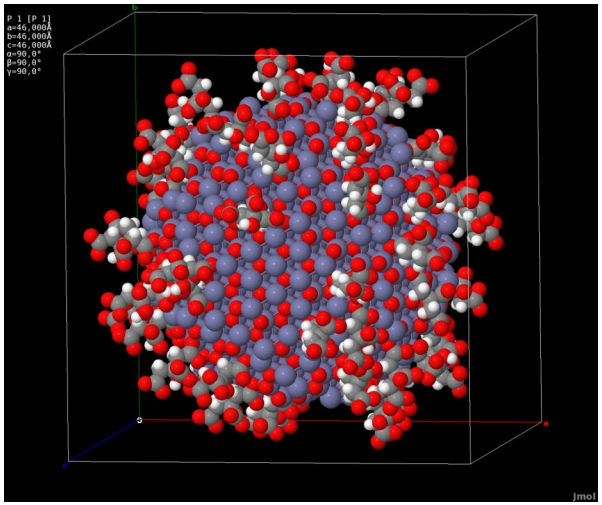
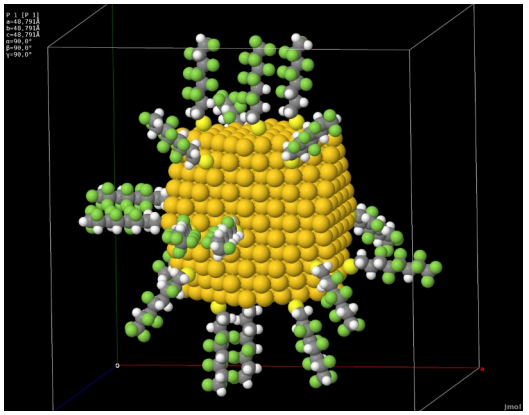
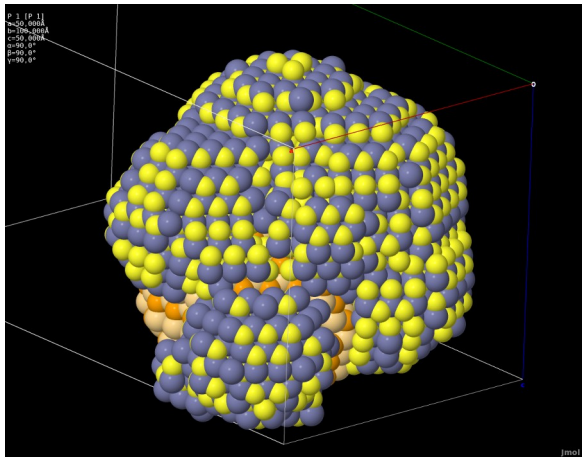
tutorial session X
parallel
(Nanoparticle)
refinement



ZnO
with organic ligand



Gold cuboctahedron
with organic ligands



CdSe / ZnS
core / shell

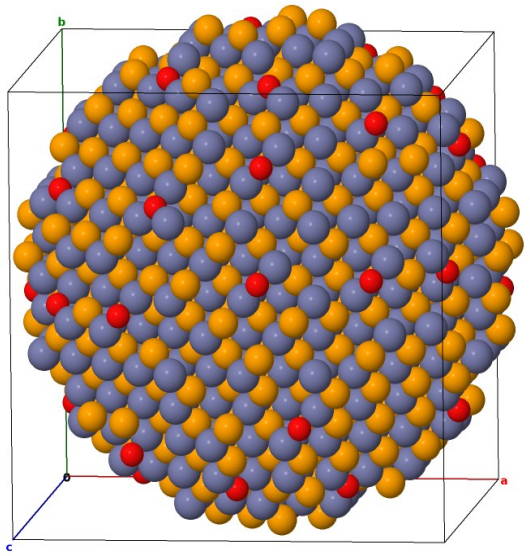
Nanoparticle refinement

Goal: ellipsoidal ZnSe nanoparticle with
Stacking faults
Refine against experimental PDF

