IRSN
INSTITUT
DE RADIOPROTECTION
ET DE SÛRETÉ NUCLÉAIRE

Enhancing nuclear safety

Stochastic global optimization practice in nuclear criticality safety assessment

Dr. Y. Richet

Criticality Assesment Study and Research Department

17 / 05 / 2011
IHP - Paris

Overview

- Nuclear criticality-safety assessment and related issues
  - Classical approach and basic tools
  - Rethinking through « Computer Experiments » framework:
    « Expert driven » supplemented by « Algorithm assisted» policy

- Components to build an operational workbench
  - Front-end GUI
  - Grid computing engine & algorithm back-end

- Feedback on two years of daily use
  - Adhesion vs. resiliency
  - Enhancing robustness with EGO/kriging improvements

- Focus on stochastic optimization
  - Related [R] packages
  - Integration of stochastic simulator

Nuclear criticality-safety assessment and related issues

Nuclear criticality-safety assessment and related issues

Classical approach and basic tools

Nuclear criticality-safety

- Physics = neutron transport, chemistry, materials science
- System = industrial storage, transport cask, …
- Hypothesis = environment conditions: flooding, earthquake, overloading
- Output of interest = neutrons multiplication factor (aka k-effective), criticality means k-effective > 1.0
- Safety demonstration = k-effective ***not to exceed 0.95, in the hypothesis range***

Assessment: modeling criticality accident risk

Expert knowledge assessment ... supplemented by numerical checking:

- Physics <- criticality simulation code
- System <- input dataset
- Hypothesis <- input dataset variables (dim<10)
- Output of interest <- (scalar) output of the code

Nuclear criticality-safety assessment and related issues

↗ Classical approach and basic tools

Parametric study for maximization of k-effective

- < 10 scalar, independent & compact parameters
- Non linear & cross effects
- (often) Several local max

(Mc) Monte Carlo simulation of k-effective

- One calculation point costs 10 s. to 1 h.
- Endpoint simulation maybe:
  - sd(k-effective) < input setting (say 0.00100)
  - quantile(k-effective,0.999) < input setting (say 0.95000 or current max.)

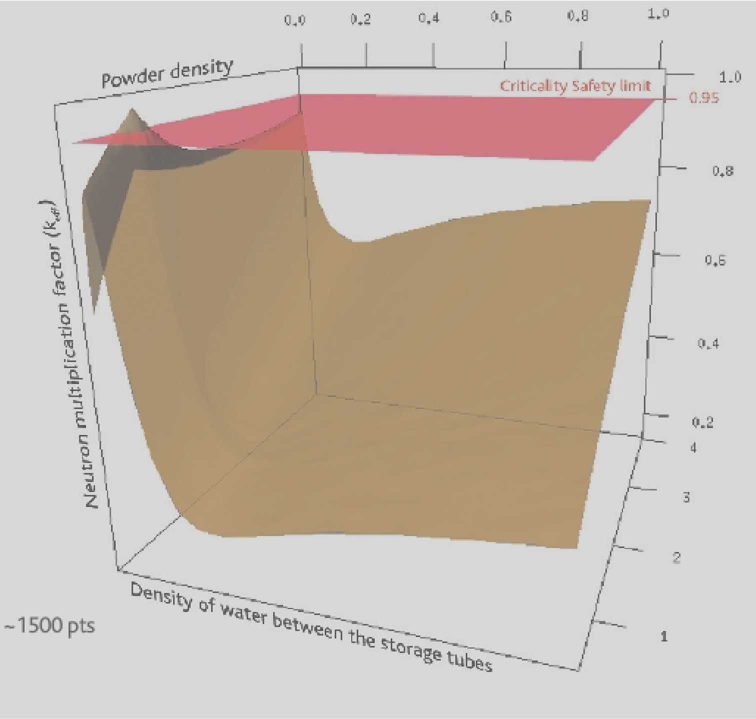
Remote/grid calculation issue at IRSN

- ~ 80 heterogeneous CPU available
- Not (yet) parallelized code (Markov chain)

Nuclear criticality-safety assessment and related issues

Classical approach and basic tools

- Case study = interim storage of dry PuO2 powder, variables being:
  - Powder loading as powder density in [0.5,4.0]
  - Storage flooding as water density between cans in [0.0,1.0]



Nuclear criticality-safety assessment and related issues

➚ Rethinking through « Computer Experiments » framework

Expert-driven approach

- OaT maximization
  - 2 or 3 parameters
  - (supposed) penalizing value for other parameters
  - 1 or 2 cycles by hand ...
- Factorial DoE
  - 5 to 10 points / dimension
  - 2 or 3 parameters
  - automated task with dedicated software (including PROMETHEE)

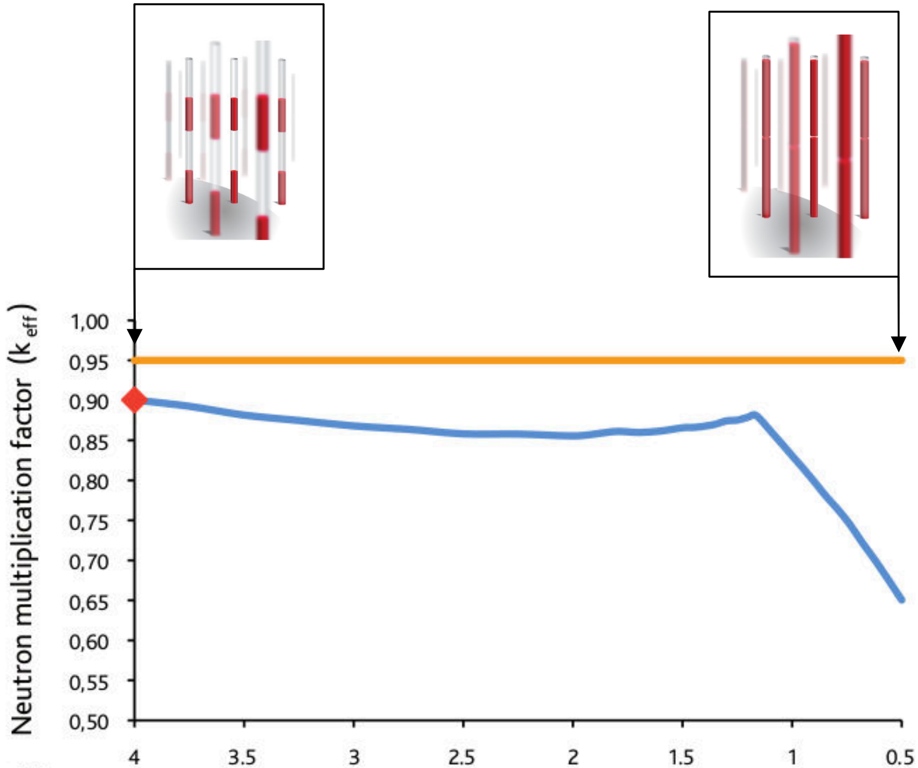
➚ Case study results

Nuclear criticality-safety assessment and related issues

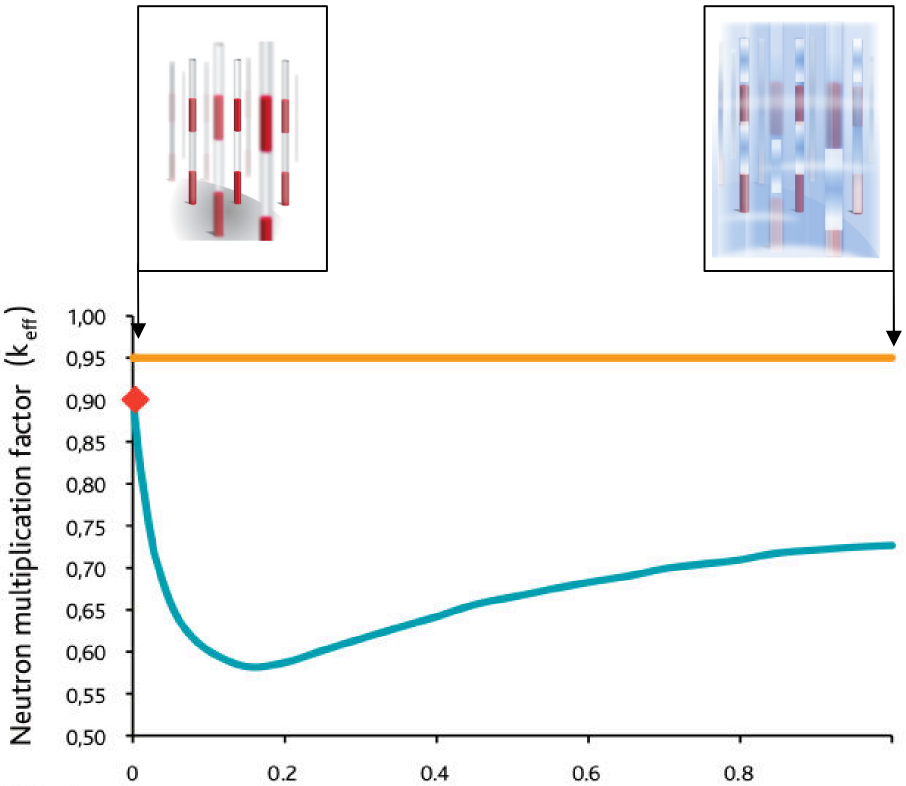
↗ Rethinking through « Computer Experiments » framework

