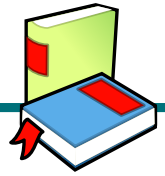
What is Yambo!?!



Theory&approximations

Many-Body Perturbation Theory

Time-dependent density functional theory

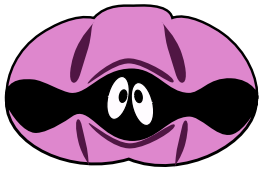
Bethe-Salpeter equation, GWA

AHC electron-phonon coupling...

Interfaces

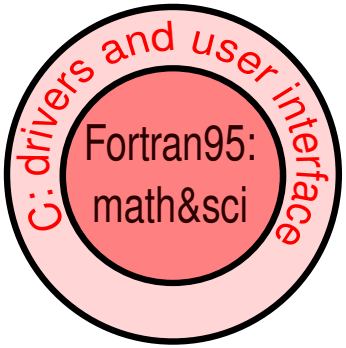
Planewave- Pseudopotential codes:

abinit.org PWscf



www.yambo-code.org

Languages



Libraries

MPI FFTW libxc NetCDF

ScaLAPACK etsf-io

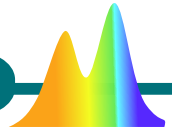
Combination of projects

Research in ab-initio treatment of excited-state systems

Andrea Marini, Conor Hogan, Myrta Grüning, Daniele Varsano, Comp. Phys. Comm. 180, 1392 (2009)

What does Yambo do?

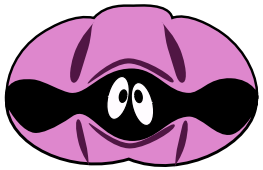
Properties



GPL:
Quasiparticles
Optical absorption
Electron energy loss
Dynamical polarizability

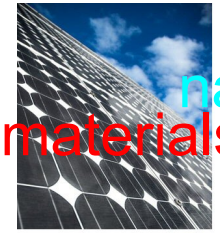
New in the GPL:
magneto-optical properties
electron-phonon
surface spectroscopy

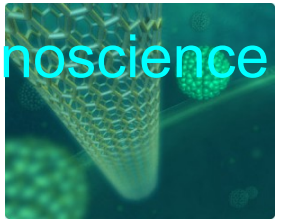
Developments:
Ultrafast spectroscopy
Yambo for HPC...

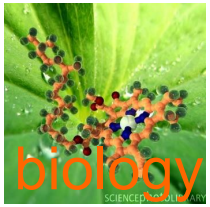


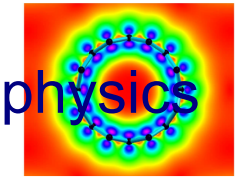
www.yambo-code.org

Applications





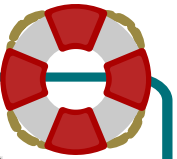


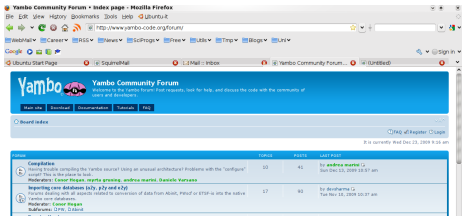


Community

Growing community of users using Yambo for forefront research publications

Support & reach-out





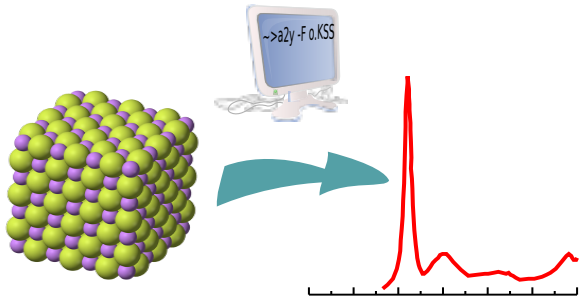
Dedicated users forum

Online documentation/tutorials

Andrea Marini, Conor Hogan, Myrta Grüning, Daniele Varsano, Comp. Phys. Comm. 180, 1392 (2009)

Step by step introduction to Yambo

You will learn:

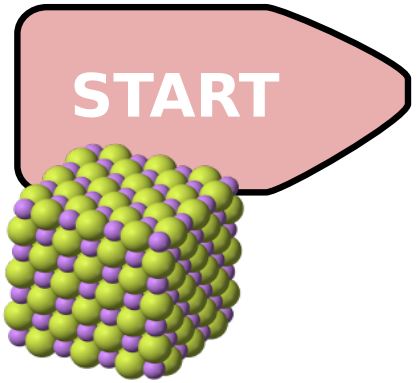


 # Flow of a Yambo calculation

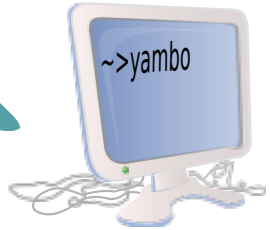
 # Yambo command line options


```
optics                      # [R OPT] Optics
chi                         # [R LR] Linear Response.
% QpntsRXd
  1 | 1 |                  # [Xd] Transferred momenta
%
% BndsRnXd
  1 |  10 |                # [Xd] Polarization function bands
%
NGsBlkXd= 1             RL  # [Xd] Response block size
% EnRngeXd
  7.50000 | 25.00000 |   eV  # [Xd] Energy range
%
% DmRngeXd
  0.10000 |  0.30000 |   eV  # [Xd] Damping range
%
ETStpsXd= 300               # [Xd] Total Energy steps
% LongDrXd
 1.000000 | 0.000000 | 0.000000 |     # [Xd] [cc] Electric Field
%
```

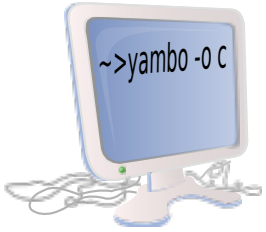
 # Yambo I/O files



1. Generate the core databases

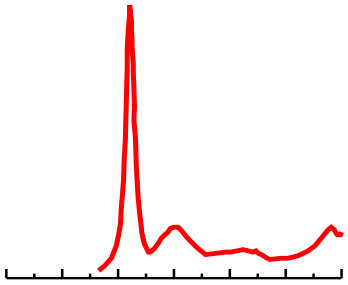


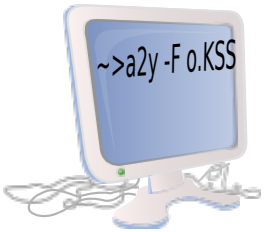
2. Run setup



3. Generate input

4. Run Yambo





1. Generate the core databases

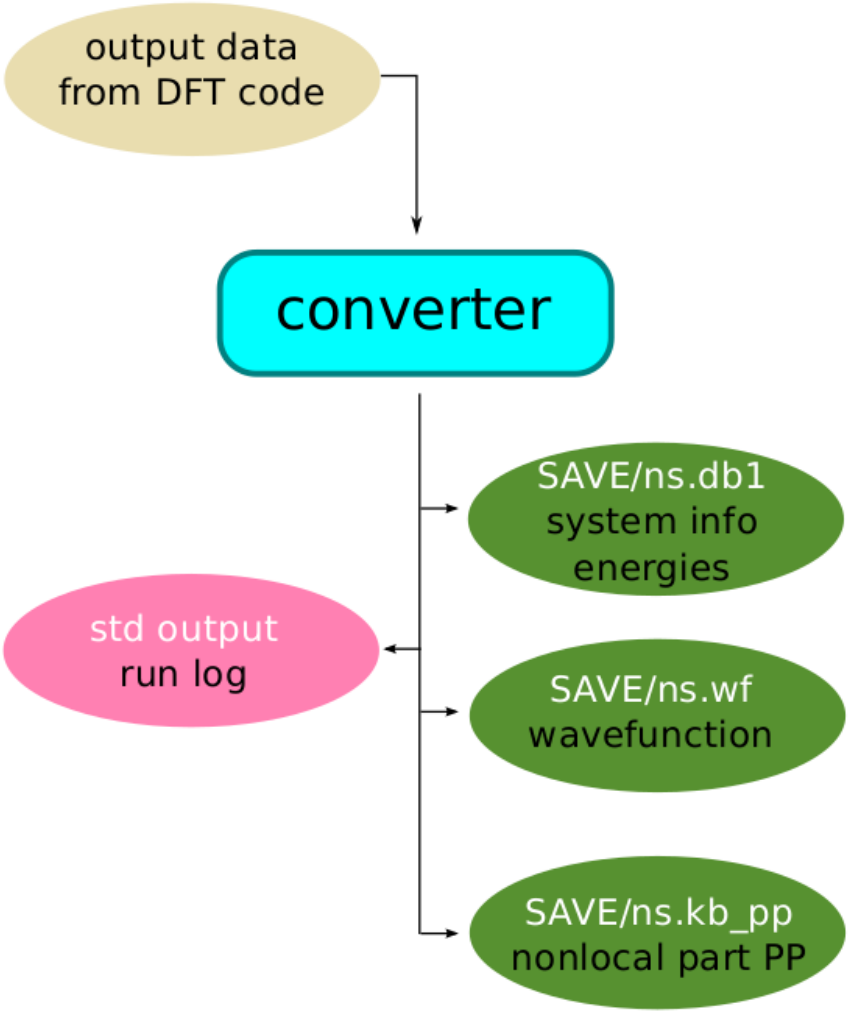
= convert data from standard ab initio DFT code (abinit, PWscf, ETSFio)