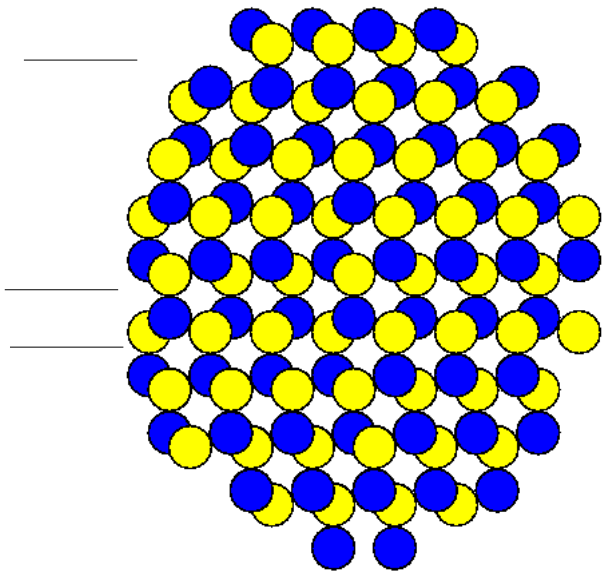
Concept: DIFFEV:
Define population
Define parameters and allowed range
Loop over generations

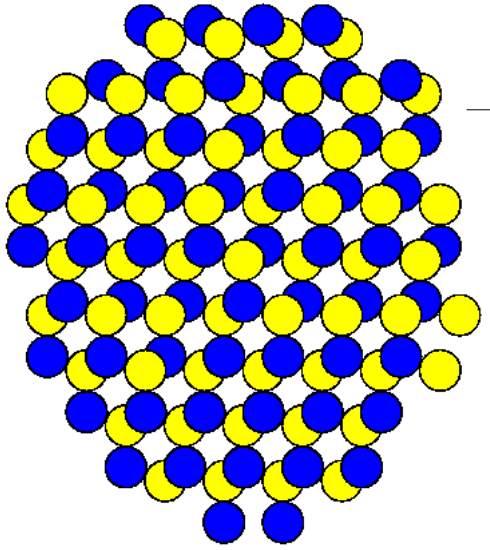
DISCUS
Build an ellipsoidal NP
Calculate PDF