Expert-driven approach – case study results

- OaT maximization



~15 pts



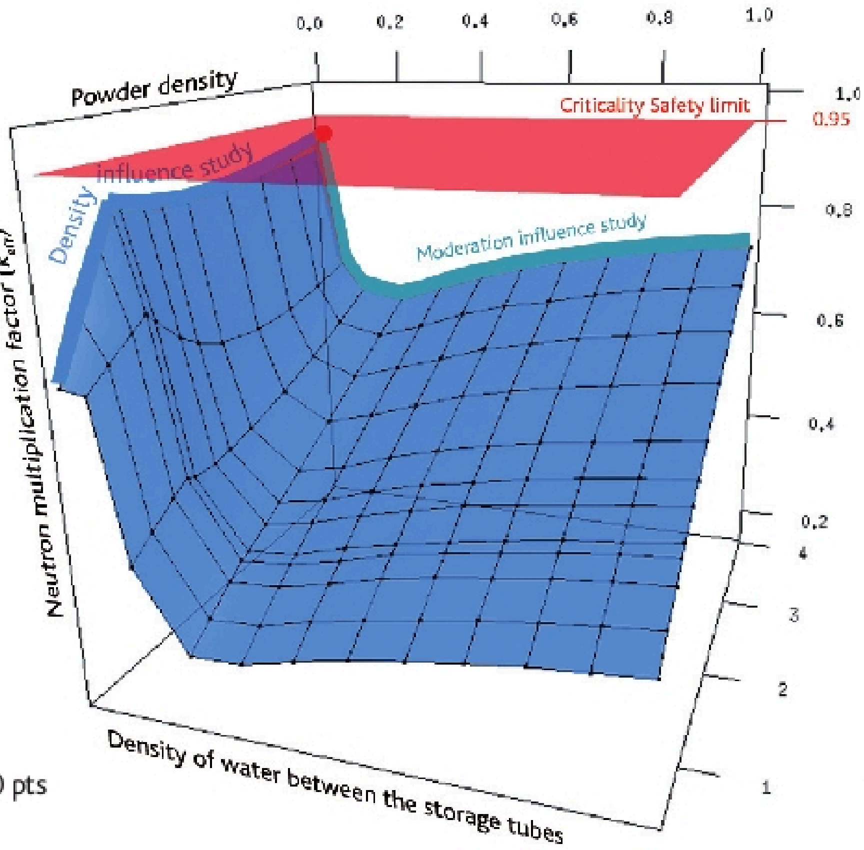
-15 pts

Nuclear criticality-safety assessment and related issues

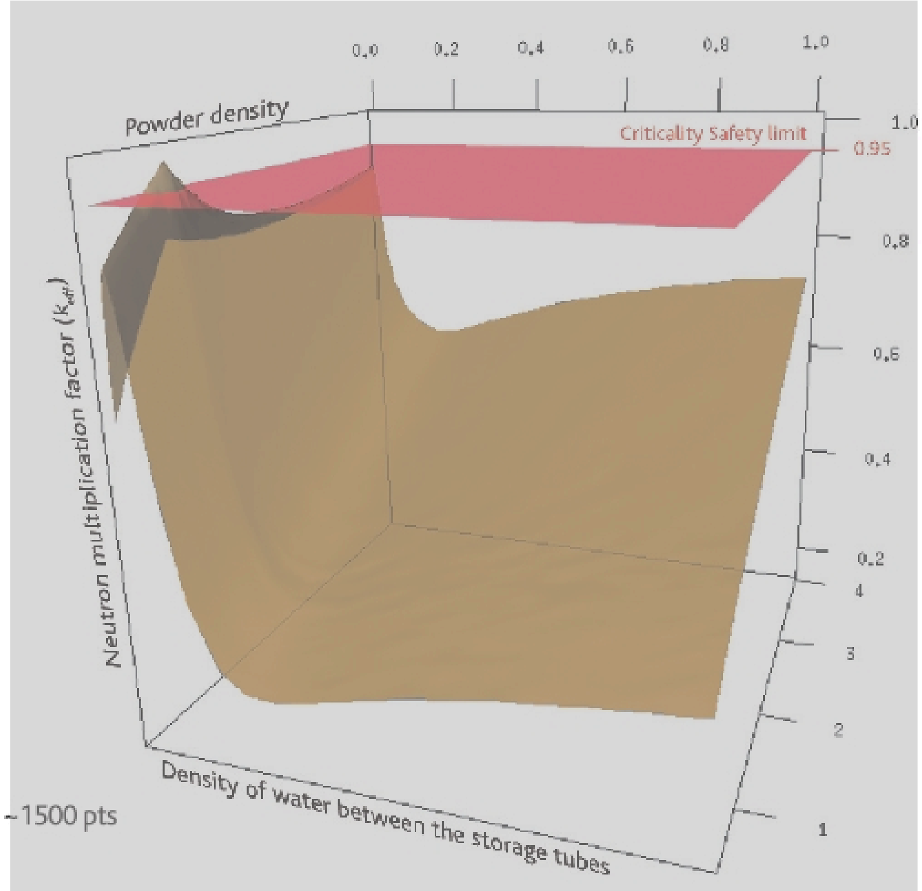
↗ Rethinking through « Computer Experiments » framework

Expert-driven approach – case study results

- Factorial DoE



~150 pts



~1500 pts

Nuclear criticality-safety assessment and related issues

↗ Rethinking through « Computer Experiments » framework

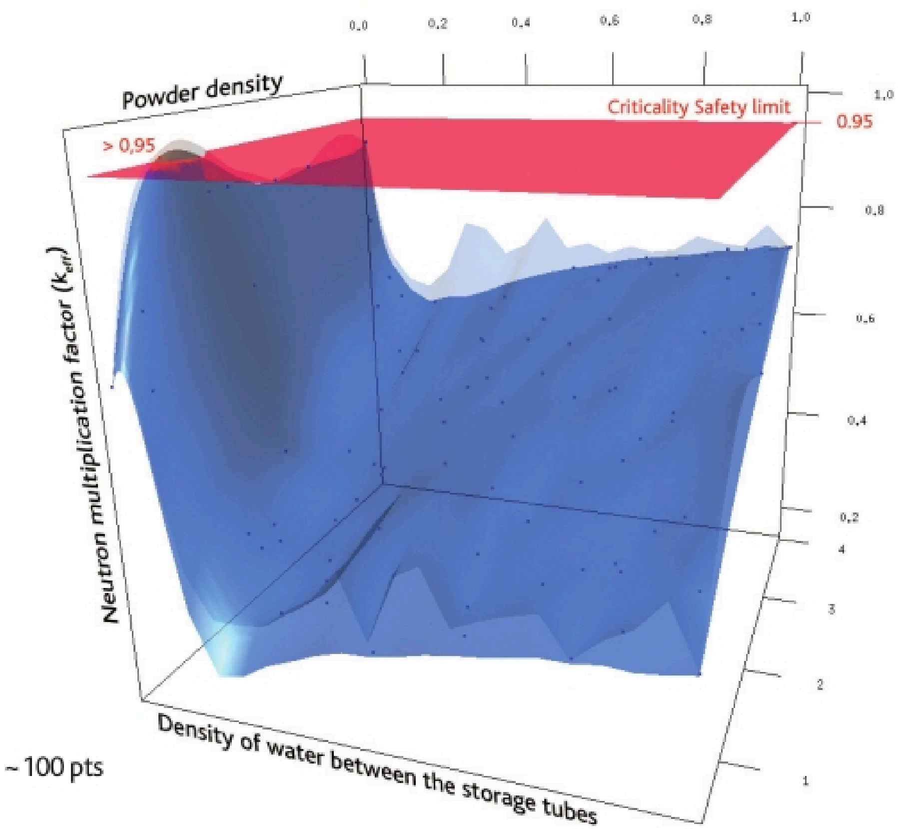
EGO-assisted approach

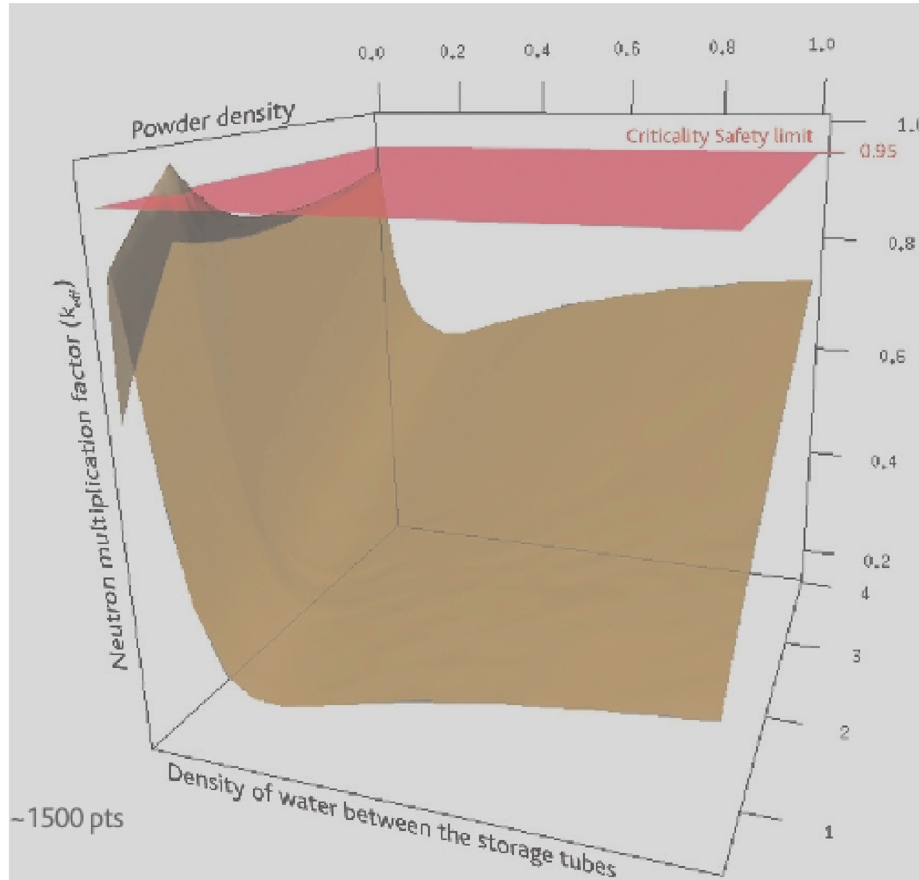
- Initial DoE
  - LHS
  - (cross) Bounds of parameters
- Kriging surrogate of noisy (as MC) experiments
- Maximization of Expected Improvement criterion
- Loop on Kriging/maxEI ...

↗ Case study results

Rethinking through « Computer Experiments » framework

EGO-assisted approach – case study results





Overview

Nuclear criticality-safety assessment and related issues

- Classical approach and basic tools
- Rethinking through « Computer Experiments » framework:
  « Expert driven » supplemented by « Algorithm assisted» policy

Components to build an operational workbench

- Front-end GUI
- Grid computing engine & algorithm back-end

Feedback on two years of daily use

- Adhesion vs. resiliency
- Enhancing robustness with EGO/kriging improvements

Focus on stochastic optimization

- Related [R] packages
- Integration of stochastic simulator

Components to build an operational workbench

Components to build an operational workbench

↗ Front-end GUI

End user input

- Based on ASCII input dataset of your code (application field agnostic)
- User free to insert « $parameter » or formulas anywhere in these file
- Select any DoE policy available for this input parameters / code output
  - Factorial DoE, Efficient Global Optimization
  - … (SA, determ. optimization, inversion, …)

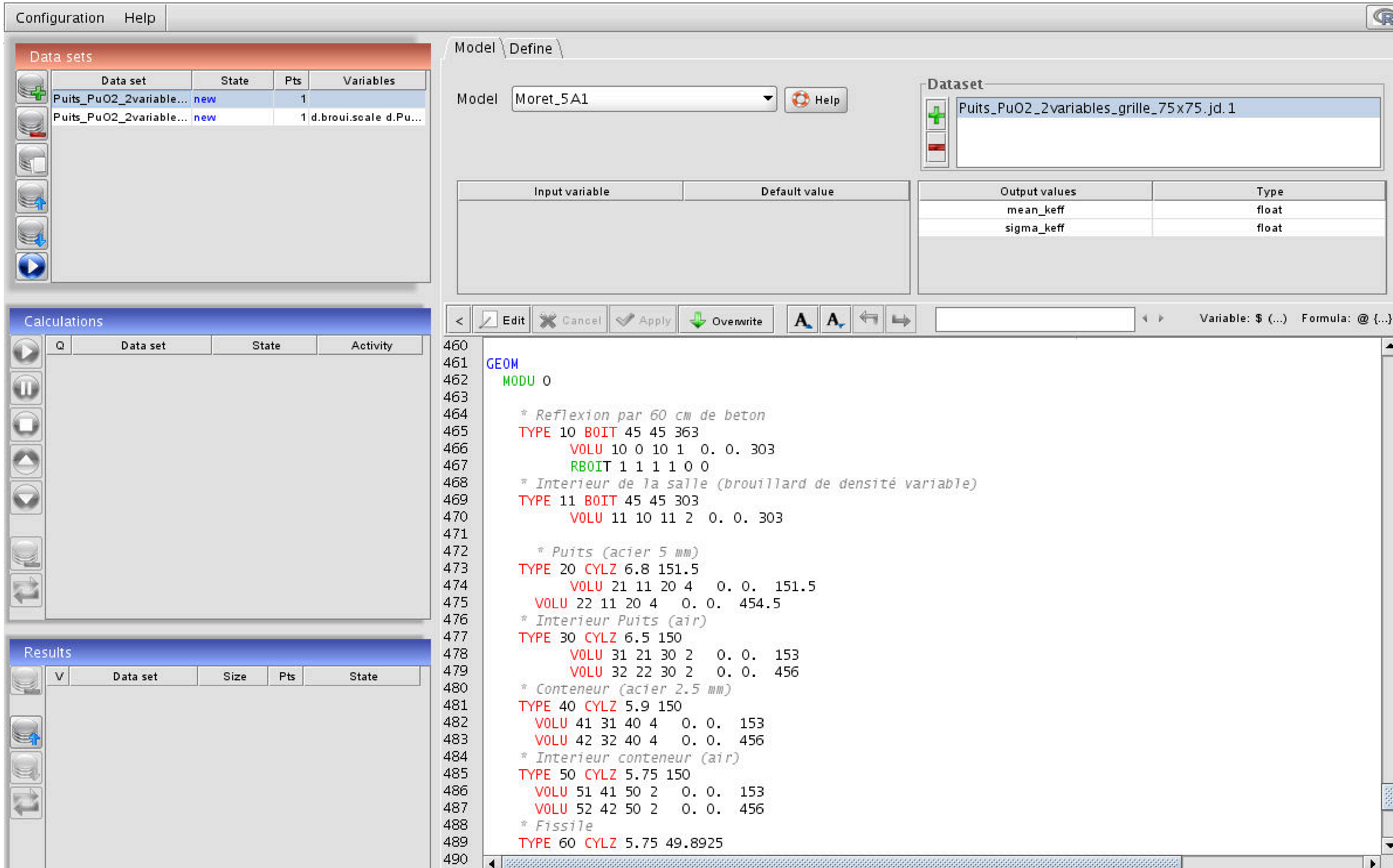
Computing

- (blindly) Launch the grid computing workflow
- (online) Follow intermediate results (if available)