Use converters (a2y, p2y, e2y) from command line with options:

`-h`	Short Help
`-H`	Long Help
`-N`	Skip MPI initialization
`-F <opt>`	Input file name/prefix
`-O <opt>`	Output directory
`-S`	DataBases fragmentation
`-a <real>`	Lattice constants factor
`-d`	States duplication (artificial spin polarization)
`-t`	Force use of spatial Inv. instead of Time Rev.
`-y`	Force no symmetries
`-w`	Force no wavefunctions

1. Generate the core databases

case study #1: convert PWscf GS file with a2y



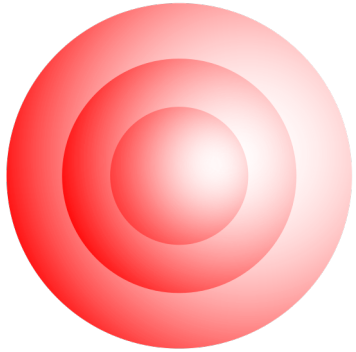

```
% ls
charge-density.dat  gvectors.dat  K00002/  K00004/  K00006/  K00008/  K00010/
data-file.xml       K00001/       K00003/  K00005/  K00007/  K00009/  Si.vbc.UPF
```

```
>p2y -F data-file.xml


 _|      _|   _|_|   _|      _|  _|_|_|      _|_|
   _|  _|     _|     _|_|  _|_|  _|     _|  _|    _|
     _|       _|_|_| _|  _|  _|  _|_|_|_|  _|    _|
   _|  _|     _|     _|      _|  _|        _|    _|
 _|      _|   _|_|_| _|      _|  _|_|_|    _|_|


<---> DBs path set to .
<---> Index file set to data-file.xml
<---> Header/K-points/Energies... done
<---> Cell data... done
<---> Atomic data... done
<---> Symmetries...[SI no]...[I yes]...[-I no]...[TR yes]
<---> XC functional...Slater exchange(X)+Perdew & Zunger(C)
<---> EXX fraction... 0.000000
<---> EXX screening... 0.000000
<---> K-points mesh... done
<---> RL vectors... done
<---> IGK arrays... done
<---> Energies... done
<--->  :: Electrons                : 8.000000
<--->  :: Temperature      [ev]: 0.000000
<--->  :: Lattice factors [a.u.]: 5.091500  5.091500  5.091500
<--->  :: K-points                 : 3
<--->  :: Bands                    :  50
<--->  :: Spinor components        : 1
<--->  :: Spin polarizations       : 1
<--->  :: Spin orbit coupling      : no
<--->  :: Symmetries   [spatial]: 24
<--->  ::                [T-rev]: yes
<--->  :: Max WF components        : 266
<--->  :: RL vectors         (WF):  369
<--->  :: RL vectors     (CHARGE): 2085
<--->  :: XC potential             : Slater exchange(X)+Perdew & Zunger(C)
<--->  :: Atomic species           : 1
<--->  :: Max atoms/species        : 2
<---> == DB1 (Gvecs and more) ...
<---> ... Database done
<---> == DB2 (wavefunctions)  ... done ==
<---> == DB3 (PseudoPotential) ... done ==
<--->  == P2Y completed ==
```

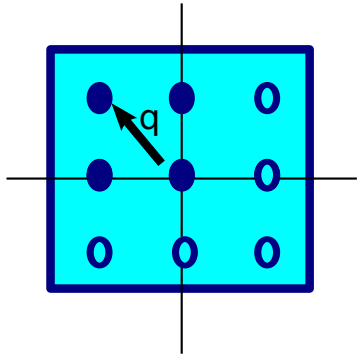
```
% ls SAVE
ns.db1      ns.kb_pp      ns.wf
```

