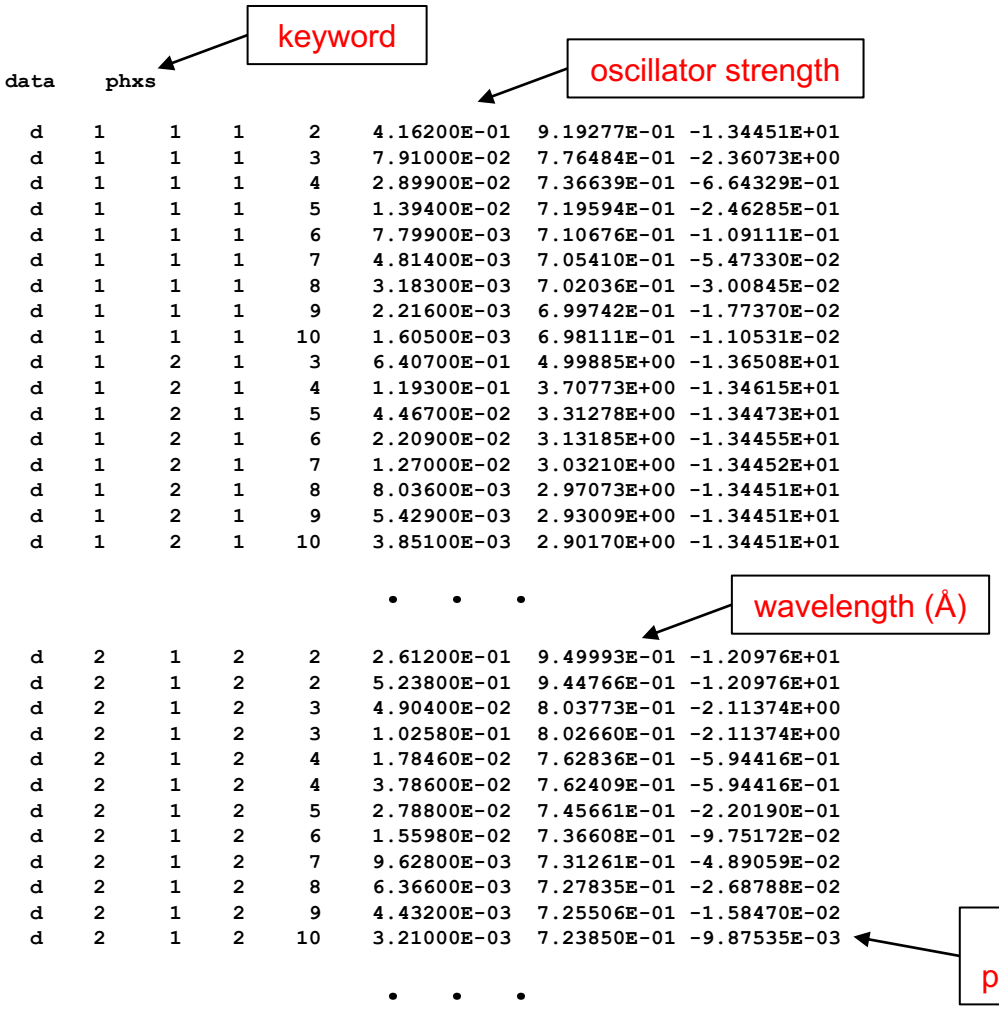



```
c n_shells
data     model

  enot    0   nucleus          0.000000
    elev    0      1   n=0                    1.      0.000000    0  0  0  0  0  0  0  0  0  0  1

  enot    1   h-like       17936.260
    elev    1      1   n=1                    2.      0.000000    1  0  0  0  0  0  0  0  0  0  1
    elev    1      2   n=2                    8.     13487.137    0  1  0  0  0  0  0  0  0  0  2
    elev    1      3   n=3                   18.     15967.394    0  0  1  0  0  0  0  0  0  0  3
    elev    1      4   n=4                   32.     16831.076    0  0  0  1  0  0  0  0  0  0  4
    elev    1      5   n=5                   50.     17229.744    0  0  0  0  1  0  0  0  0  0  5
    elev    1      6   n=6                   72.     17445.958    0  0  0  0  0  1  0  0  0  0  6
    elev    1      7   n=7                   98.     17576.196    0  0  0  0  0  0  1  0  0  0  7
    elev    1      8   n=8                  128.     17660.669    0  0  0  0  0  0  0  1  0  0  8
    elev    1      9   n=9                  162.     17718.555    0  0  0  0  0  0  0  0  1  0  9
    elev    1     10   n=10                 200.     17759.947    0  0  0  0  0  0  0  0  0  1 10

  enot    2   he-like      17294.980
    elev    2      1   n=1                    1.      0.000000    2  0  0  0  0  0  0  0  0  0  1
    elev    2      2   n=2                   16.     13079.289    1  1  0  0  0  0  0  0  0  0  2
    elev    2      3   n=3                   36.     15433.598    1  0  1  0  0  0  0  0  0  0  3
    elev    2      4   n=4                   64.     16250.373    1  0  0  1  0  0  0  0  0  0  4
    elev    2      5   n=5                  100.     16627.416    1  0  0  0  1  0  0  0  0  0  5
    elev    2      6   n=6                  144.     16831.772    1  0  0  0  0  1  0  0  0  0  6
    elev    2      7   n=7                  196.     16954.845    1  0  0  0  0  0  1  0  0  0  7
    elev    2      8   n=8                  256.     17034.651    1  0  0  0  0  0  0  1  0  0  8
    elev    2      9   n=9                  324.     17089.330    1  0  0  0  0  0  0  0  1  0  9
    elev    2     10   n=10                 400.     17128.428    1  0  0  0  0  0  0  0  0  1 10
    elev    2     11   n=2,2                 28.     26499.120    0  2  0  0  0  0  0  0  0  0  2
    elev    2     12   n=2,3                144.     28903.581    0  1  1  0  0  0  0  0  0  0  3
    elev    2     13   n=2,4                256.     29726.957    0  1  0  1  0  0  0  0  0  0  4
    elev    2     14   n=2,5                400.     30105.899    0  1  0  0  1  0  0  0  0  0  5
    elev    2     15   n=3,3                153.     31362.275    0  0  2  0  0  0  0  0  0  0  3
    elev    2     16   n=3,4                576.     32200.388    0  0  1  1  0  0  0  0  0  0  4
    elev    2     17   n=3,5                900.     32582.819    0  0  1  0  1  0  0  0  0  0  5
    elev    2     18   n=4,4                496.     33055.056    0  0  0  2  0  0  0  0  0  0  4
    elev    2     19   n=4,5               1600.     33442.553    0  0  0  1  1  0  0  0  0  0  5
    elev    2     20   n=5,5               1225.     33836.903    0  0  0  0  2  0  0  0  0  0  5

  enot    3   li-like       4108.960
    elev    3      1   n=2                    8.      0.000000    2  1  0  0  0  0  0  0  0  0  2
    elev    3      2   n=3                   18.      2299.032    2  0  1  0  0  0  0  0  0  0  3
    elev    3      3   n=4                   32.      3093.487    2  0  0  1  0  0  0  0  0  0  4
    elev    3      4   n=5                   50.      3460.251    2  0  0  0  1  0  0  0  0  0  5

                                                                                 .  .  .
```