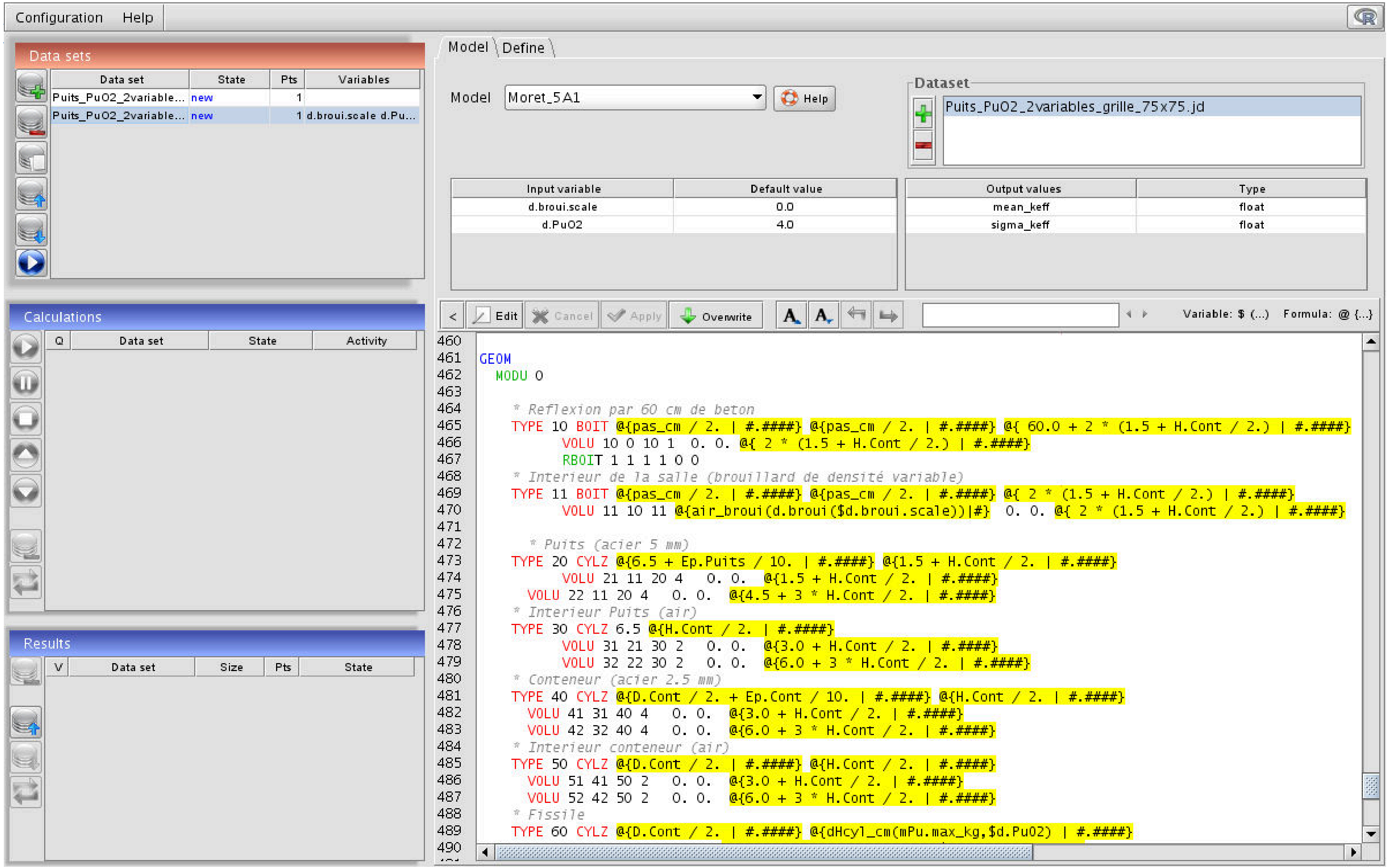
Analysis

- Get results
- … and a true applicable conclusion (or a new question :)

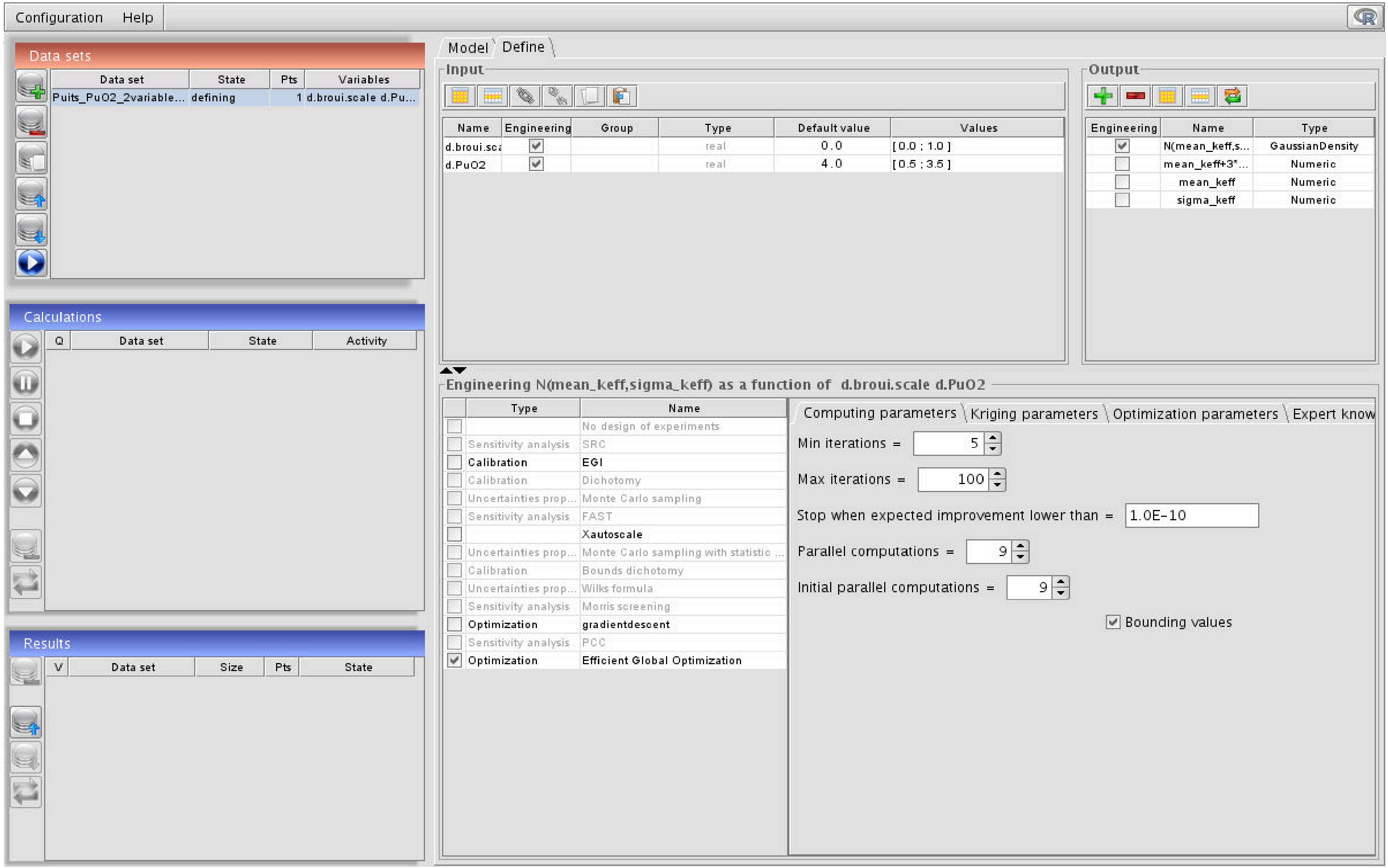
Components to build an operational workbench



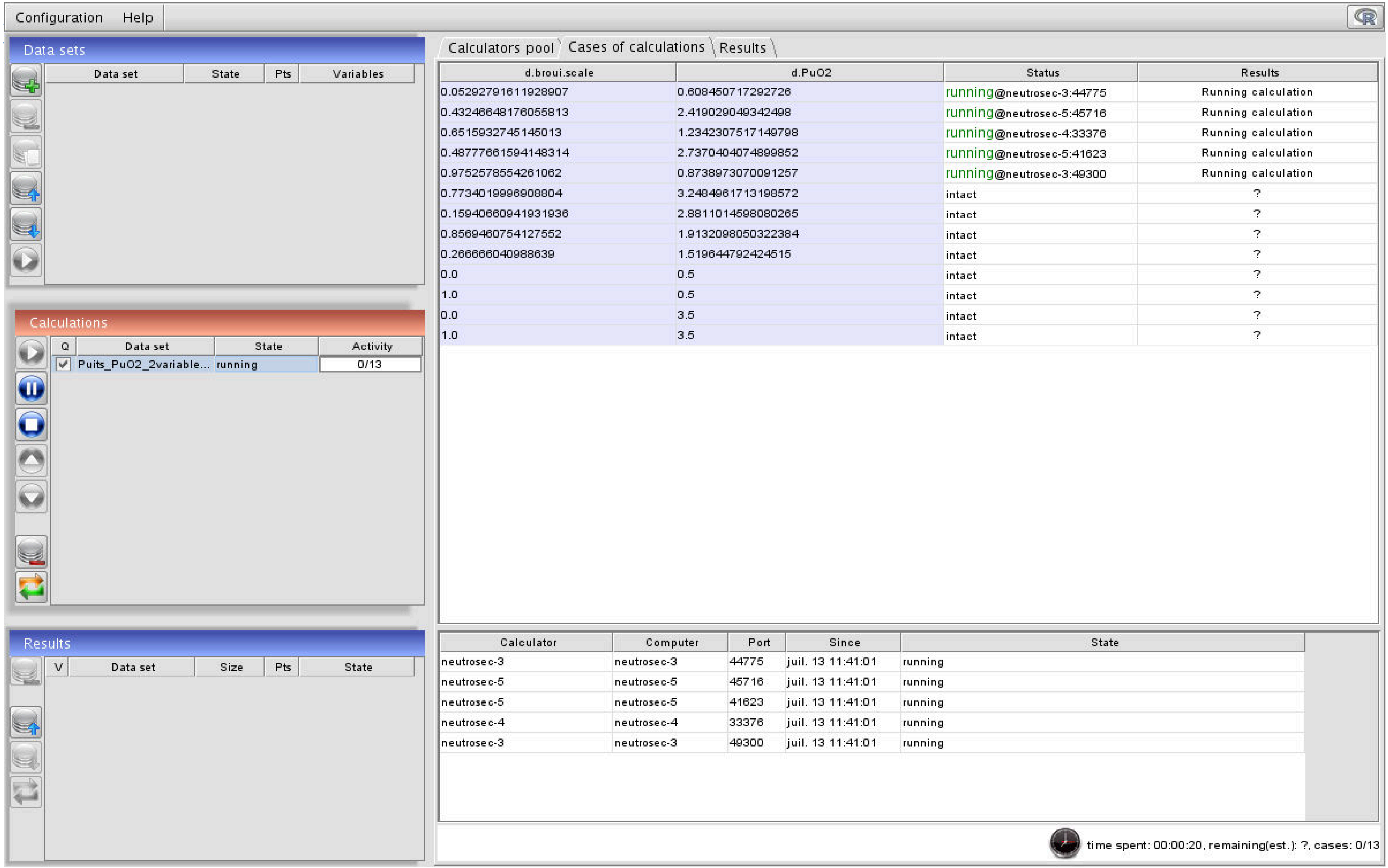
Components to build an operational workbench