2. Run setup

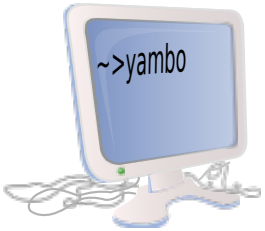
= prepare general purpose databases for later use



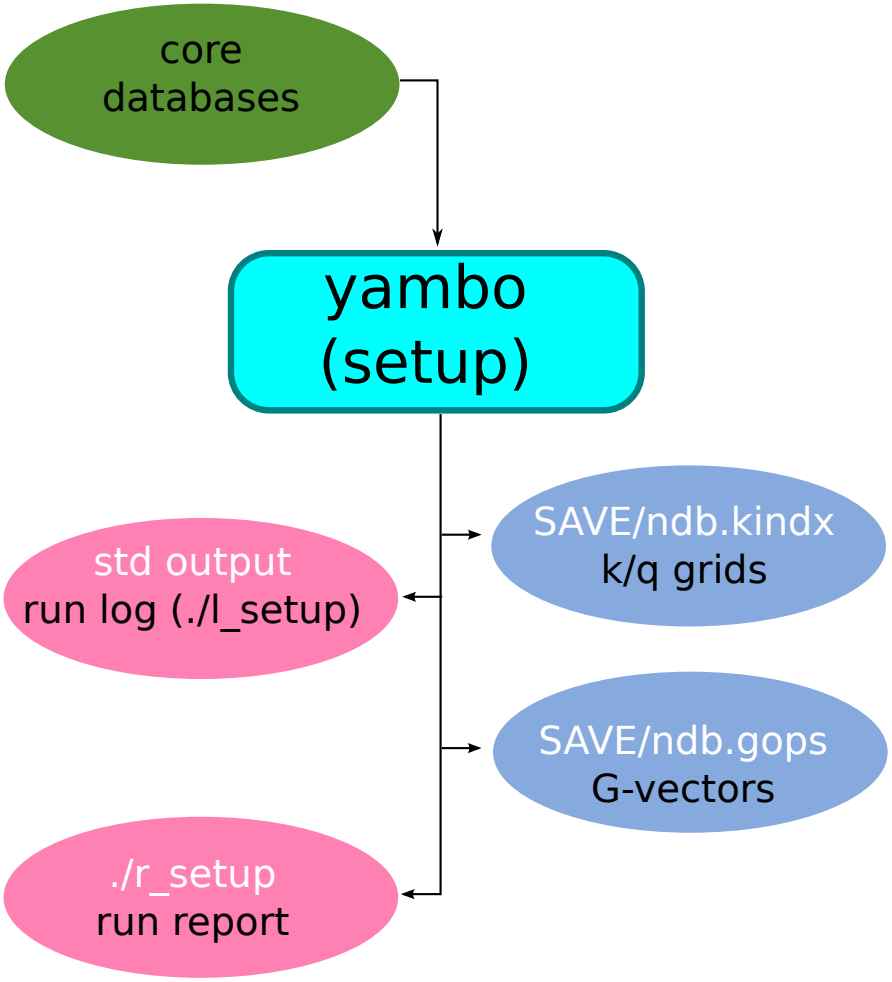
* **Data initialization:**
reorders G-vectors into spherical shells
calculates Fermi level and electronic occupations
sets up energy grids



* **Brillouin-zone sampling:**
expands k-points to full BZ
generates q-point meshes
checks on uniformity of grids



2. Run setup

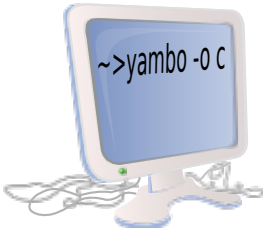


case study #2: setup run

```
% ls SAVE/
ns.db1  ns.kb_pp  ns.wf
```

```
% yambo

 <---> [01] Files & I/O Directories
 <---> [02] CORE Variables Setup
 <---> [02.01] Unit cells
 <---> [02.02] Symmetries
 <---> [02.03] RL shells
 <---> Shells finder |####################| [100%] --(E) --(X)
 <---> [02.04] K-grid lattice
 <---> [02.05] Energies [ev] & Occupations
 <---> [03] Transferred momenta grid
 <---> X indexes |####################| [100%] --(E) --(X)
 <---> SE indexes |####################| [100%] --(E) --(X)
 <---> [04] Game Over & Game summary
```

```
% ls
r_setup  SAVE
% ls SAVE
ndb.gops  ndb.kindx  ns.db1  ns.kb_pp  ns.wf
```

