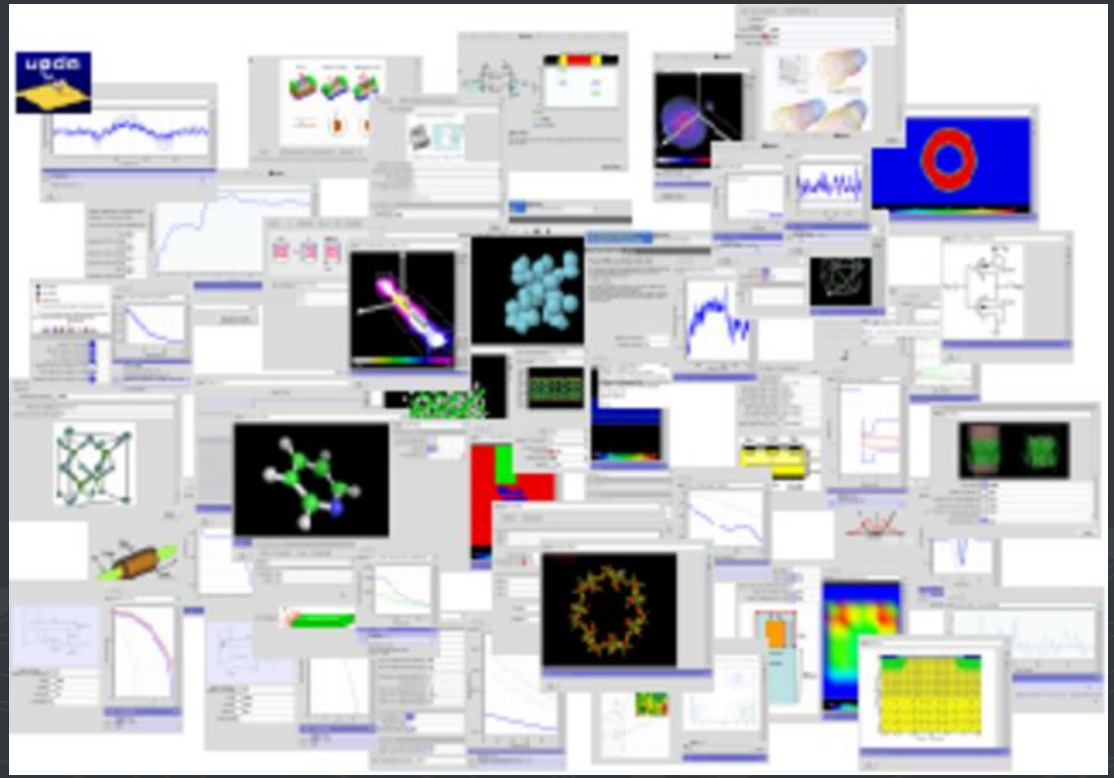
Modeling Uncertainty with nanoHUB tools using BOINC as a Computational Resource

Steven Clark, Martin Hunt, Ben Haley
nanoHUB.org
HUBzero.org
Research Computing, Purdue University