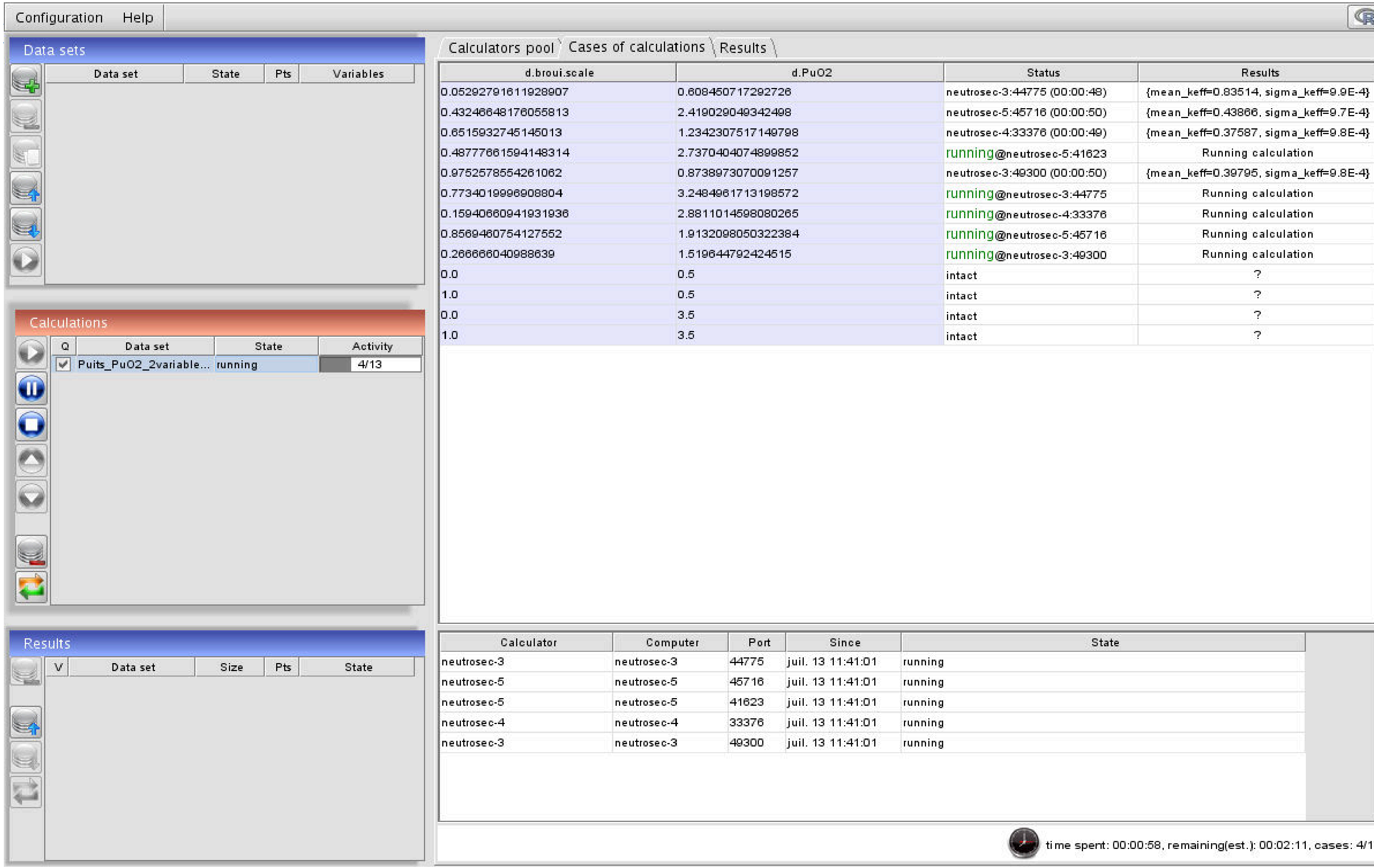
Components to build an operational workbench



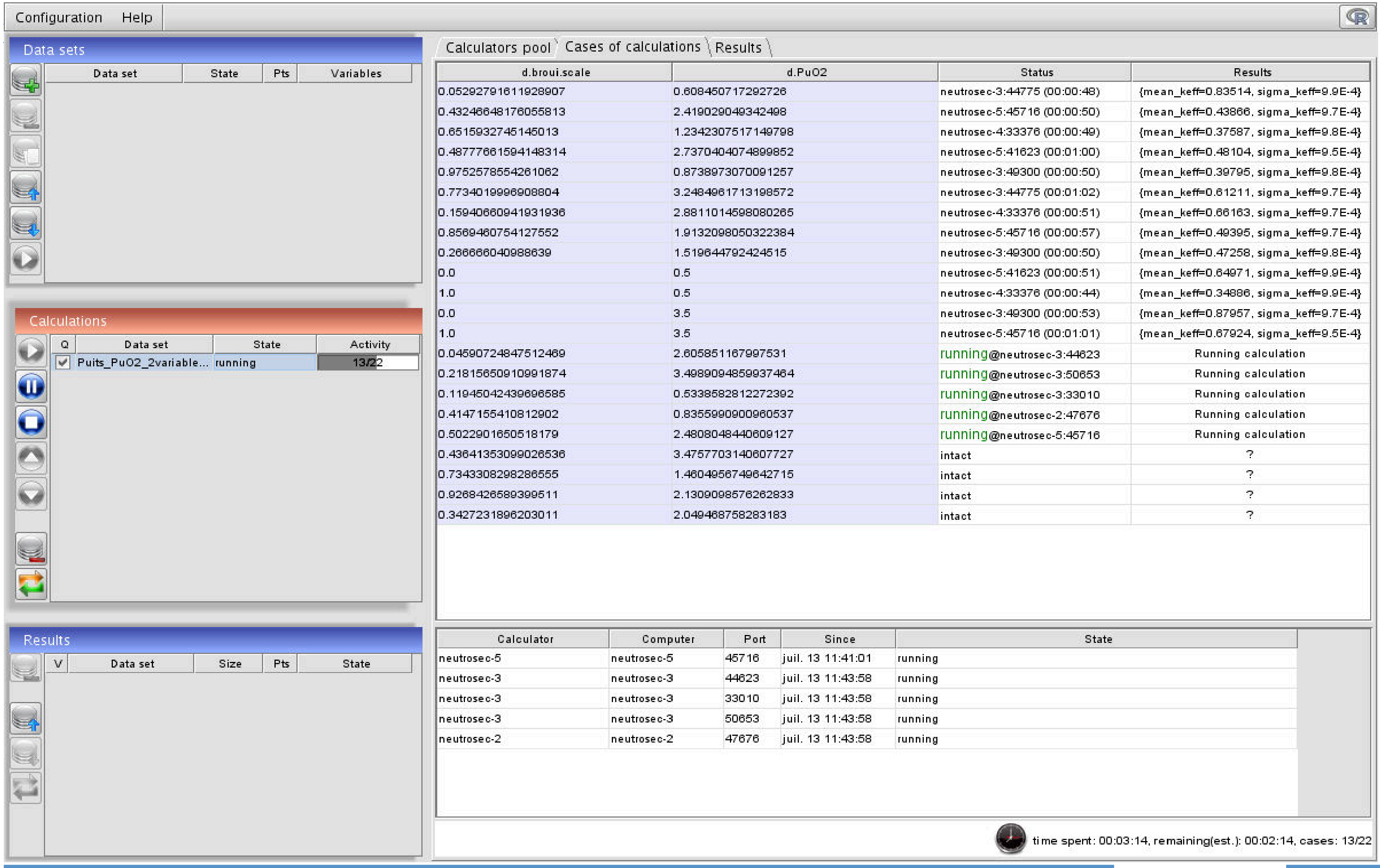
Components to build an operational workbench



Components to build an operational workbench



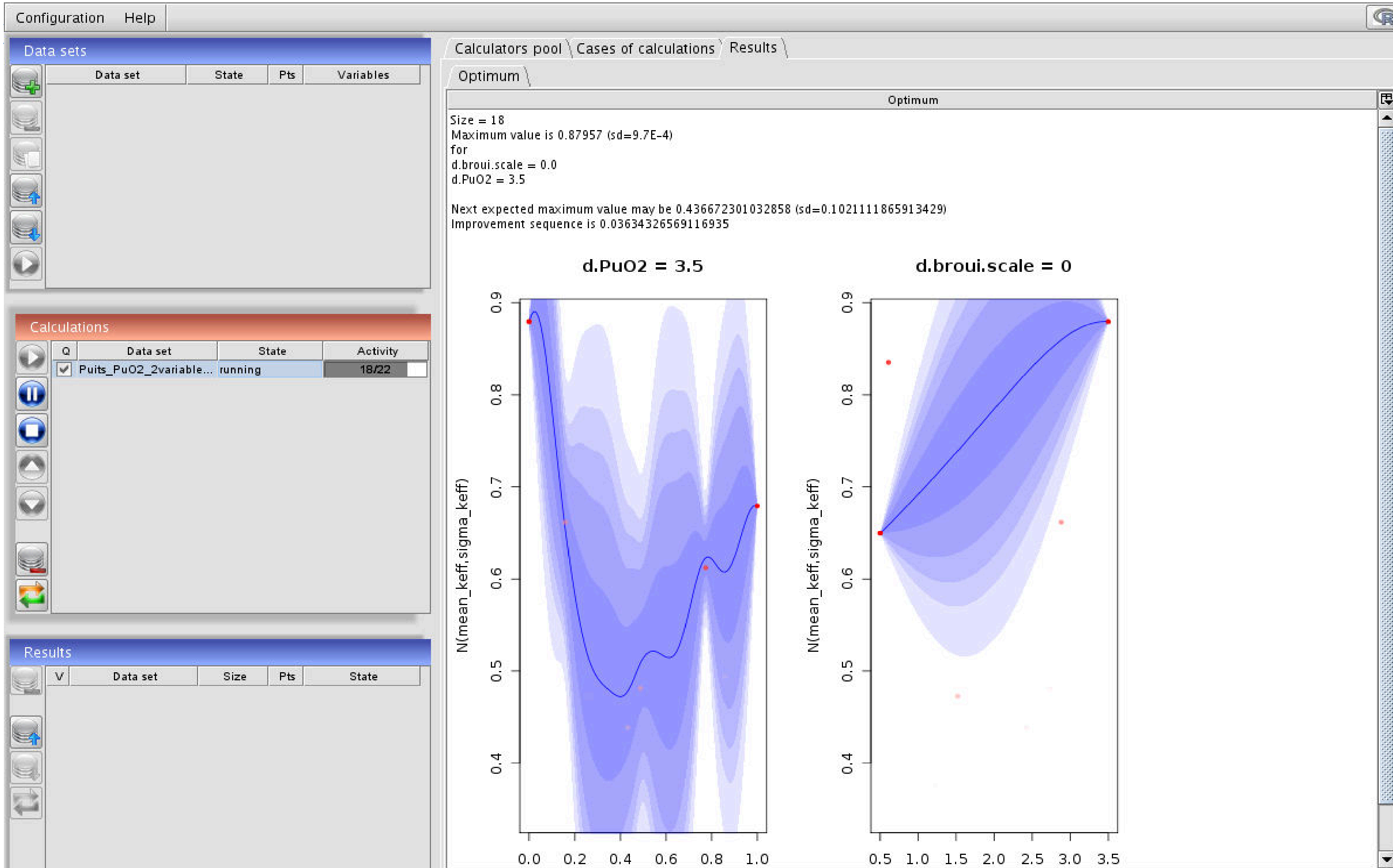
Components to build an operational workbench



