

MTK (Mexico Tool Kits) : un package R pour la gestion de l'exploration numérique des modèles

Juhui WANG
MIA-Jouy, INRA

Plan

- *Contexte de travail*
 - *Fil conducteur*
 - *Objectifs : facilité, généricité, extensibilité et interopérabilité.*
- *Le package « mtk »*
 - *Architecture orientée-objet et ouverte au Web computing*
 - *Fonctionnalités : utilisation interactive, intégration avec des plate-formes de simulation, et intégration des contributions tierces.*
- *Exemples*
 - *Utilisation interactive*
 - *Utilitaires d'extension : add-ons*

Contexte du travail

- Méthodes existantes sont :
 - Abondantes
 - Diversifiées
- Difficultés :
 - Utilisation
 - Comparaison
 - Unification

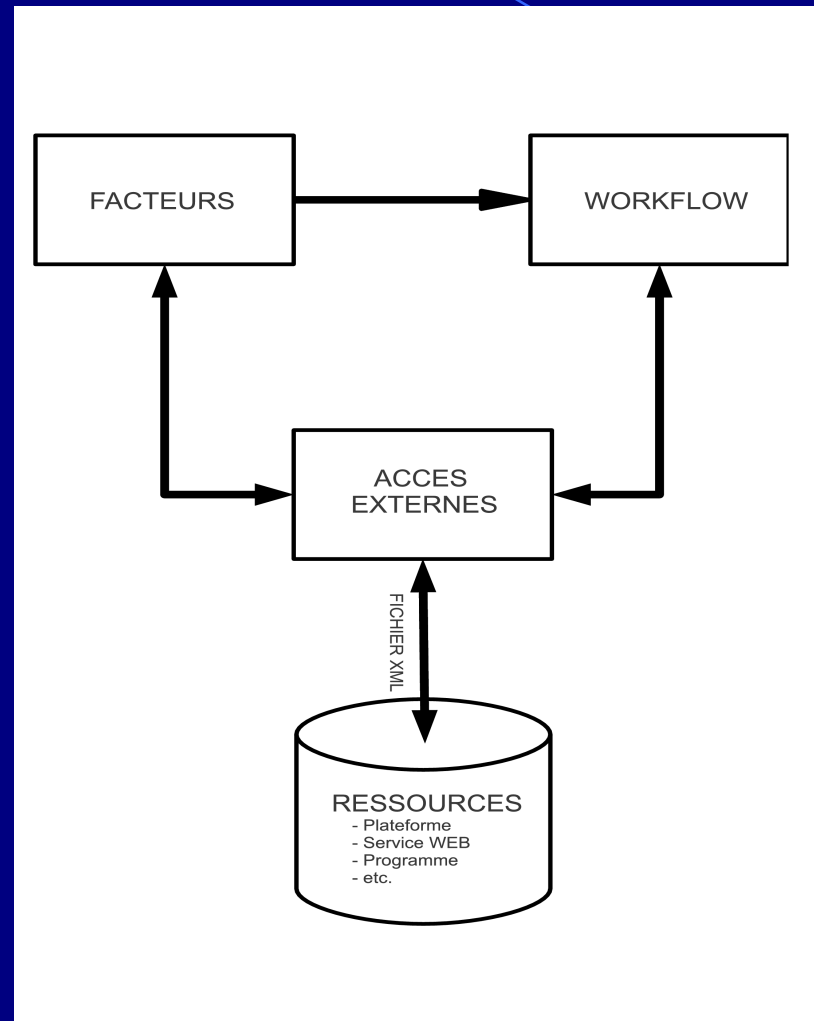
Structuration en une suite de traitements :

- Factor design
- Experiment design
- Evaluation
- Analysis
- Reporting
- + transformer (filter)