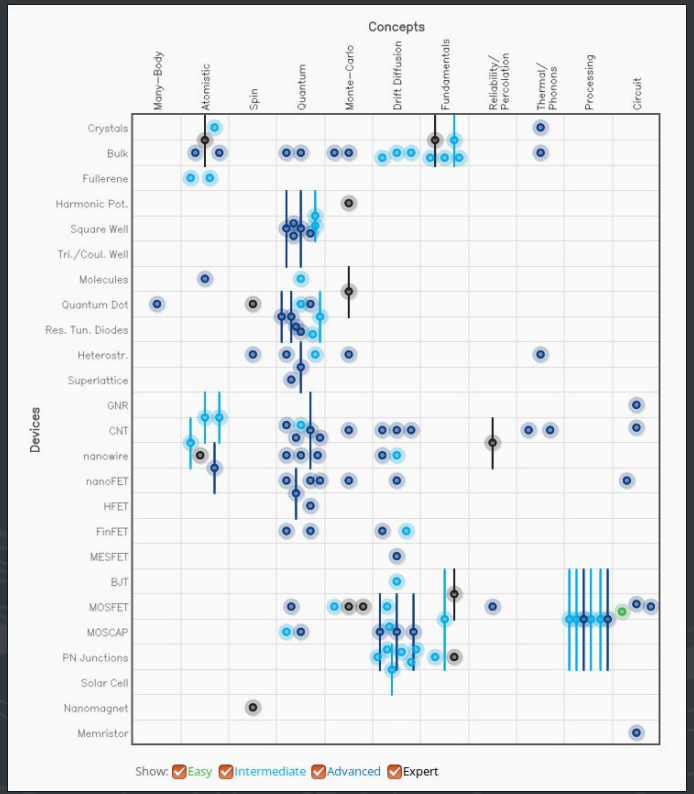
nanoHUB

- Collaborate
- Simulate
- Explore
- Learn



PLATFORM FOR COLLABORATIVE SCIENTIFIC COMPUTATION

- 1.44 million users annually
- 511 published tools
- 1.41 million simulations
- Varying degrees of skill required
- Wide range of topics covered
- Diagram highlights tools related to nanoelectronics, concepts vs devices



PLATFORM FOR COLLABORATIVE SCIENTIFIC COMPUTATION

- User perspective
  - Production level code
  - Powerful computing resources
  - No downloading, no compiling, ...
  - Automatically runs most updated version
  - Access regardless of location
- Developers perspective
  - GUI development environments
    - RAPPTURE
    - Jupyter Notebooks
  - Source code management - subversion/GIT
  - Rich development platform
  - Powerful computing resources