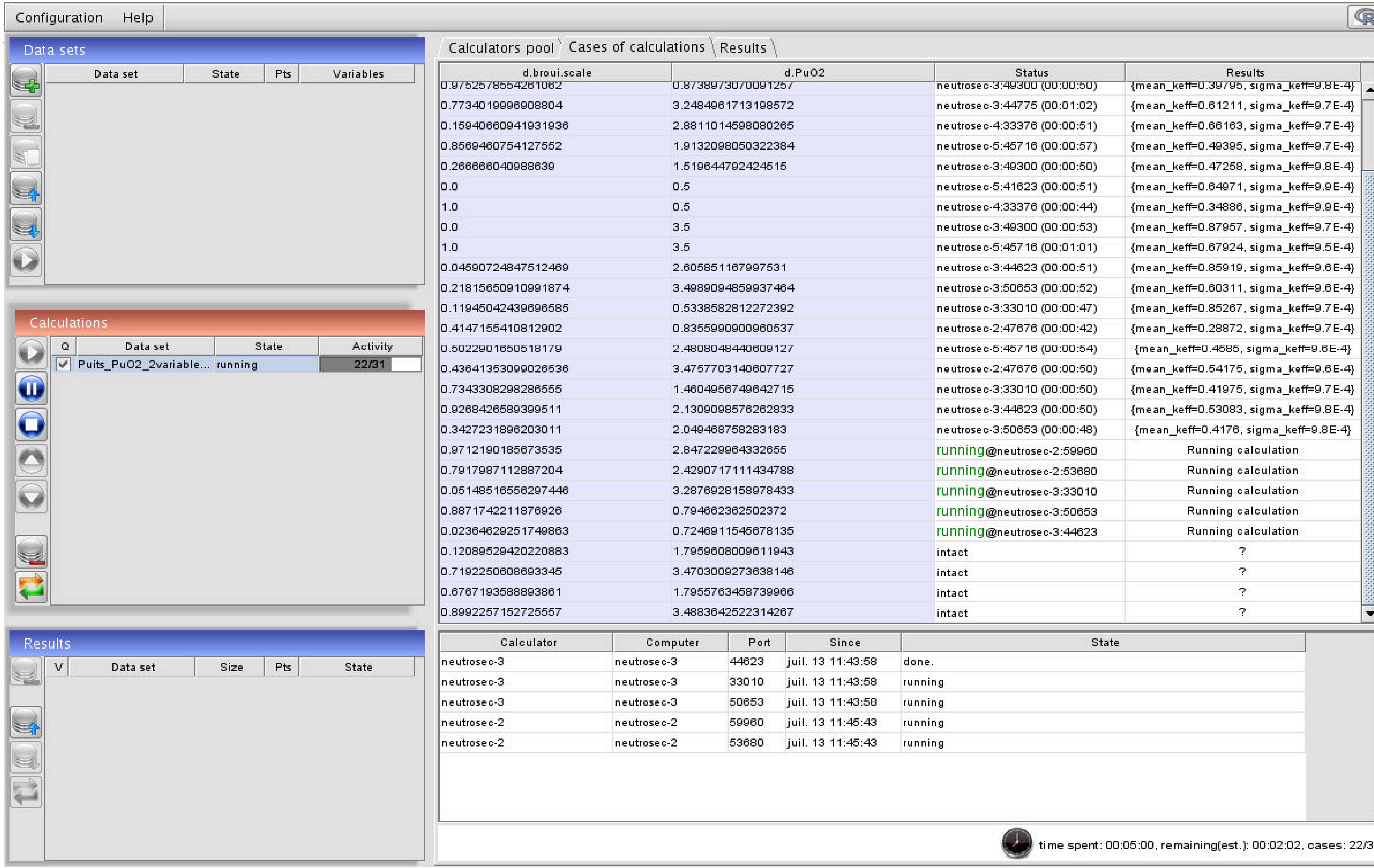
d.broui.scale	d.PuO2	Status	Results
0.05292791611928907	0.608450717292726	neutrosec-3:44775 (00:00:48)	{mean_keff=0.83514, sigma_keff=9.9E-4}
0.43246648176055813	2.419029049342498	neutrosec-5:45716 (00:00:50)	{mean_keff=0.43866, sigma_keff=9.7E-4}
0.6515932745145013	1.2342307517149798	neutrosec-4:33376 (00:00:49)	{mean_keff=0.37587, sigma_keff=9.8E-4}
0.48777661594148314	2.7370404074899852	neutrosec-5:41623 (00:01:00)	{mean_keff=0.48104, sigma_keff=9.5E-4}
0.9752578554261062	0.8738973070091257	neutrosec-3:49300 (00:00:50)	{mean_keff=0.39795, sigma_keff=9.8E-4}
0.7734019996908804	3.2484961713198572	neutrosec-3:44775 (00:01:02)	{mean_keff=0.61211, sigma_keff=9.7E-4}
0.15940660941931936	2.8811014598080265	neutrosec-4:33376 (00:00:51)	{mean_keff=0.66163, sigma_keff=9.7E-4}
0.8569460754127552	1.9132098050322384	neutrosec-5:45716 (00:00:57)	{mean_keff=0.49395, sigma_keff=9.7E-4}
0.266666040988639	1.519644792424515	neutrosec-3:49300 (00:00:50)	{mean_keff=0.47258, sigma_keff=9.8E-4}
0.0	0.5	neutrosec-5:41623 (00:00:51)	{mean_keff=0.64971, sigma_keff=9.9E-4}
1.0	0.5	neutrosec-4:33376 (00:00:44)	{mean_keff=0.34886, sigma_keff=9.9E-4}
0.0	3.5	neutrosec-3:49300 (00:00:53)	{mean_keff=0.87957, sigma_keff=9.7E-4}
1.0	3.5	neutrosec-5:45716 (00:01:01)	{mean_keff=0.67924, sigma_keff=9.5E-4}
0.04590724847512469	2.605851167997531	running@neutrosec-3:44623	Running calculation
0.21815650910991874	3.4989094859937464	running@neutrosec-3:50653	Running calculation
0.11945042439696585	0.5338582812272392	running@neutrosec-3:33010	Running calculation
0.4147155410812902	0.8355990900960537	running@neutrosec-2:47676	Running calculation
0.5022901650518179	2.4808048440609127	running@neutrosec-5:45716	Running calculation
0.43641353099026536	3.4757703140607727	intact	?
0.7343308298286555	1.4604956749642715	intact	?
0.9268426589399511	2.1309098576262833	intact	?
0.3427231896203011	2.049468758283183	intact	?

Calculator	Computer	Port	Since	State
neutrosec-5	neutrosec-5	45716	juil. 13 11:41:01	running
neutrosec-3	neutrosec-3	44623	juil. 13 11:43:58	running
neutrosec-3	neutrosec-3	33010	juil. 13 11:43:58	running
neutrosec-3	neutrosec-3	50653	juil. 13 11:43:58	running
neutrosec-2	neutrosec-2	47676	juil. 13 11:43:58	running

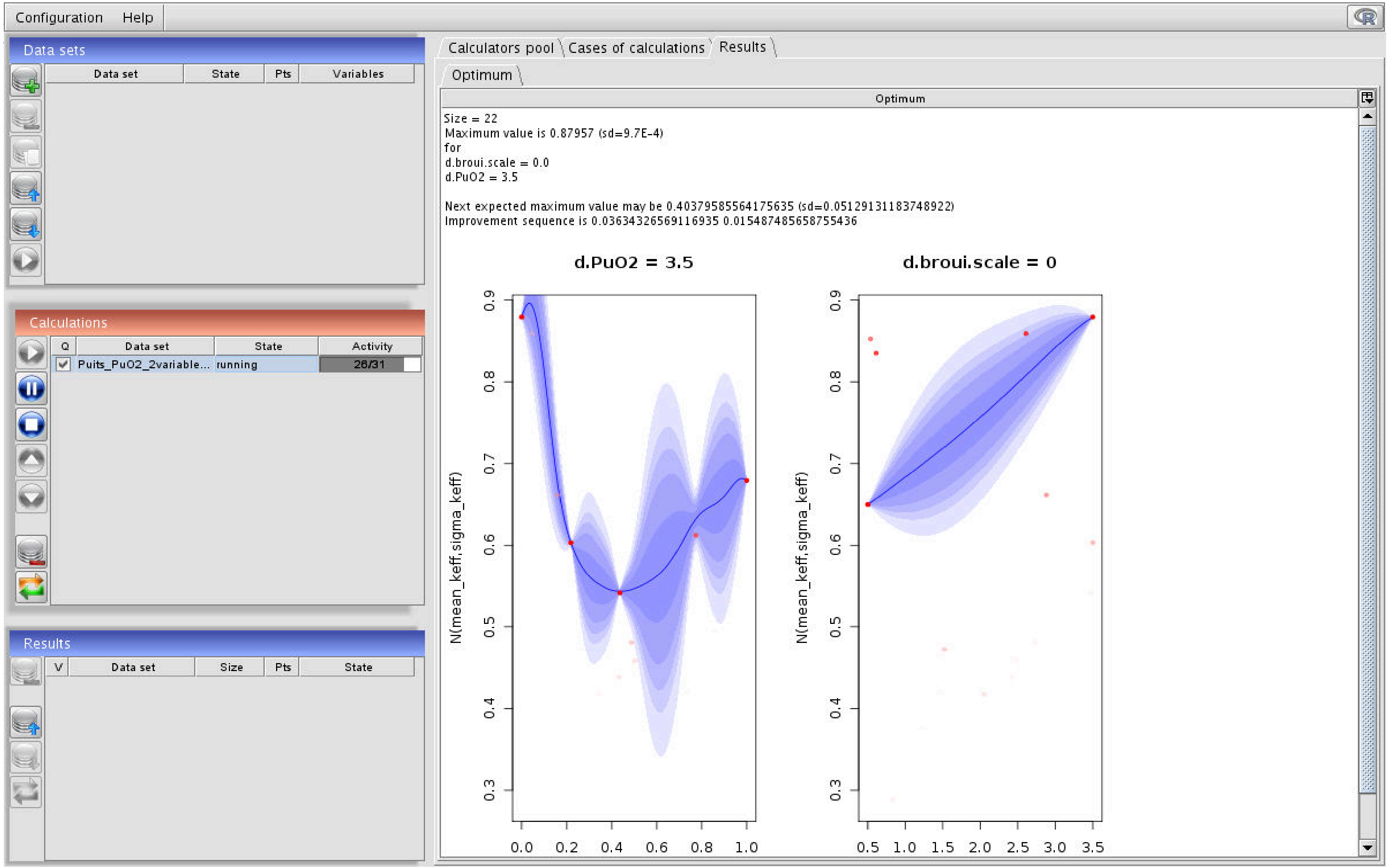
Components to build an operational workbench



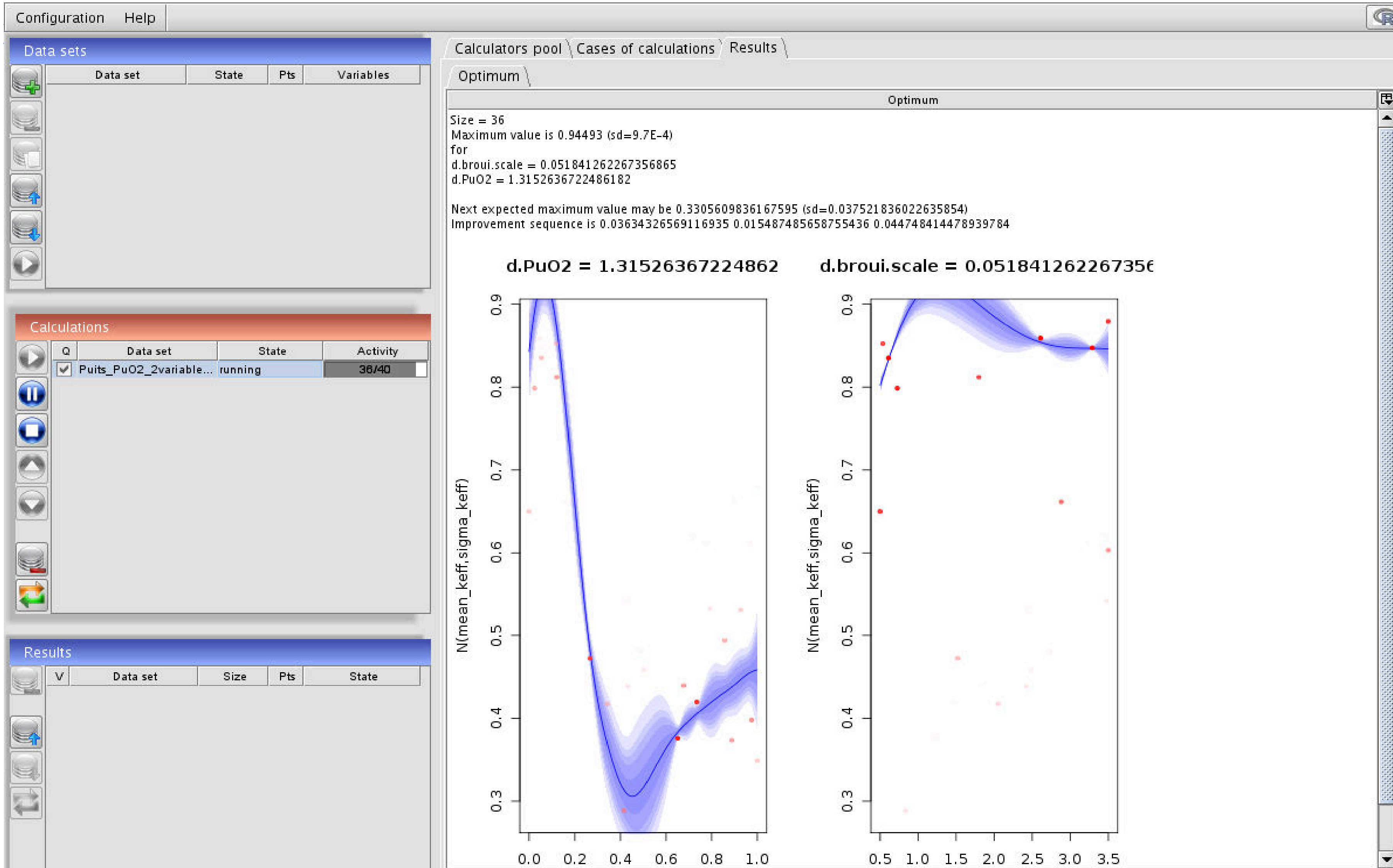
Components to build an operational workbench