KUPLOT
Average
Shape corrections
Calculate R-value



Aspects of PDF calculation for small nanoparticles

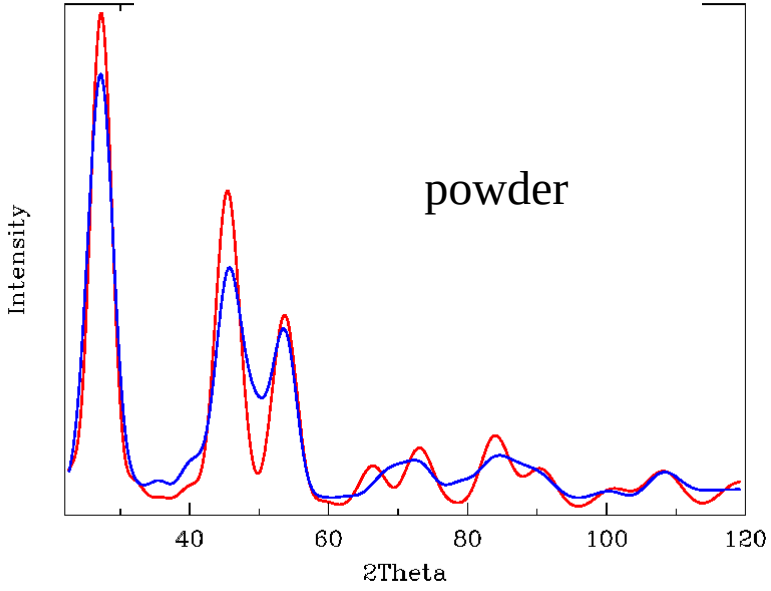


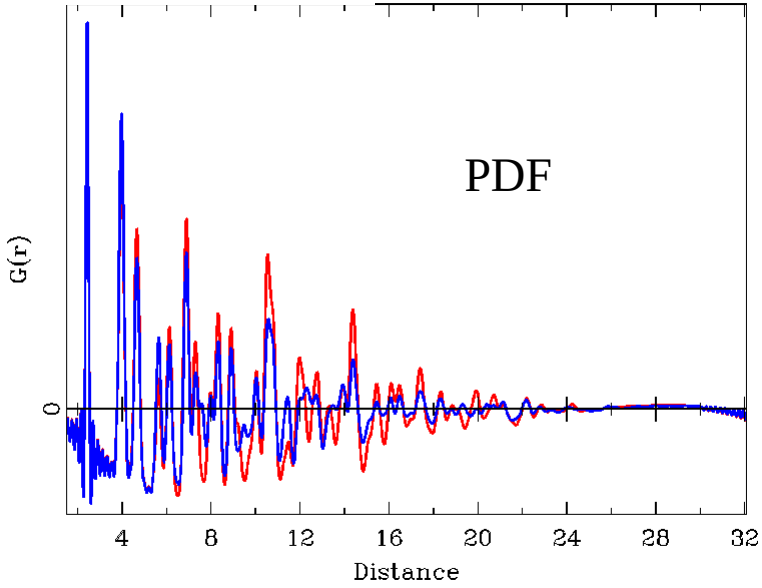


10 layers
simulated with
identical parameters

individual location
of stacking faults

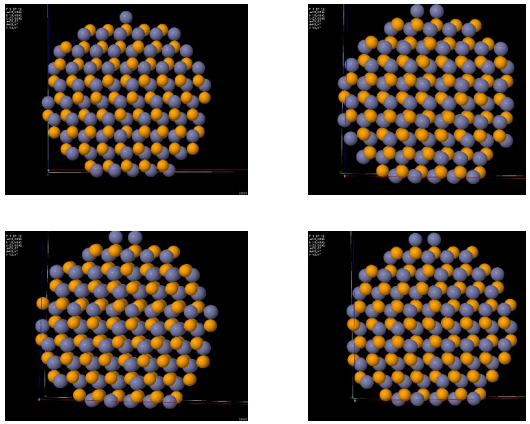
each particle is not
representative
==> need to average