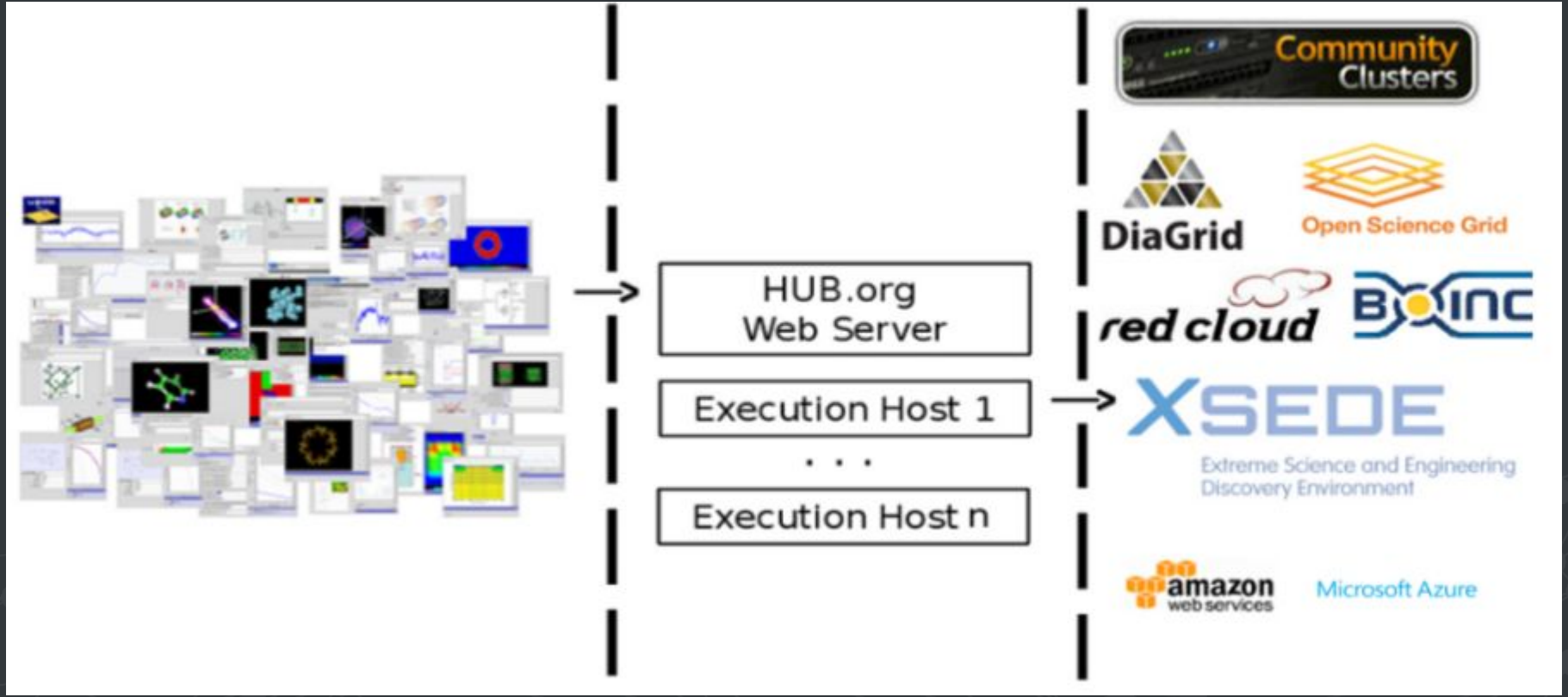
SIMULATION USE CASES

- **On demand**
  - UI used to declare inputs for simulation
  - Command line
  - Single simulation or parametric sweep
- **Cache resolution**
  - Input (driver.xml) files are placed in a cache queue
  - External process pulls input from cache queue, does the simulation, saves the result
  - If cache result exists no simulation is required simply pull the existing result
  - Faster response time provides better user experience
  - > 100,000 jobs processed