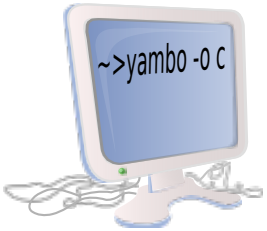


3. Generate input file

= select runlevel(s) and choose parameter values

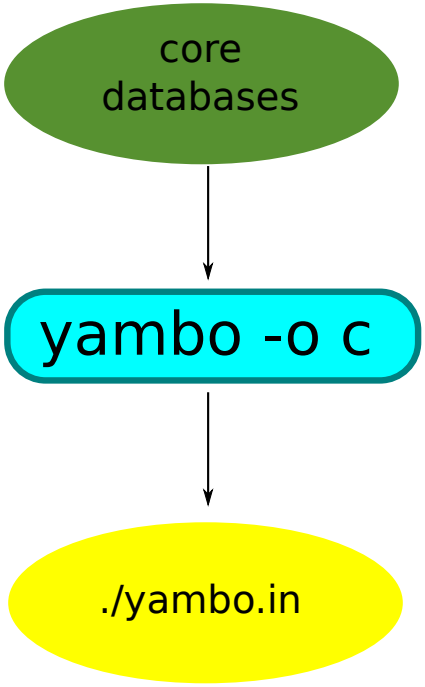
yambo acts as input file generator
when we add command line options

Calculation type	**Command line**
Common:	
Setup	yambo -i
Coulomb interaction	yambo -c
QP corrections:	
Dynamical screened interaction	yambo -d
Hartree-Fock self-energy	yambo -x
GW	yambo -g [solver] -p [approximation]
Optics:	
Optics in G-space	yambo -o c -k [kernel]
Optics in eh-space + solvers	yambo -o b -k [kernel] -y [solver]
Static screened interaction	yambo -b



3. Generate input file

case study #3: G-space optics (rpa level)

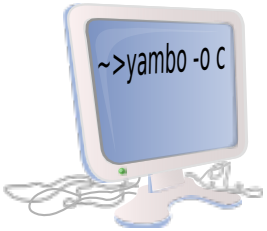


directly open in editor

```
optics                  # [R OPT] Optics
chi                     # [R CHI] Dyson equation for Chi.
Chimod= "IP"            # [X] IP/Hartree/ALDA/LRC
% QpntsRXd
  1 | 19 |              # [Xd] Transferred momenta
%
% BndsRnXd
  1 | 10 |              # [Xd] Polarization function bands
%
% EnRngeXd
  0.00000 | 10.00000 |   eV  # [Xd] Energy range
%
% DmRngeXd
  0.10000 |  0.10000 |   eV  # [Xd] Damping range
%
ETStpsXd= 100             # [Xd] Total Energy steps
% LongDrXd
 1.000000 | 0.000000 | 0.000000 |      # [Xd] [cc] Electric Field
%
```