Interpretation of shell occupations depends upon labels:

- n_shells
- ls_shells
- jj_shells
- shell …

Photoexcitation section – dca_36




```
data    phxs

  d    1     1    1     2    4.16200E-01  9.19277E-01 -1.34451E+01
  d    1     1    1     3    7.91000E-02  7.76484E-01 -2.36073E+00
  d    1     1    1     4    2.89900E-02  7.36639E-01 -6.64329E-01
  d    1     1    1     5    1.39400E-02  7.19594E-01 -2.46285E-01
  d    1     1    1     6    7.79900E-03  7.10676E-01 -1.09111E-01
  d    1     1    1     7    4.81400E-03  7.05410E-01 -5.47330E-02
  d    1     1    1     8    3.18300E-03  7.02036E-01 -3.00845E-02
  d    1     1    1     9    2.21600E-03  6.99742E-01 -1.77370E-02
  d    1     1    1    10    1.60500E-03  6.98111E-01 -1.10531E-02
  d    1     2    1     3    6.40700E-01  4.99885E+00 -1.36508E+01
  d    1     2    1     4    1.19300E-01  3.70773E+00 -1.34615E+01
  d    1     2    1     5    4.46700E-02  3.31278E+00 -1.34473E+01
  d    1     2    1     6    2.20900E-02  3.13185E+00 -1.34455E+01
  d    1     2    1     7    1.27000E-02  3.03210E+00 -1.34452E+01
  d    1     2    1     8    8.03600E-03  2.97073E+00 -1.34451E+01
  d    1     2    1     9    5.42900E-03  2.93009E+00 -1.34451E+01
  d    1     2    1    10    3.85100E-03  2.90170E+00 -1.34451E+01
                 .    .    .
  d    2     1    2     2    2.61200E-01  9.49993E-01 -1.20976E+01
  d    2     1    2     2    5.23800E-01  9.44766E-01 -1.20976E+01
  d    2     1    2     3    4.90400E-02  8.03773E-01 -2.11374E+00
  d    2     1    2     3    1.02580E-01  8.02660E-01 -2.11374E+00
  d    2     1    2     4    1.78460E-02  7.62836E-01 -5.94416E-01
  d    2     1    2     4    3.78600E-02  7.62409E-01 -5.94416E-01
  d    2     1    2     5    2.78800E-02  7.45661E-01 -2.20190E-01
  d    2     1    2     6    1.55980E-02  7.36608E-01 -9.75172E-02
  d    2     1    2     7    9.62800E-03  7.31261E-01 -4.89059E-02
  d    2     1    2     8    6.36600E-03  7.27835E-01 -2.68788E-02
  d    2     1    2     9    4.43200E-03  7.25506E-01 -1.58470E-02
  d    2     1    2    10    3.21000E-03  7.23850E-01 -9.87535E-03
                 .    .    .
```