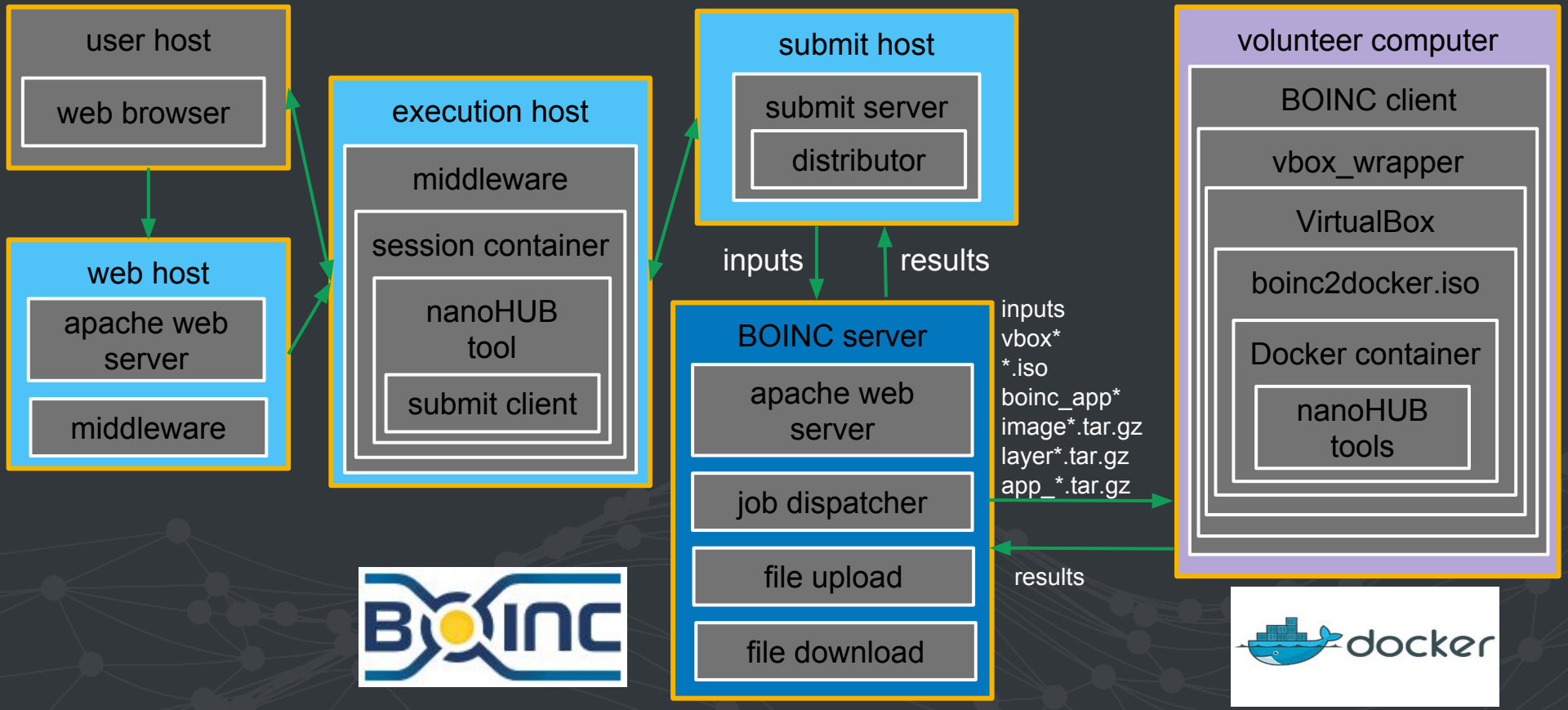
SIMULATION USE CASES

- **Uncertainty quantification**
  - Inputs declared as distributions
  - Statistical methods used to determine input samples
  - A simulation is run for each sample
  - Results include a response surface model which can be used to approximate simulation, sensitivity analysis, and probability distribution function (PDF) for outputs.
- **Exploratory simulation**
  - Allow for interactive selection of multidimensional input space
  - Automatically generate simulation input samples covering the space
  - Execute simulation for each sample

SUBMITTING JOBS TO FOREIGN RESOURCES



SUBMIT ARCHITECTURE - nanoHUB/BOINC



SUBMIT/BOINC - INTEGRATION

- **nanoHUB Application**
  - UI
  - User supplied data
  - Application files
- **submit Server**
  - Common interface between local and all remote resources (HPC + HTC)
- **BOINC Server**
  - Job execution manager for all BOINC submissions - submit_api batch submission
  - stage_docker_image - combines docker save and stage_file to place tarballs in the download directory. Also creates nanoHUB specific VirtualBox initialization file and boinc_app startup script.

SUBMIT/BOINC - INTEGRATION

- **Volunteer Host**
  - VirtualBox - nanoHUB applications run in Linux environment. VirtualBox provides access to Windows and MAC volunteer hosts.
  - boinc2docker - introduction of docker containers allows simpler change management. Nine docker containers support 282 nanoHUB applications.
  - Mounted volumes - allow for reduced memory requirement when loading docker container and greater flexibility creating new applications.
  - nanoHUB application files sent as tar balls and are not removed at job completion to reduce bandwidth requirement
  - User supplied data is also sent as a tarball but is job specific and is removed at job completion

SUBMIT SIMPLE SIMULATION

- Execution on local host

```
$ mpirun -n 16 namd2-2.9 prog.namd
```

- Execution on foreign host

```
$ submit --nCpus 16 --wallTime 10 \
         --inputfile par_all27_prot_lipid.inp \
         --inputfile ubq_ws.pdb \
         --inputfile ubq_ws.psf \
         namd2-2.9 prog.namd
```

- This command will request sixteen cores for ten minutes to run namd2-2.9 where it is installed
- Submit deduces that the file prog.namd needs to be transferred for the job to run
- Additional files that need to be transferred are specified by additional command line arguments
- Upon job completion all files created or modified by the job will be returned to the user