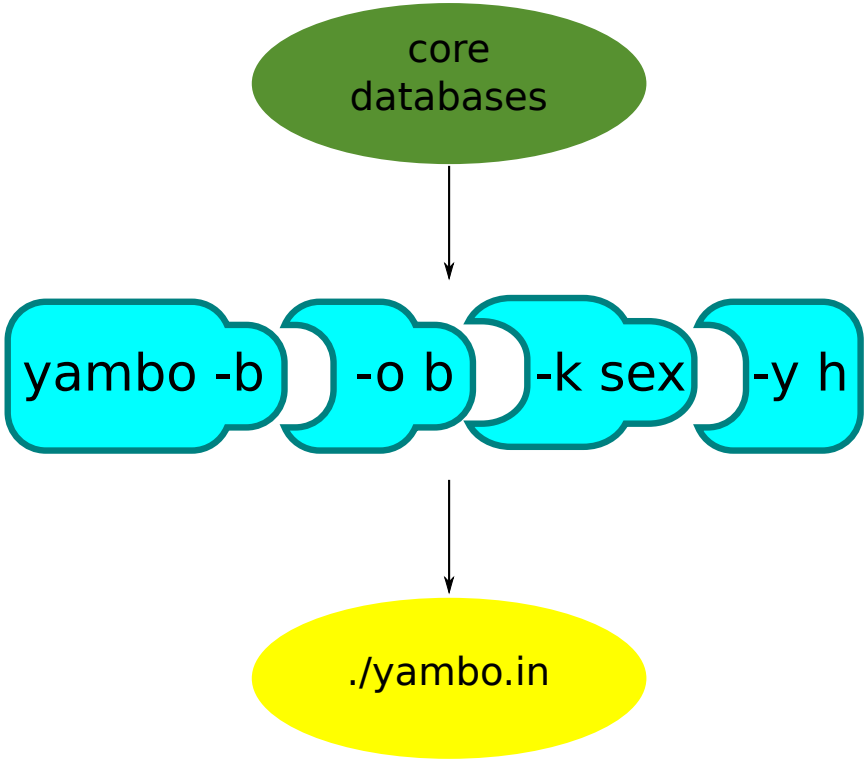
runlevels

default values from existing databases (compatibility)



3. Generate input file

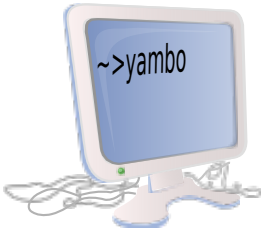
case study #4: eh-space optics (BSE kernel)