Notes for transitions:

1. Active levels are identified with (iso1,i1), (iso2,i2) labels
2. Two levels may be connected by multiple transitions

Notes for photoexcitations:

1. Transition energy / wavelength need not match difference in level energies
2. Additional data fields may contain matrix element values or other information

Identifying radiative transitions

How to identify transitions showing up in a calculated spectrum?

1. Identify likely charge states
2. Construct a sorted transition list
3. Find candidate transitions by energy + oscillator strength
4. (Optional) confirm by tweaking spectral calculation
   - details will be in lectures on rad transport and spectral simulations

Comments

- Overlapping and/or wide lines can make identifying transition energies difficult
- Lines will not shift due to continuum lowering but corrections are applied to some K-shell transitions to improve energies (and for plasma polarization)
- Lines will shift due to Doppler shifts and Stark effect

<u>All these effects (except corrections) can be minimized or turned off</u>

Constructing a sorted radiative transition list

CRETIN dca_36 radlist kr - or - switch(61) >0

transition list for model dca_36

photoexcitations

					de	lambda	fosc
					--	------	----
1 :	33	18	33	19	4.100E-03	3.024E+06	4.954E-04
2 :	33	19	33	21	4.100E-03	3.024E+06	2.442E-04
3 :	33	2	33	3	4.100E-03	3.024E+06	2.477E-04
4 :	33	25	33	26	4.100E-03	3.024E+06	2.477E-04
5 :	33	29	33	30	4.100E-03	3.024E+06	2.477E-04
6 :	33	15	33	16	4.100E-03	3.024E+06	2.477E-04
7 :	33	8	33	9	4.100E-03	3.024E+06	4.954E-04
8 :	33	10	33	12	4.100E-03	3.024E+06	2.477E-04
			. . .				
36502 :	3	1	3	10	1.298E+04	9.552E-01	2.221E-01
36503 :	4	4	4	19	1.298E+04	9.552E-01	4.382E-01
36504 :	4	3	4	18	1.298E+04	9.552E-01	4.382E-01
36505 :	4	2	4	17	1.298E+04	9.552E-01	4.382E-01
36506 :	4	1	4	16	1.298E+04	9.552E-01	3.756E-01
36507 :	3	4	3	13	1.305E+04	9.502E-01	5.080E-01
36508 :	3	3	3	12	1.305E+04	9.502E-01	5.080E-01
36509 :	3	2	3	11	1.305E+04	9.502E-01	5.080E-01
36510 :	3	1	3	10	1.305E+04	9.502E-01	4.445E-01
36511 :	2	4	2	13	1.305E+04	9.500E-01	1.306E-01
36512 :	2	3	2	12	1.305E+04	9.500E-01	1.306E-01
36513 :	2	1	2	2	1.305E+04	9.500E-01	2.612E-01
36514 :	2	2	2	11	1.305E+04	9.500E-01	1.143E-01
36515 :	2	5	2	14	1.305E+04	9.500E-01	1.306E-01
36516 :	2	4	2	13	1.312E+04	9.448E-01	2.619E-01
36517 :	2	3	2	12	1.312E+04	9.448E-01	2.619E-01
36518 :	2	1	2	2	1.312E+04	9.448E-01	5.238E-01
36519 :	2	2	2	11	1.312E+04	9.448E-01	2.292E-01
36520 :	2	5	2	14	1.312E+04	9.448E-01	2.619E-01
36521 :	1	1	1	2	1.349E+04	9.193E-01	4.162E-01
36522 :	35	17	35	31	1.405E+04	8.824E-01	1.341E-03
36523 :	35	16	35	30	1.405E+04	8.824E-01	1.341E-03

energy (eV)