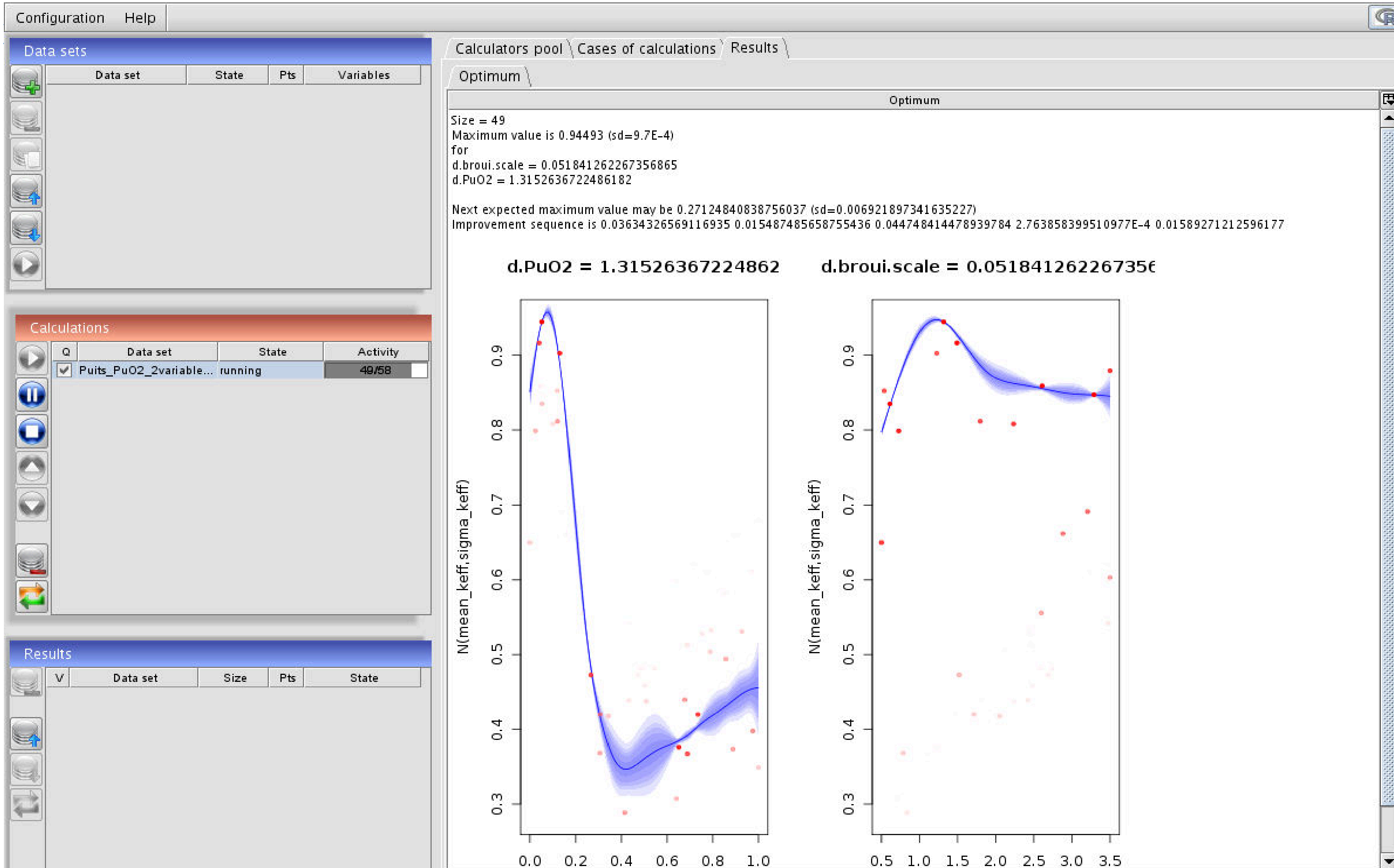
Components to build an operational workbench



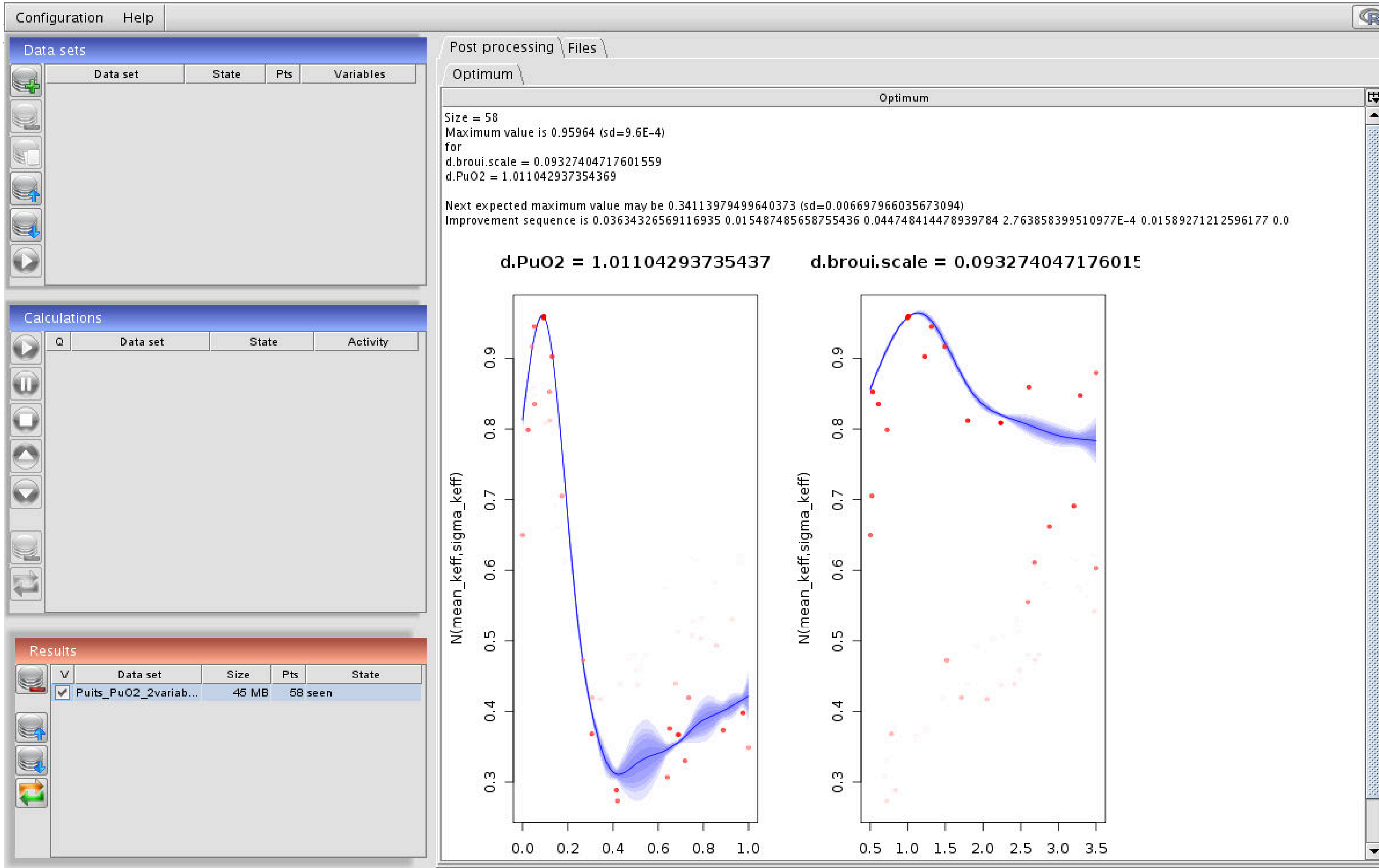
Components to build an operational workbench



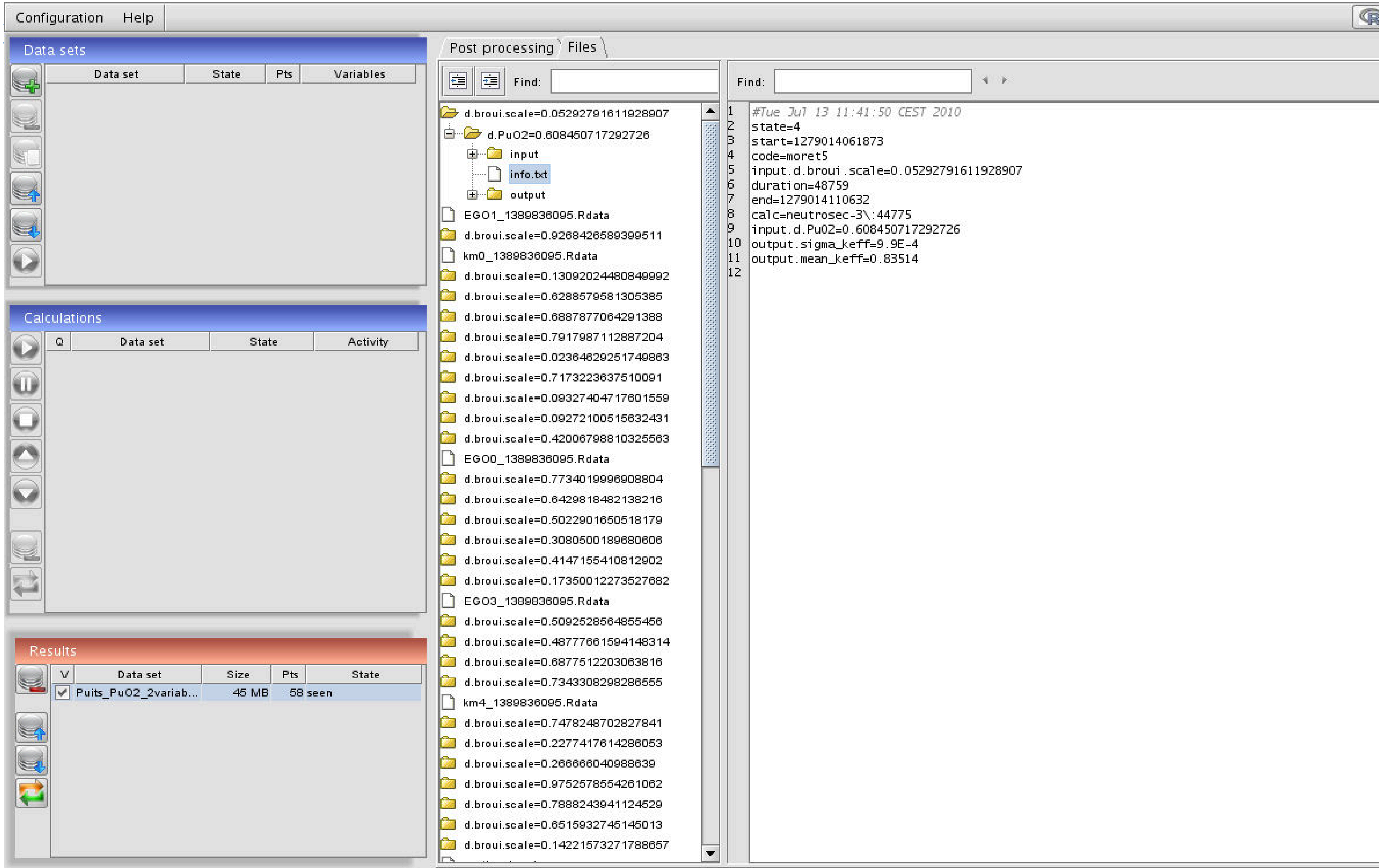
Components to build an operational workbench