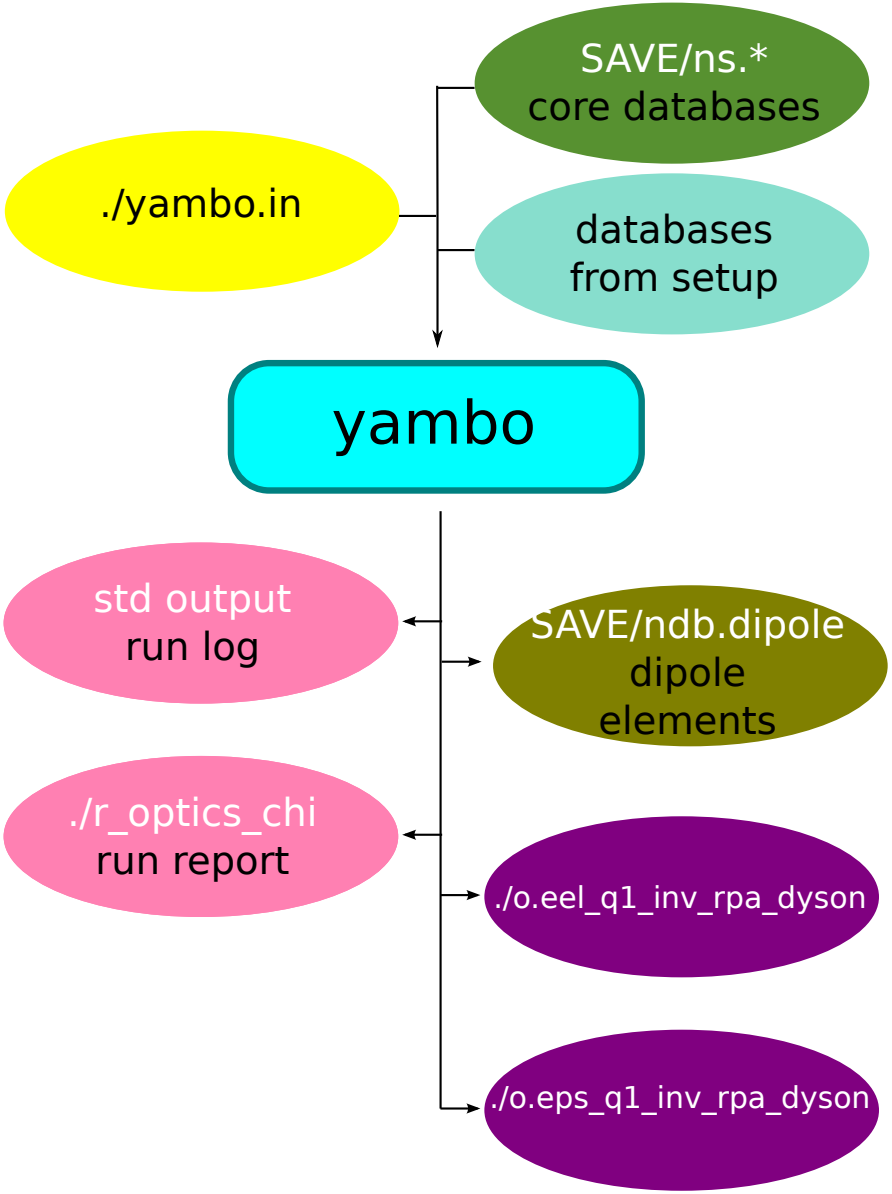
combine runlevels

-b : static inverse dielectric matrix
-o b : optics in eh-space
-k w : Bethe-Salpeter kernel
-y h : iterative diag solver (Haydock)

```
em1s                    # [R Xs] Static Inverse Dielectric Matrix
optics                  # [R OPT] Optics
bse                     # [R BSE] Bethe Salpeter Equation.
bss                     # [R BSS] Bethe Salpeter Equation solver
Chimod= "Hartree"           # [X] IP/Hartree/ALDA/LRC/BSfxc
BSEmod= "causal"            # [BSE] resonant/causal/coupling
BSKmod= "SEX"               # [BSE] IP/Hartree/HF/ALDA/SEX/BSfxc
BSEmode= "causal"  # [BSE] resonant/causal/coupling
BSENGexx= 1885 RL # [BSK] Exchange components
BSENGBlk= 1       RL # [BSK] Screened interaction block size
% BSEBands
 1 | 10 |                # [BSK] Bands range
%
% QpntsRXs
 1 | 19 |                # [Xs] Transferred momenta
%
% BndsRnXs
 1 | 10 |                # [Xs] Polarization function bands
%
NGsBlkXs= 1             RL  # [Xs] Response block size
% LongDrXs
 1.000000 | 0.000000 | 0.000000 |      # [Xs] [cc] Electric Field
%
BSSmod= "h"                 # [BSS] Solvers `h/d/(p/f)i/t`
% BEnRange
  0.00000 | 10.00000 |   eV  # [BSS] Energy range
%
% BDmRange
  0.10000 |  0.10000 |   eV  # [BSS] Damping range
%
BEnSteps= 100               # [BSS] Energy steps
% BLongDir
 1.000000 | 0.000000 | 0.000000 |      # [BSS] [cc] Electric Field
%
```



4. Run yambo



case study #5: G-space optics (RPA)

```
% ls
r_setup  SAVE  yambo.in
```

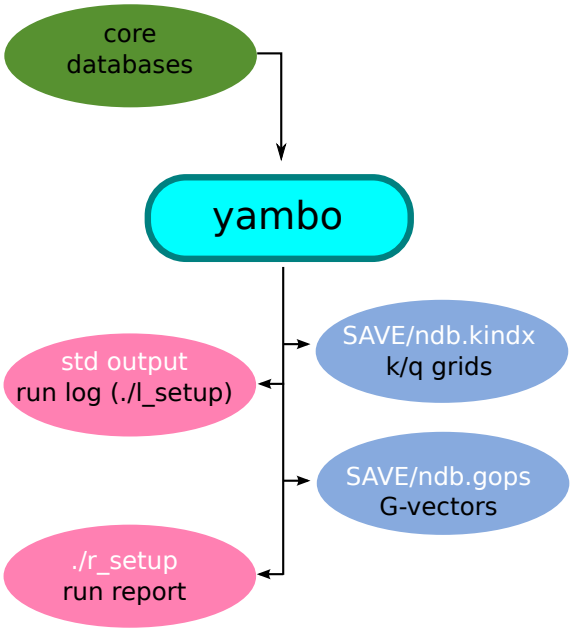
```
% yambo

 <---> [01] Files & I/O Directories
 <---> [02] CORE Variables Setup
 <---> [02.01] Unit cells
 <---> [02.02] Symmetries
 <---> [02.03] RL shells
 <---> [02.04] K-grid lattice
 <---> [02.05] Energies [ev] & Occupations
 <---> [03] Transferred momenta grid
 <---> [04] Optics
 <---> [WF-Oscillators/G space loader] Wfs (re)loading [100%] --(E) --(X)
 <---> Dipole (T): |####################| [100%] --(E) --(X)
 <---> [X-CG] R(p) Tot o/o(of R)  :   222   6144    100
 <---> [FFT-X] Mesh size: 18  18  18
 <---> [WF-X loader] Wfs (re)loading |#####| [100%] --(E) --(X)
 <---> [X] Upper matrix triangle filled
 <---> Xo@q[1] 1-100 |####################| [100%] --(E) --(X)
 <---> X @q[1] 1-100 |####################| [100%] --(E) --(X)
 <---> [05] Game Over & Game summary
```

```
% ls
o.eel_q1_inv_rpa_dyson  o.eps_q1_inv_rpa_dyson
r_optics_chi  r_setup  SAVE  yambo.in
```