λ (Å)

oscillator strength

active levels

Li-like satellites

He-α

Ly-α

n=1→2 transitions

inner shell

Constructing atomic models

Why look at atomic model construction?

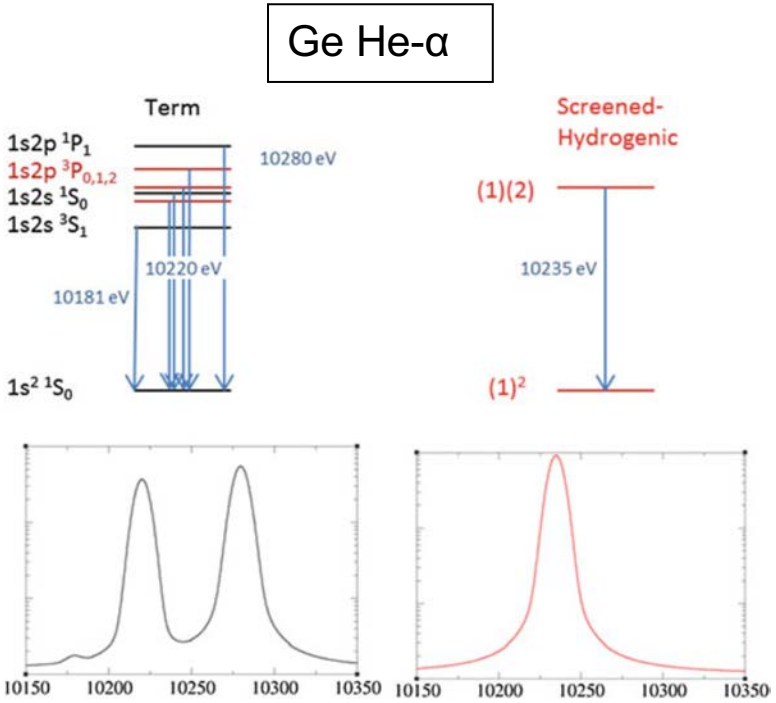
- Atomic structure matters
  - Level of detail
    - Fine structure for spectroscopy
    - Averaged structure for rad-hydro
  - State space coverage affects ionization balance
    - Low density: few excited states, high N for dielectronic recombination channels
    - High density: many excited states, max N decreases due to continuum lowering
- Models in /usr/gdata/dca were constructed to give decent results over a wide range of conditions in rad-hydro simulations (+ K-shell spectroscopy)

Cretin is agnostic about atomic models, but your physics may not be

Different conditions / purposes may require different atomic models

- Rad-hydro codes concentrate on moving around mass and energy
  - Often insensitive to details of absorption / emission spectra (fortunately!)
  - Usually use a small number of frequencies