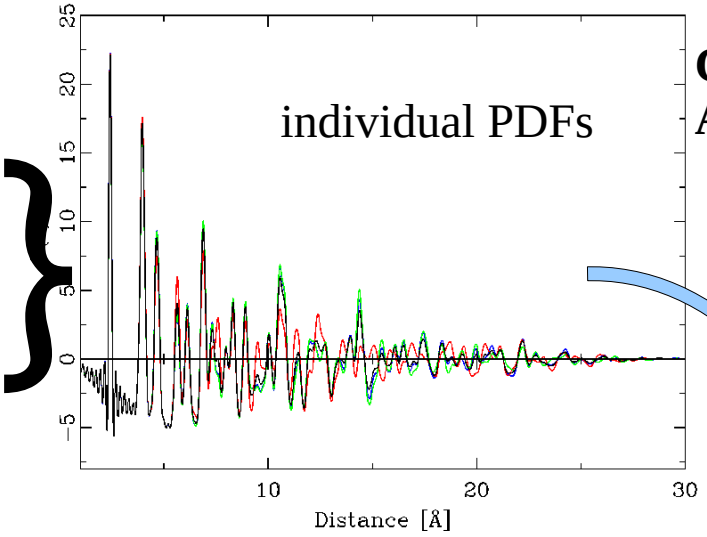




Bottom-Up Simulation and Refinement

Ensemble modelling

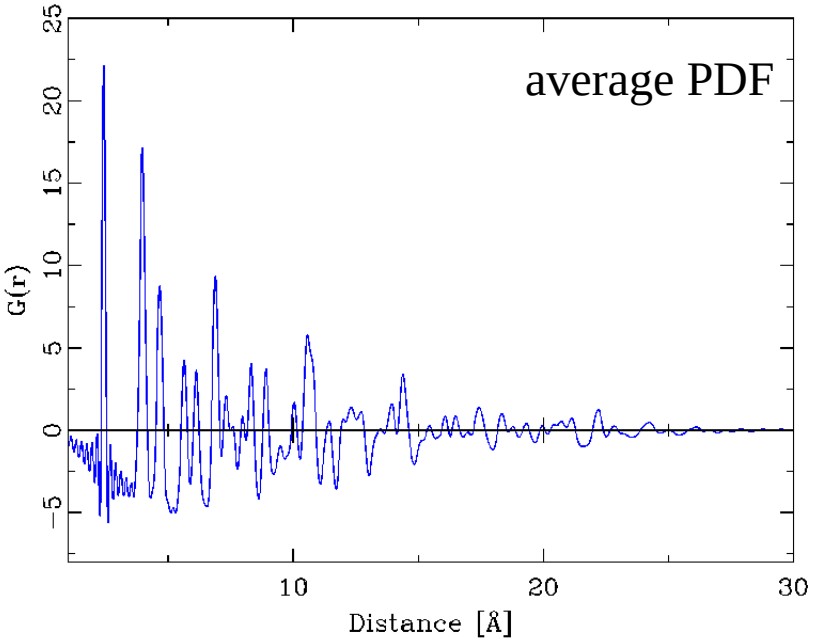




Calculate (many) individual nanos
Average PDF / powder pattern

coordinates in asymmetric unit
symmetry
diameter
defects

2 lattice parameters
1 coordinate
1 ADP
2 radii
1 probability

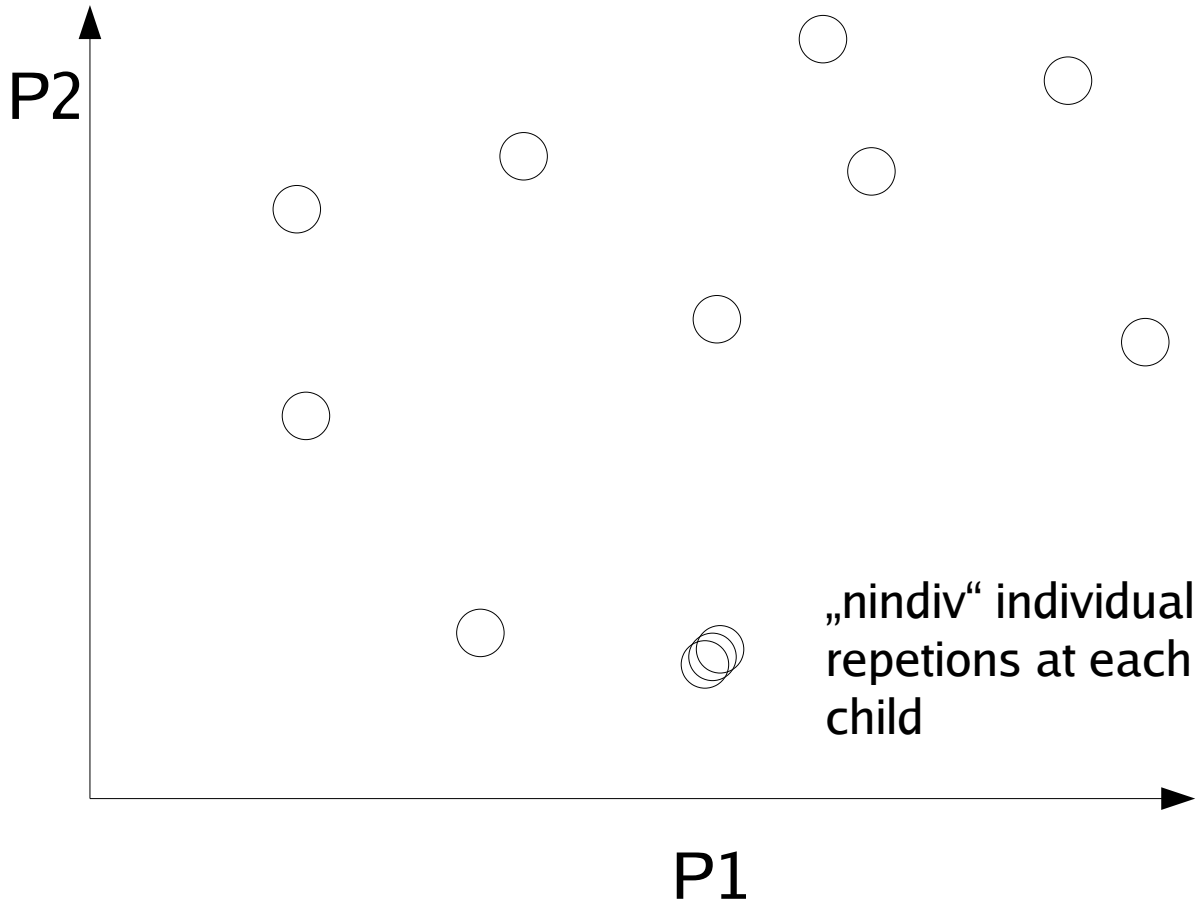


BUT

incoherent average of PDFs
requires evolutionary refinement
no Least-Squares

expensive

Niederdraenk et al. Phys.stat.sol c (2007);
Niederdraenk et al. PCCP (2011)



$y = P_1x + P_2$

N sets of parameters
(P_1, P_2)