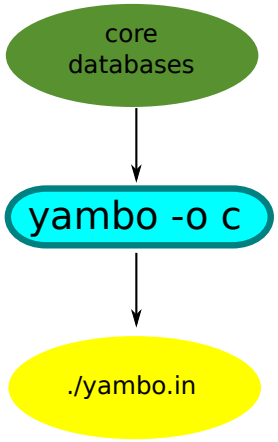
```
% ls SAVE
ndb.gops  ndb.kindx  ndb.dipole  ns.db1  ns.kb_pp  ns.wf
```

To remember: one source many execs

run setup

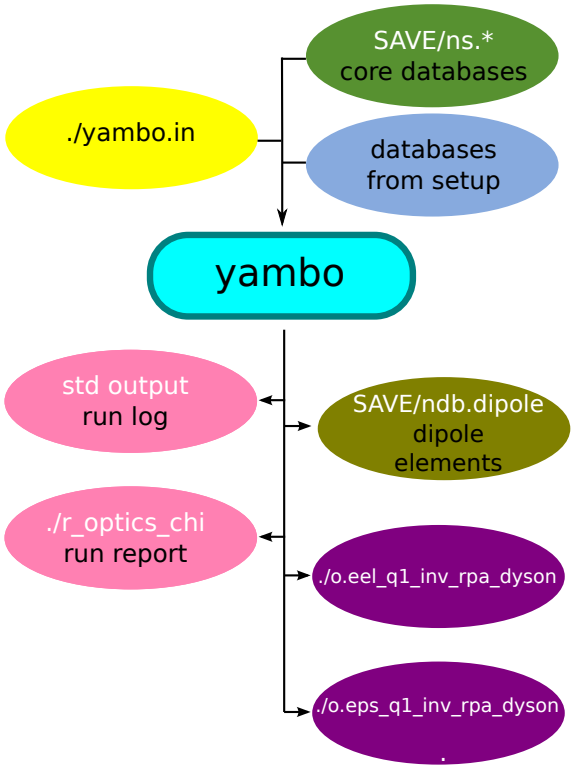


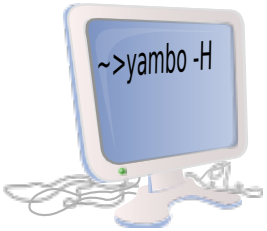
generate input



select runlevel(s)

run specific task(s)





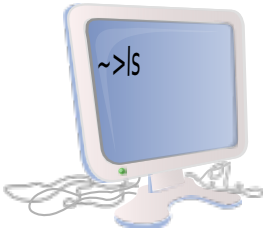
To remember: command line options

`-h`	Short Help
`-H`	Long Help
`-J <opt>`	Job string identifier
`-V <int>`	Input file verbosity
`-F <opt>`	Input file
`-I <opt>`	Core I/O directory
`-O <opt>`	Additional I/O directory
`-C <opt>`	Communications I/O directory
`-N`	Skip MPI initialization
`-D`	DataBases properties
`-S`	DataBases fragmentation
`-i`	Initialization
`-o <opt>`	Optics [opt=(c)hi is (G)-space / (b)se is (eh)-space]
`-k <opt>`	Kernel [opt=ip/tdh/tdlda/tdhf/w]
`-y <opt>`	BSE solver [opt=h/d/(p/f)i/t)]
`-c`	Coulomb interaction
`-x`	Hartree-Fock Self-energy and Vxc
`-d`	Dynamical Inverse Dielectric Matrix
`-b`	Static Inverse Dielectric Matrix
`-p <opt>`	GW approximations [opt=(p)PA /(c)HOSEX]
`-g <opt>`	Dyson Equation solver [opt=(n)ewton/(s)ecant]

display list of options

uppercase options added at run-time to specify I/O, MPI initialization, etc.

lowercase options drive input file generator



To remember: yambo I/O files

generated by the converters
geometry, basis set,
energies, wavefunctions

generated during setup run
k/q grids, G-vectors

generated at run-time
info specific to a runlevel

SAVE/ns.*
core databases

SAVE/ndb.*
stable databases

SAVE/ndb.*
job-dependent
databases

yambo

./yambo.in

./o.*
output files

./l_* ./r_*
run-logs and
reports

generated by yambo+command line options
select runlevel(s)
parameter values

generated at the end of a runlevel
for human use:
e.g. plot or inspection

ENJOY!

BUT RUN RESPONSIBLY