SUBMIT PARAMETRIC SWEEPS

- Parametric sweeps via single command

```
submit --parameters @@cap=10pf,100pf,1uf sim.exe @:indeck
submit --parameters @@vth=0:0.2:5 --parameters @@cap=10pf,100pf,1uf sim.exe @:indeck
submit --parameters params sim.exe @:indeck

submit --data input.csv sim.exe --voltage @@volts --input @:infile

submit --parameters @@num=1:1000 sim.exe input@@num
submit --parameters @@file=glob:indeck* sim.exe @@file
```

- One simulation is run for each combination of parameters

SUBMIT CONFIGURATION - OVERVIEW

- **sites** - core set of parameters for remote resources
- **aggregators** - mechanism for grouping multiple sites for the purpose of setting limits on job submission and prioritizing users
- **tools** - specific set of parameters for individual tools
- **managers** - commands to run before and after application execution
- **identities -** configuration parameters for managing shared community credentials
- **monitors** - parameters for configuring job tracking monitors located on remote resources
- **appaccess** - parameters used to manage who can execute which applications on remote resources
- **environmentwhitelist -** permissible user environment variables
- **tunnels** - used to facilitate ssh tunnelling to remote resources

SUBMIT CONFIGURATION - BOINC SITE

- Core set of parameters for remote resources

```
[boinc]
venues = submit.nanohub.org
remotePpn = 1
maximumCores = 1
remoteBatchSystem = BOINC
remoteUser = USER
identityManagers = user
venueMechanism = local
remoteBinDirectory = /var/gridman/submit/bin/Boinc
executableClassificationsAllowed = staged
remoteManager = serial
siteMonitorDesignator = boinc
checkProbeResult = False
```

SUBMIT CONFIGURATION - BOINC TOOL

- **Specific set of parameters for individual tools**

```
[nanoplasticity_r71-slip]
destinations = boinc
executablePath = /apps/nanoplasticity/r71/bin/slip_wrap.py
toolFiles = nanohub_apps_nanoplasticity_r71-slip
environment = PATH=/apps/nanoplasticity/r71/bin:${PATH}
remoteManager = boinc_nanoplasticity
```

SUBMIT/BOINC EXECUTION - Rappture Tool

```
driver=driver_${ToolName}_${Revision}.xml

cat > toolparameters.${ToolName}_${Revision} << EOFPARAMS
file(execute):${driver}
EOFPARAMS

submit --venue boinc \
       --inputfile ${driver} \
       --env TOOL_PARAMETERS=toolparameters.${ToolName}_${Revision} \
       ${ToolName}_${Revision} -w headless
```

- `${ToolName}_${Revision}` **- tag referencing invoke script**
- **TOOL_PARAMETERS** - do not render UI (execute only Rappture)
- **-w headless** - do not use window manager

SUBMIT EXECUTION - Jupyter Notebook

- **Leverage existing Rappture Tools**
  - Load Rappture tool description file (tool.xml)
  - Accept defaults or set input values
  - Save Rappture job description file (driver.xml)
  - Use HUB web api to launch job
  - Ingest results from run description file (run.xml)
  - Standardized analytics provided
- **Create application input file(s) from UI input**
  - Inputs may be single values or parametric sweeps
  - Execute submit command using template files for parametric sweeps
  - Ingest results from application output files
  - Can create custom analytics

UQ - ROSENBROCK FUNCTION

- **Rosenbrock Function**
  - F(X,Y) = 100*(Y - X**2)**2 + (1.-X)**2
- **Standard Rappture UQ widget**
  - Constant
  - Uniform - minimum, maximum
  - Gaussian - mean, standard deviation
  - Available to all Rappture tools
- **Provides Standard Set of Results**
  - PDF for F
  - Sensitivity of F wrt X and Y
  - Response surface - surrogate polynomial model, RMSE
- **BOINC will be available to the 275 supported Rappture tools.**

DEMO

UQ - NANOPLASTICITY/RAPPTURE

The nanoplasticity tool on nanoHUB models crystalline materials with many internal grain boundaries. It calculates the yield stress, a critical parameter that indicates how much stress a material can absorb before permanent (plastic) deformation.