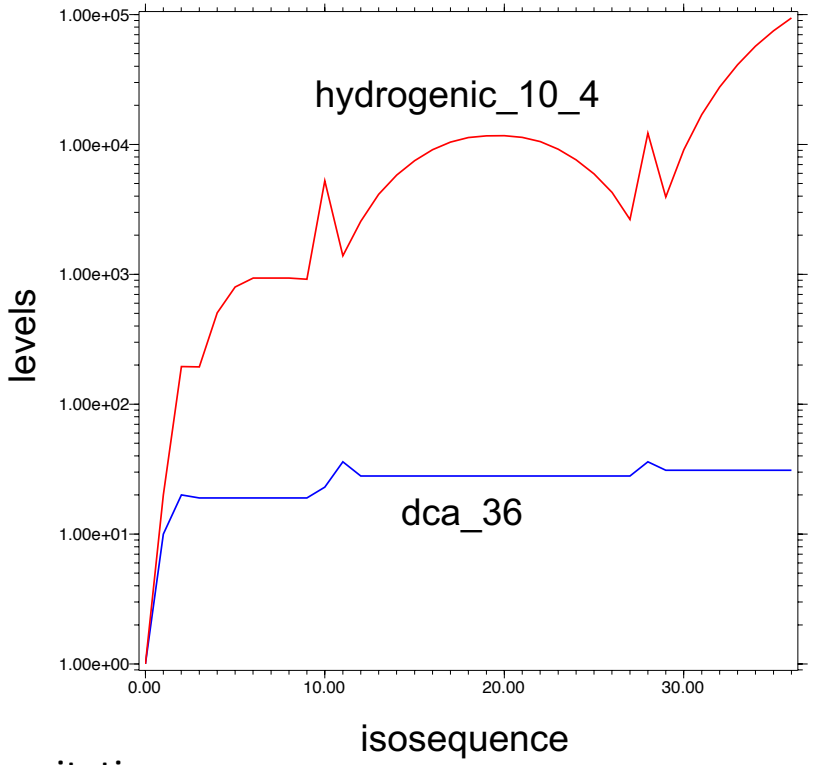
- Spectral simulations are done for fixed density / temperature distributions
  - Used to postprocess rad-hydro simulations
  - Details of the emerging intensity carry information about conditions
  - Level of detail can matter greatly
    e.g. shape of He-α complex



Constructing screened-hydrogenic atomic models

atoms hydrogenic ar

- Default structure:
  - All charge states
  - Excited states up to *n*=10 (PQNs only)
  - 2 excitations from valence shell, 1 from inner shells
- Options to change the structure
  - hydrogenic_*N*_*L* : excited states up to *n*=N split up to ℓ=L
  - hydrogenic(isomin:isomax)_*N*_*L* : include charge states within [isomin,isomax]
  - modeltype dca : use the structure of the latest dca_*xx* models
  - modeltype term : include term splitting of photoexcitations



Increasing the # of excitations or using (n,ℓ) to specify states increases size (and expense) exponentially

Screened-hydrogenic atomic model options

atoms hydrogenic ar
  modeltype *option*
  isorange *N L* | *all* $n_{x,inner}$ $\#N_{x1}$ $\#N_{x2}$ n_{mx1} n_{mx2} $n_{\Delta n=0}$ n_{split} nx nx_K nx_L

- "modeltype" *option* applies to the entire model
  option = dca , dca_v15 , term , fly , eonly , radonly
- "isorange" specifies various aspects of generated states for each charge state
- "isorange" values can be set for all (other) charge states with switches

$n_{x,inner}$	: switch(126)	max N for state w/ inner shell holes
$\#N_{x1}$, $\#N_{x2}$	: switch(127)	max # shells w/ inner shell holes
n_{mx1} , n_{mx2}	: switch(128)	max N for multiply excited states
$n_{\Delta n=0}$	: switch(130)	max N for $\Delta n=0$ transitions
n_{split}	: switch(160)	max N for term splitting
nx	: switch(187)	maximum # of excitations
nx_K , nx_L	: switch(187)	maximum # of excitations from K-, L-shells
	: switch(164)	min iso with >2 excitations (if ≠0)
	: switch(166)	min iso with >1 excitation from 1st inner shell (if ≠0)
	: switch(167)	max N for all but 1st excitation (if ≠0)

Constructing alternative atomic models

- Atomic data can come from a combination of
  - atomic physics codes: FAC, HULLAC, LANL codes
  - data compilations: SCRAM datasets
  - screened-hydrogenic data from Cretin
- Difficulties arise in
  - obtaining a complete set (atomic structure + all transition types)
  - constructing a consistent atomic structure (all states coupled to neighbors)
  - removing discontinuities in structures calculated with different approximations
- Other issues
  - averaging states and transitions to keep the size down
  - combining different parameterizations for transitions

Tools to help with this are available in /usr/apps/cretin/tools

Some useful command line options for atomic models

CRETIN *name* **hydrogenic(ISOMIN:ISOMAX)_N_L** *symbol* [**DCA**]

1. Build a screened-hydrogenic model for element *symbol*
2. Print the model to file *name.z01*
3. **DCA** : equivalent to "modeltype dca term"

CRETIN *model* **atoms** *symbol*

1. Produce a binary (PDB or HDF5) version of *model* as file *model.adf*

CRETIN *model* **reconst**

1. Produce an ascii version of (PDB or HDF5) *model* as file *model.z00*

CRETIN *model* **radlist** [*symbol*]

1. Produce a sorted list of radiative transitions in file *model.tbl*