Components to build an operational workbench



Components to build an operational workbench



Components to build an operational workbench

Grid computing engine & algorithm back-end

Distributed computing engine

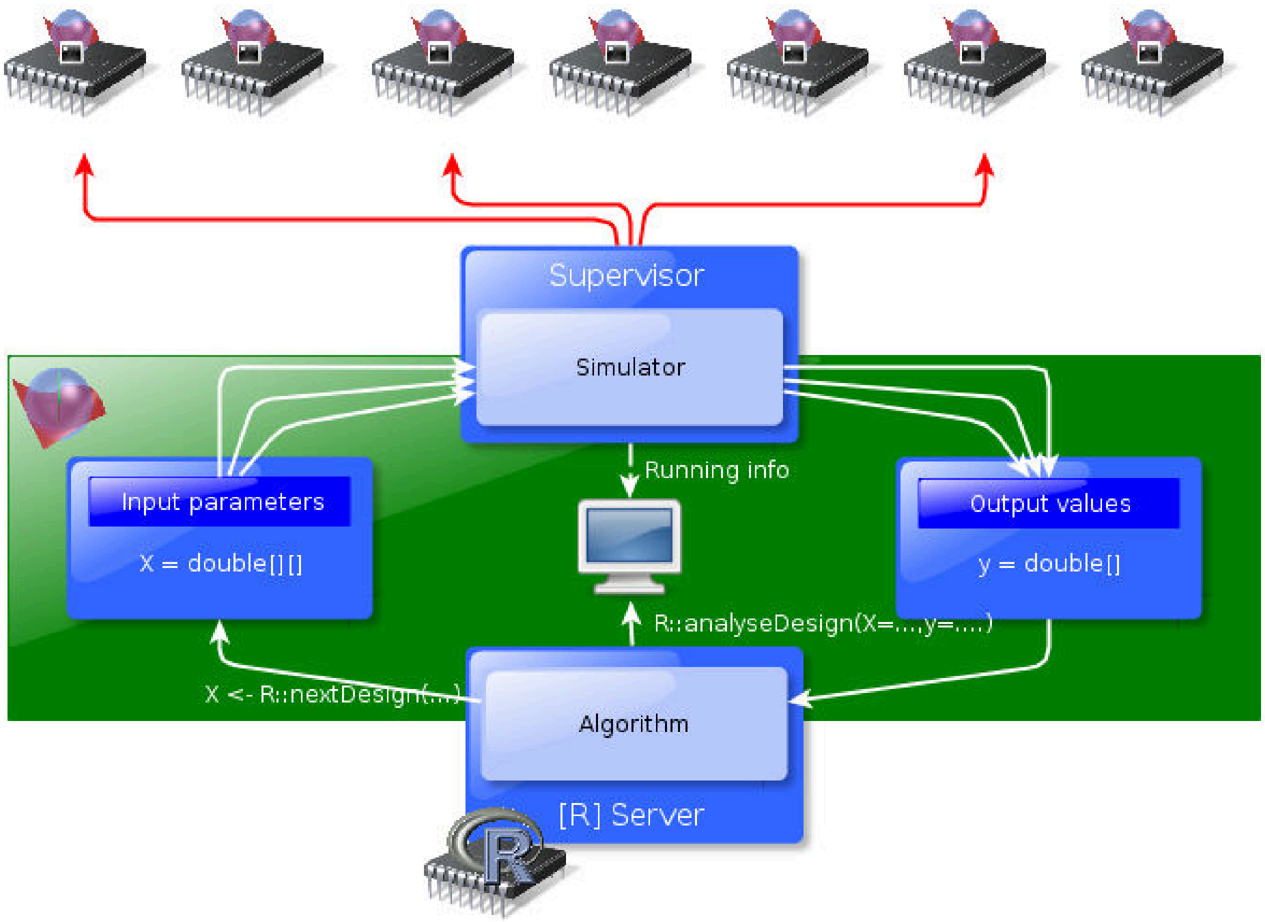
- Asynchronous, remote & parallel distribution
- Back-end daemon compatible with larger set of CPU boxes :
  server, workstation, grid, cluster, ... and even Windows office desktop
- Dynamic merge of heterogeneous computing power
- Failover

Algorithm back-end

- [R] (remote or local)
- Light wrapping script

Components to build an operational workbench

↗ Grid computing engine & algorithm back-end



Components to build an operational workbench

↗ Grid computing engine & algorithm back-end

```
#' constructor and initializer of R session
init <- function() {
    library(lhs)
    ...
}

#' first design building. All variables are set in [0,1]. d is the dimension, or number of variables
#' @param d number of variables
#' @return next design of experiments
buildInitialDesign <- function(d) {
    set.seed(1)
    lhs <- maximinLHS(n=initBatchSize,k=d)
    ...
}

#' iterated design building.
#' @param X data frame of current doe variables (in [0,1])
#' @param Y data frame of current results
#' @return  next doe step
prepareNextDesign <- function(X,Y) {
    ...
    return(as.matrix(Xnext))
}

#' final analysis. All variables are set in [0,1]. Return HTML string
#' @param X data frame of doe variables (in [0,1])
#' @param Y data frame of results
#' @return HTML string of analysis
analyseDesign <- function(X,Y) {
    ...
    html=paste(sep="<br/>",paste("<HTML>... ",m),...,"</HTML>")
    ...
    return(paste(html,plot))
}
```

Components to build an operational workbench

↗ Grid computing engine & algorithm back-end

```r
buildInitialDesign <- function(d) {...}

#' iterated design building.
#' @param X data frame of current doe variables (in [0,1])
#' @param Y data frame of current results
#' @return next DoE step
prepareNextDesign <- function(X,Y) {
    if (iEGO > iterations) return();
    iEGO <<- iEGO + 1
    ...
    noise.var <<- as.array(Y[,2])
    if (search_min) {y=Y[,1]} else {y=-Y[,1]}
    ...
    kmi <- km(design=X, response=y, trend, optim.method='gen',... )
    EGOi <- max_qEI.CL(model=kmi, npoints=batchSize, L=liar(as.array(Y[,1])), ...)
    return(as.matrix(EGOi$par))
}

#' final analysis. All variables are set in [0,1]. Return HTML string
#' @param X data frame of doe variables (in [0,1])
#' @param Y data frame of results
#' @return HTML string of analysis
analyseDesign <- function(X,Y) {
    analyse.files <<- paste("sectionview_",iEGO-1,".png",sep="")
    png(file=analyse.files,bg="transparent",height=resolution,width = resolution)
    ...
    html <- paste(sep="<br/>",paste("<HTML>minimum is ",m),...
```

Overview

Nuclear criticality-safety assessment and related issues

- Classical approach and basic tools
- Rethinking through « Computer Experiments » framework: « Expert driven » supplemented by « Algorithm assisted» policy

Components to build an operational workbench

- Front-end GUI
- Grid computing engine & algorithm back-end

Feedback on two years of daily use

- Adhesion vs. resiliency
- Enhancing robustness with EGO/kriging improvements

Focus on stochastic optimization

- Related [R] packages
- Integration of stochastic simulator

Feedback on two years of daily use

Feedback on two years of daily use

Adhesion vs. Resiliency

Resiliency of « old school » practices

- Coverage: Only 50-80% of assessment practice is covered by the « computer experiments » framework...
  - => **Let users do whatever they want (including previous old practice)**
- Quality Insurance: workflow is already mapped on existing tools...
  - => The workbench have to be **flexible enough** to permit same level of QI
- Users: to master new concepts, the learning curve is sometimes too steep
  - => **Take time to explain...**

Adhesion to this "new" assessment framework

- <at first> focus on early adopters (easier 20% of target)
- <then> capitalize on « real world » feedback, to adapt the solution
- <try to> convince wider and wider... adapt again and again...
- <finally> involve people in R&D policy

- Efficiency measure based on « real world » test cases
- Smooth (software) transition between *old* and *new* practices

Feedback on two years of daily use

↗ Enhancing robustness with EGO/kriging improvements

Noisy kriging

- Noise to model random Gaussian output of MC code
- Heteroscedasticity to support arbitrary « fidelity » of experiments

Automatic input « scaling »

- Support for input parameters transformations
  log/exp/logistic emulated as local 2^{nd} degree splines
- To be published in DiceKriging 1.2 (soon)

Parallel EGO

- Constant Liar heuristic
- Tunnable deepening of EGO

Overview

Nuclear criticality-safety assessment and related issues

- Classical approach and basic tools
- Rethinking through « Computer Experiments » framework: « Expert driven » supplemented by « Algorithm assisted» policy

Components to build an operational workbench

- Front-end GUI
- Grid computing engine & algorithm back-end

Feedback on two years of daily use

- Adhesion vs. resiliency
- Enhancing robustness with EGO/kriging improvements

Focus on stochastic optimization

- Related [R] packages
- Integration of stochastic simulator

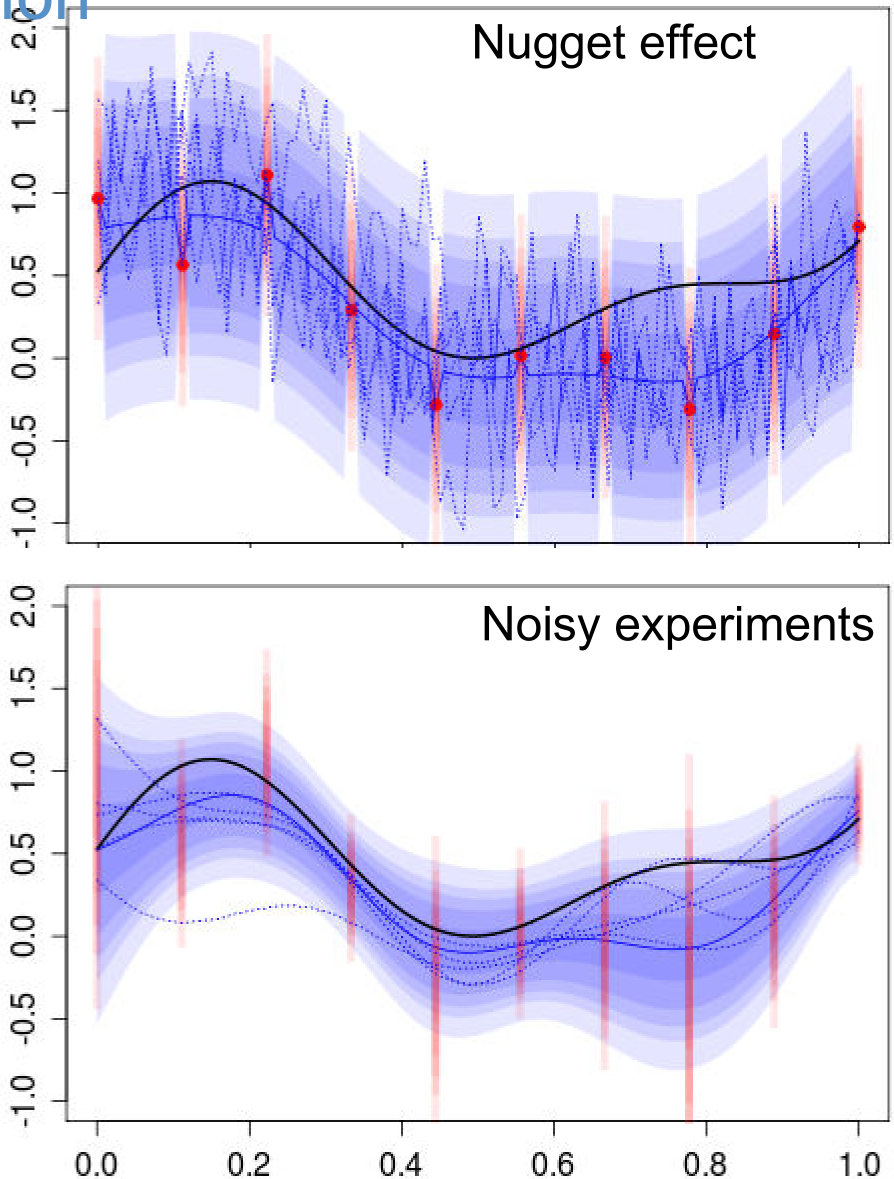
Focus on stochastic optimization

Focus on stochastic optimization

↗ Related [R] packages

DiceKriging

- Ordinary / Universal kriging
- Nugget / Noise



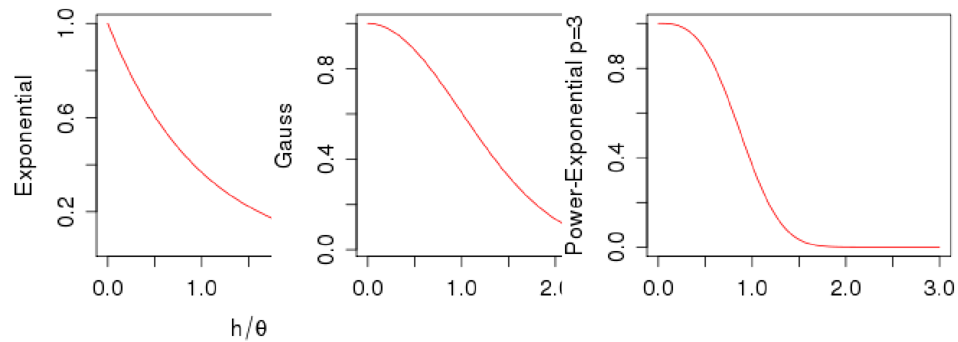
- Kernel:
  - Exponential
  - Gauss
  - Power-exponential
  - Matern 3/2
  - Matern 5/2