- **Custom UI for variable definition**
- **Standard Outputs**
- **Configured to use BOINC for all simulations**

DEMO

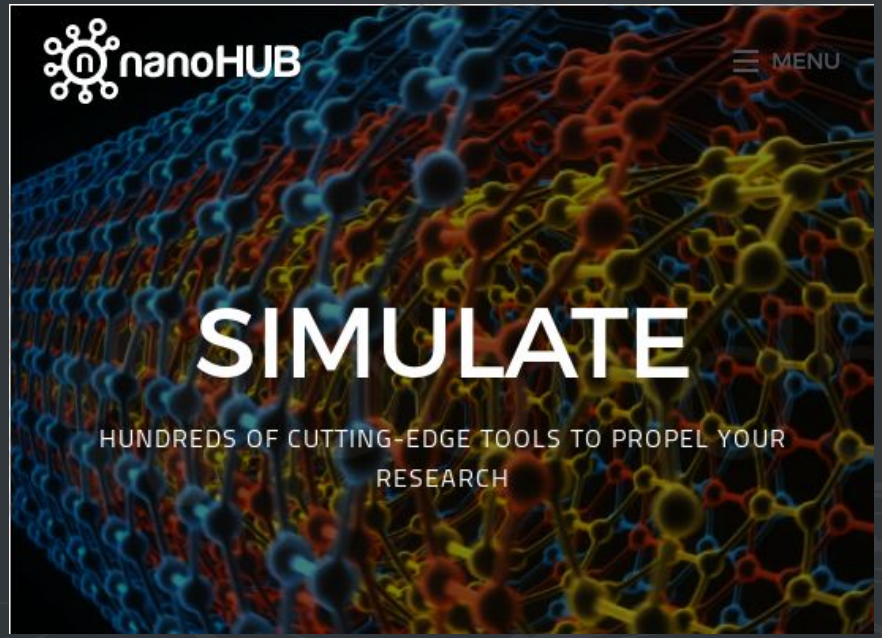
UQ - NANOPLASTICITY/JUPYTER NOTEBOOK

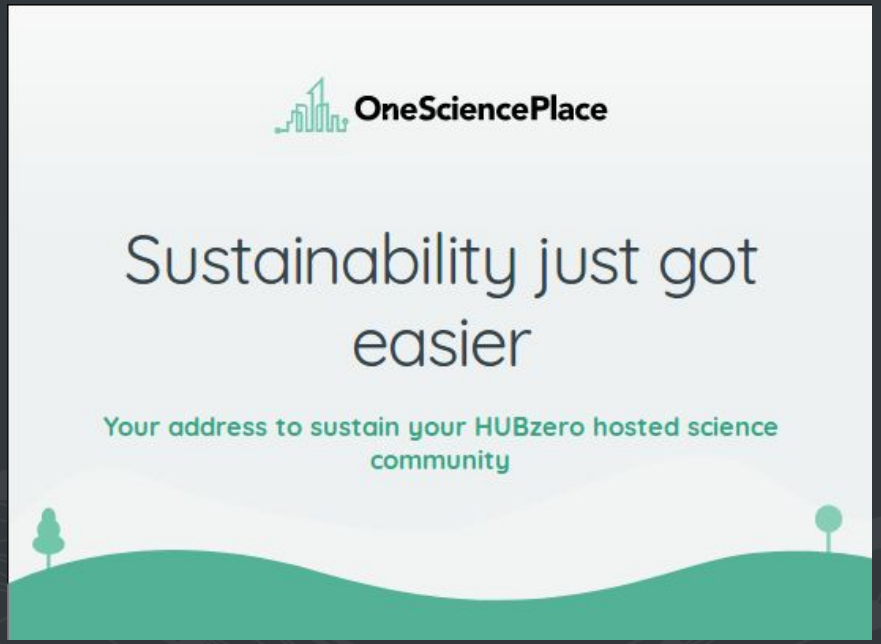
The nanoplasticity tool on nanoHUB models crystalline materials with many internal grain boundaries. It calculates the yield stress, a critical parameter that indicates how much stress a material can absorb before permanent (plastic) deformation.