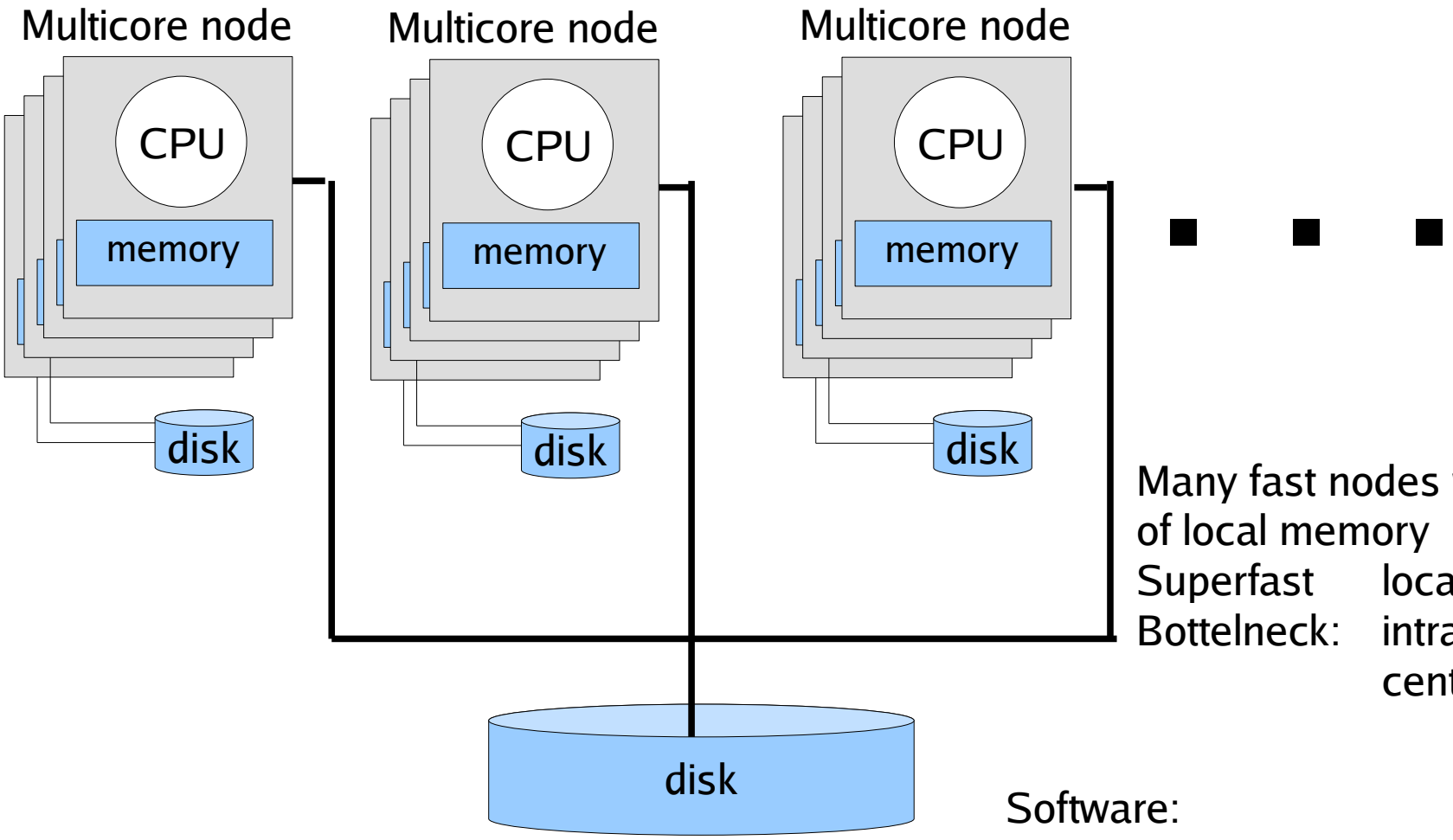
R-values for each set

Each parameter vector
Represents a nanoparticle
With:
Lattice parameters a,b,c,…
Atom positions…
Shape…
Defects...

Diffev calls these
„children, kids“

Easily some 200 children X 30 individuals = 6000 nanoparticle per generation

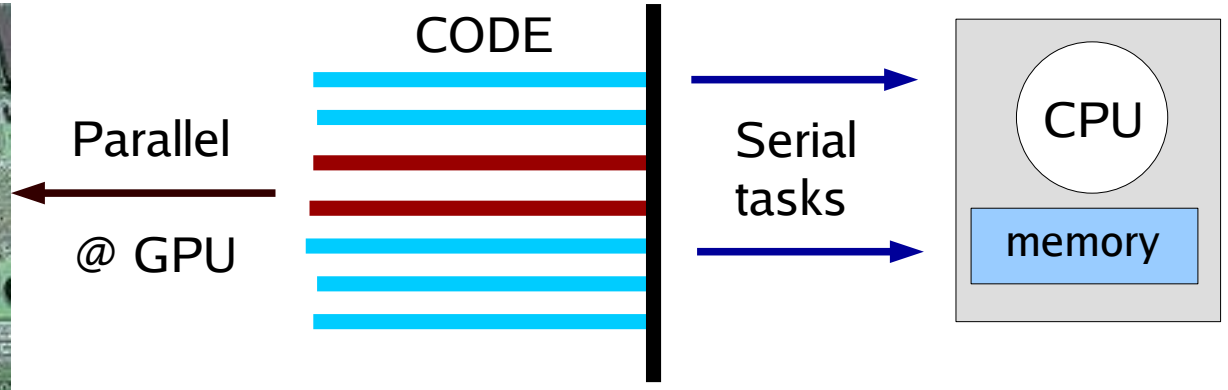
Parallel refinement, Architecture



Many fast nodes with lots of local memory
Superfast local disk
Bottelneck: intranet to central disk

Emmy Cluster RRZE, Erlangen
500 nodes, each with 10 cores

Software:
MPI Message Passing Interface
OpenMP
and combinations
hybrid programming



GPU

Thousands of cores
little local memory

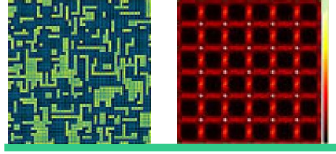
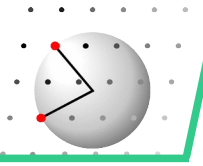
Software:

Proprietary CUDA
OpenCL
GPUOpen
???

Message passing Interface

```
Easy installation on Linux, Cygwin ! Install like any other package
#
mpiexec -n 5 discus_suite -macro suite.mac ! discus_suite started indirectly
      ! -n 5 : run 5 tasks in parallel
```

Required changes in ZNSE macros: None !

The essential refinement macro

```
@cleanup.mac          ! discard previous results !!!!!
#
@setup.mac            ! define number of individual repetitions
@diffev_setup.mac     ! define all diffev essentials
#
init silent           ! create the first generation
#
do i[0]=1,500         ! run for a fixed number of cycles/generations
   echo „Generation %5d“,pop_gen[0]  ! keep user informed
   run_mpi discus, nano.znse.mac, nindiv, LOGFILES/d ! Do magic
   compare silent     ! Compare parent and child generation
enddo                 ! End of loop indicator
enddo                 ! Return to suite
```