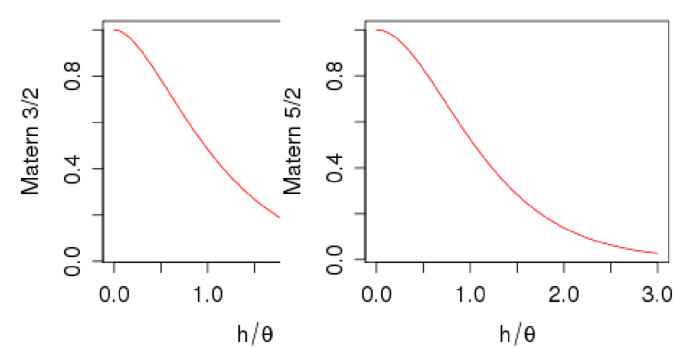
Focus on stochastic optimization

↗ Related [R] packages

DiceKriging

- Ordinary / Universal kriging
- Nugget / Noise
- Kernel: $C(x,y)=s^2 k(x,y)$
  - Exponential
    $$k(h=|x-y|)=e^{-\frac{h}{\theta}}$$
  - Gauss
    $$k(h=|x-y|)=e^{-\left(\frac{h}{\theta}\right)^2}$$
  - Power-exponential
    $$k(h=|x-y|)=e^{-\left(\frac{h}{\theta}\right)^p}$$
  - Matern 3/2
    $$k(h=|x-y|)=\left(1+\sqrt{3}\frac{h}{\theta}\right)e^{-\sqrt{3}\frac{h}{\theta}}$$
  - Matern 5/2
    $$k(h=|x-y|)=\left(1+\sqrt{5}\frac{h}{\theta}+\frac{5}{3}\left(\frac{h}{\theta}\right)^2\right)e^{-\sqrt{5}\frac{h}{\theta}}$$



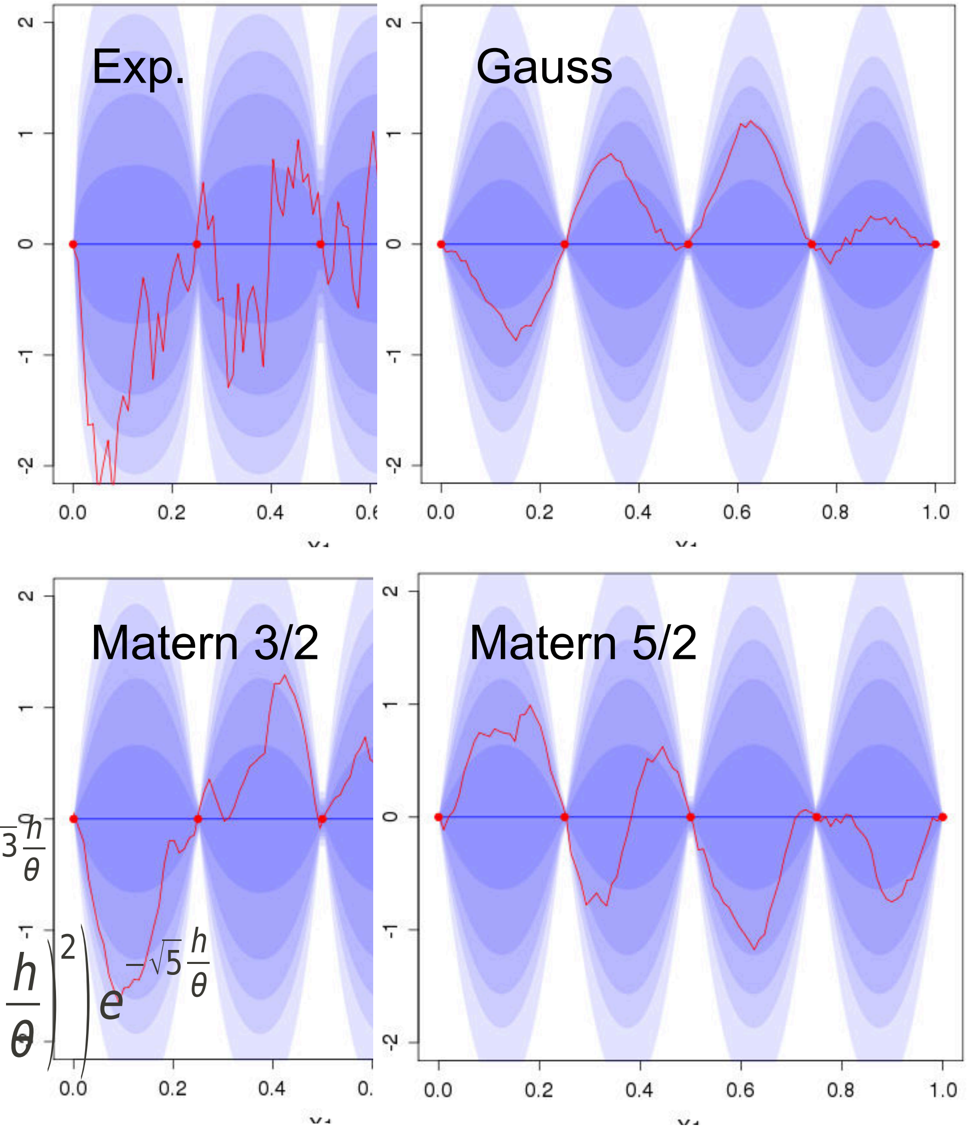


Focus on stochastic optimization

➚ Related [R] packages

DiceKriging

- Ordinary / Universal kriging
- Nugget / Noise
- Kernel: $C(x,y)=s^2 k(x,y)$
  - Exponential
    $$k(h=|x-y|)=e^{-\frac{h}{\theta}}$$
  - Gauss
    $$k(h=|x-y|)=e^{-\left(\frac{h}{\theta}\right)^2}$$
  - Power-exponential
    $$k(h=|x-y|)=e^{-\left(\frac{h}{\theta}\right)^p}$$
  - Matern 3/2
    $$k(h=|x-y|)=\left(1+\sqrt{3}\frac{h}{\theta}\right)e^{-\sqrt{3}\frac{h}{\theta}}$$
  - Matern 5/2
    $$k(h=|x-y|)=\left(1+\sqrt{5}\frac{h}{\theta}+\frac{5}{3}\left(\frac{h}{\theta}\right)^2\right)e^{-\sqrt{5}\frac{h}{\theta}}$$



Focus on stochastic optimization

Related [R] packages

DiceOptim

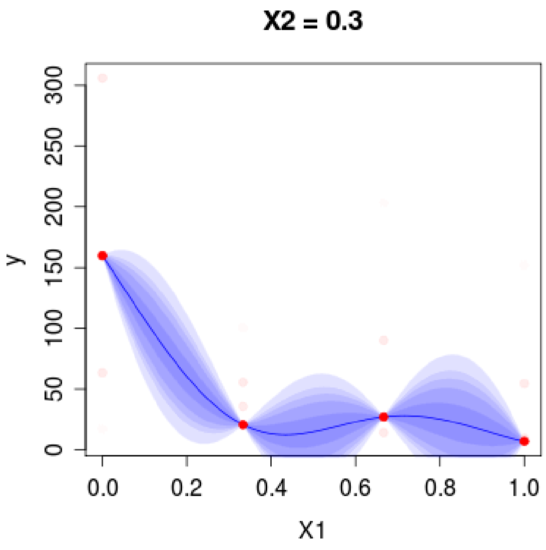
- Expected Improvement (local) criterion
  - Low cost criterion (compared to global ones)
  - Maybe extended to noisy kriging
- Parallel maximization of EI based on « Constant Liar » heuristic
  - min/max/mean/kriging mean/...
  - Used as a tunning parameter for global/local optimization
- Integrated / Decoupled call of cost function
  - Suitable for « computer experiments » framework

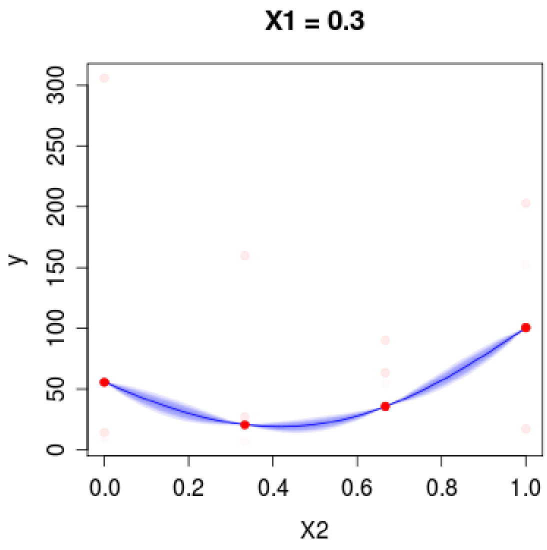
Focus on stochastic optimization

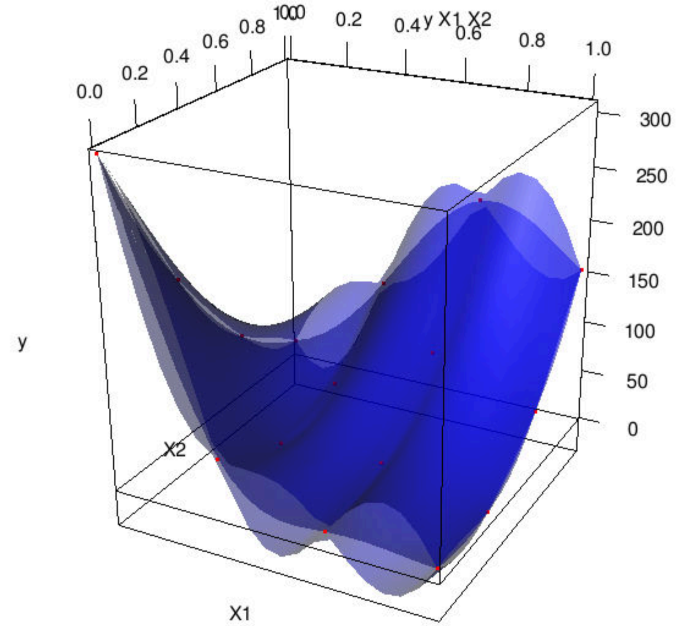
↗ Related [R] packages

- DiceView

```
d <- 2; n <- 16
design.fact <- expand.grid(seq(0, 1, length = 4), seq(0, 1, length = 4))
design.fact <- data.frame(design.fact); names(design.fact) <- c("x1", "x2")
y <- branin(design.fact)

m <- km(design = design.fact, response = y)

sectionview(m, center = c(.3, .3))

sectionview3d(m)
```







Focus on stochastic optimization

↗ Integration of stochastic simulator

Controlled convergence heuristic for MC criticality code

- MC code => sd estimator of Gaussian output (k-effective) is available
  (each MC sample increase, for instance)
- Code endpoint may be
  - Sample size :(
  - Estimator sd target
  - Estimator quantile (0.999) target
- Early endpoint reached when k-effective 99.9%-quantile << 1.0

OR

- Early endpoint reached when k-effective 99.9%-quantile << current max.

Resource-constraint optimization using simulator fidelity

- On-line control of experiments fidelity using « Start & Stop »
- Quantile based Expected Improvement
- *To be published in Technometrics 2011*

- **Promethee** workbench http://www.irsn.fr/promethee

- Dice* kriging packages (http://dice.emse.fr/)

DiceKriging
http://cran.r-project.org/web/packages/DiceKriging/index.html

DiceOptim
http://cran.r-project.org/web/packages/DiceOptim/index.html

DiceView
http://cran.r-project.org/web/packages/DiceView/index.html