- **Using Rappture tool as foundation**
  - Input, execution, output characterized by set of XML files
- **Programmable UI**
  - Developers can determine layout and presentation of inputs
- **Standard Outputs**
  - Presentation of results can be customized
  - Results can be presented with interactive displays

DEMO

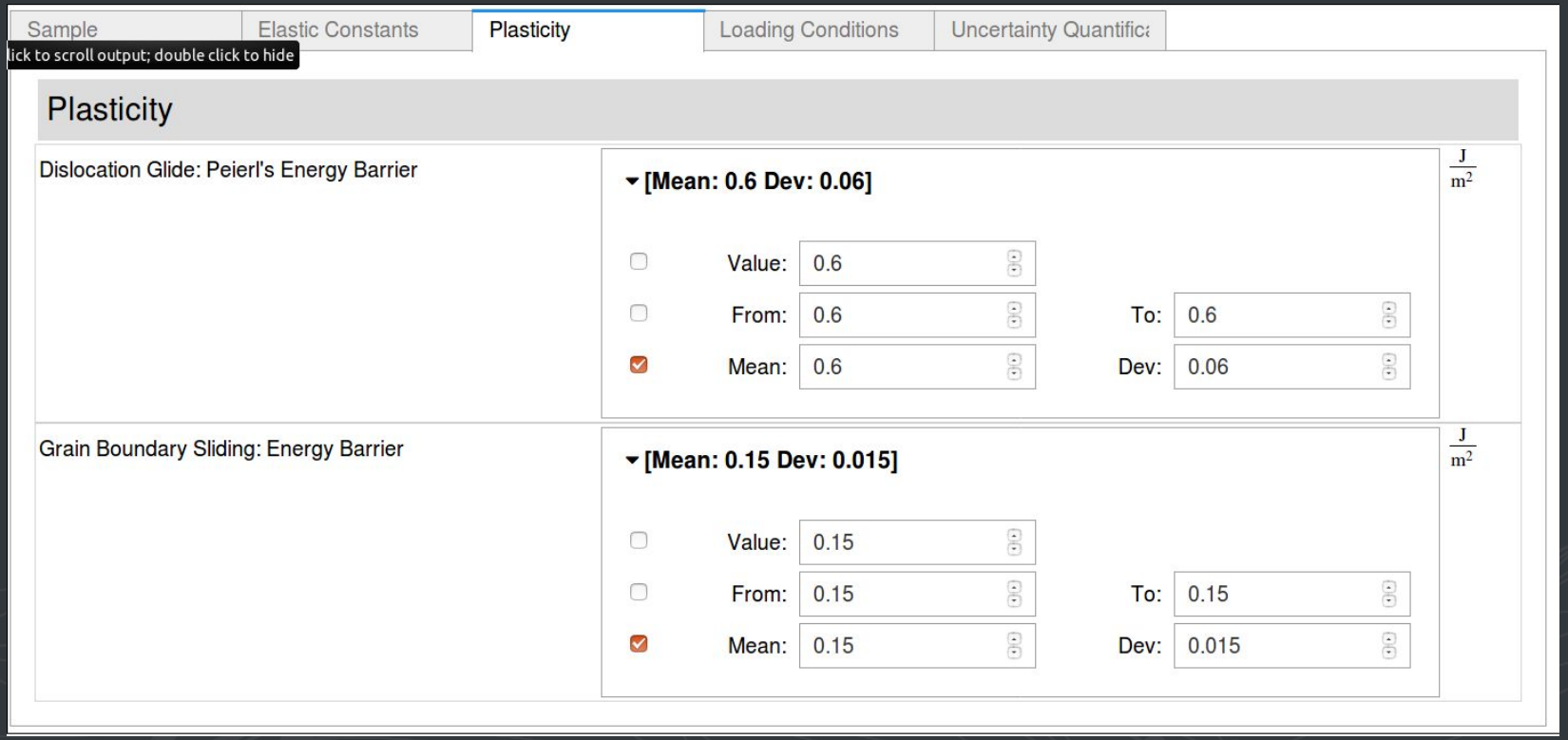
FOR MORE INFORMATION



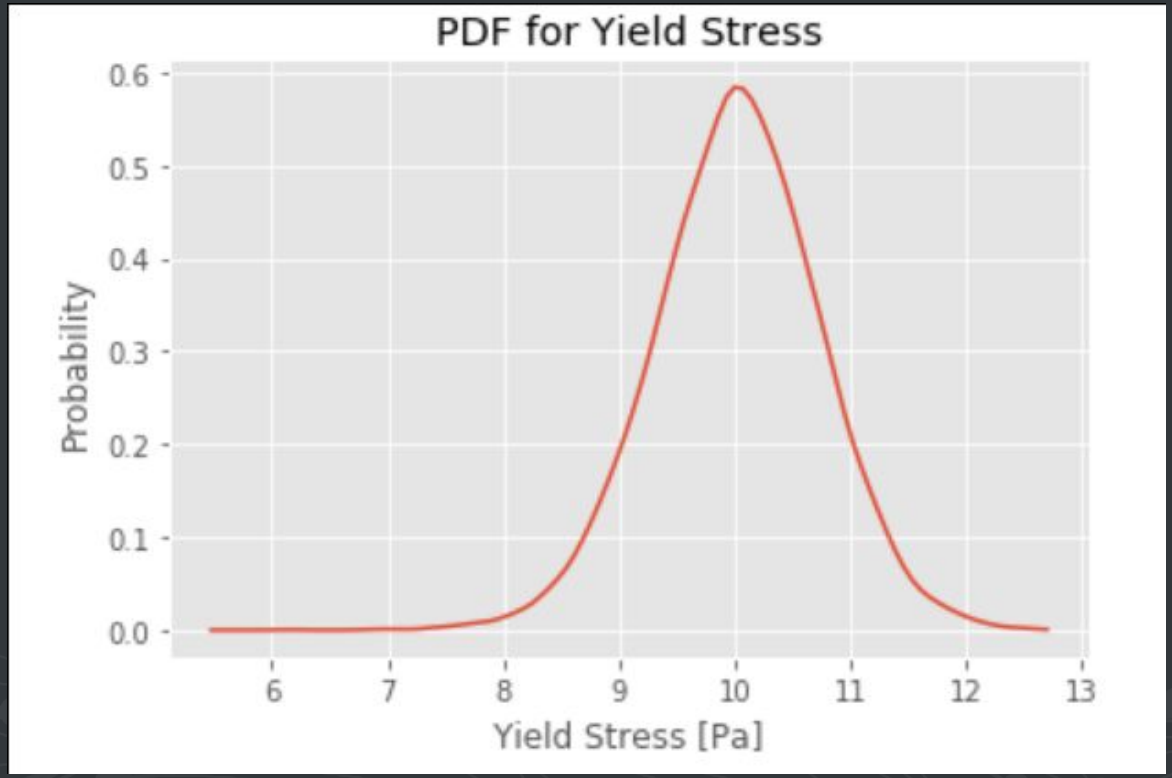


QUESTIONS & ANSWERS

NANOPLASTICITY - INPUT



NANOPLASTICITY - OUTPUT PDF



NANOPLASTICITY - OUTPUT SURROGATE MODEL

```
Response Function RMSE: 2.65%

Response Function = 483.678419180705*glide**3 +
586.628255978445*glide**2*peierls -
839.602512569905*glide**2 -
12.1588916568988*glide*peierls**2 -
7.837850295691 99*glide*peierls + 164.392925675031*glide +
24.0988601649252*peierls**3 - 58.5217324585619*peierls**2 +
38.5932277811335*peierls - 11.1102054574008
```