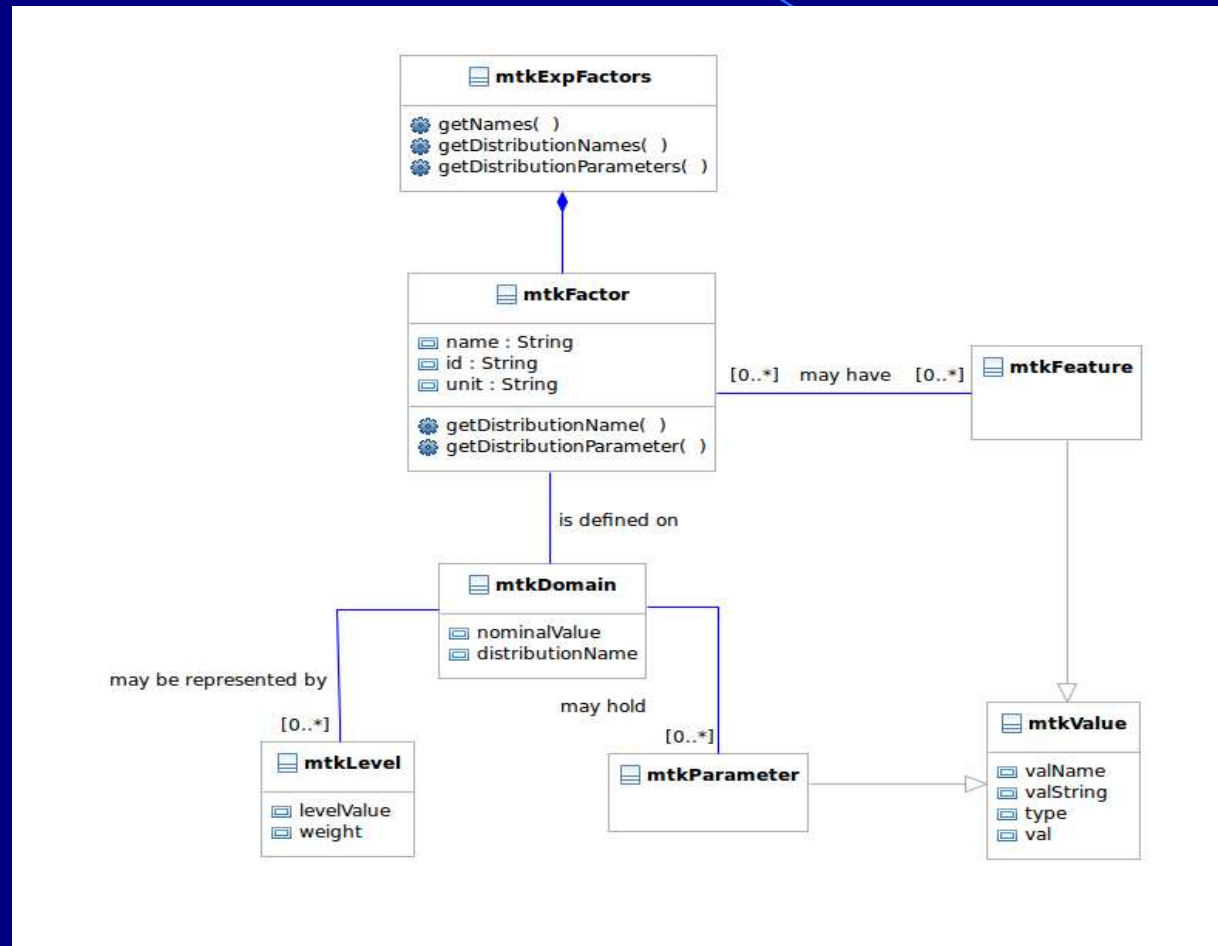
Objectifs

- *Facilité :*
 - Synthaxe d'utilisation générique
- *Généricité :*
 - *Encapsulation uniforme des méthodes existantes.*
- *Extensibilité :*
 - Intégration facile des nouvelles méthodes réalisées en R
- *Interopérabilité*
 - Intégration transparente avec les plates-formes existantes

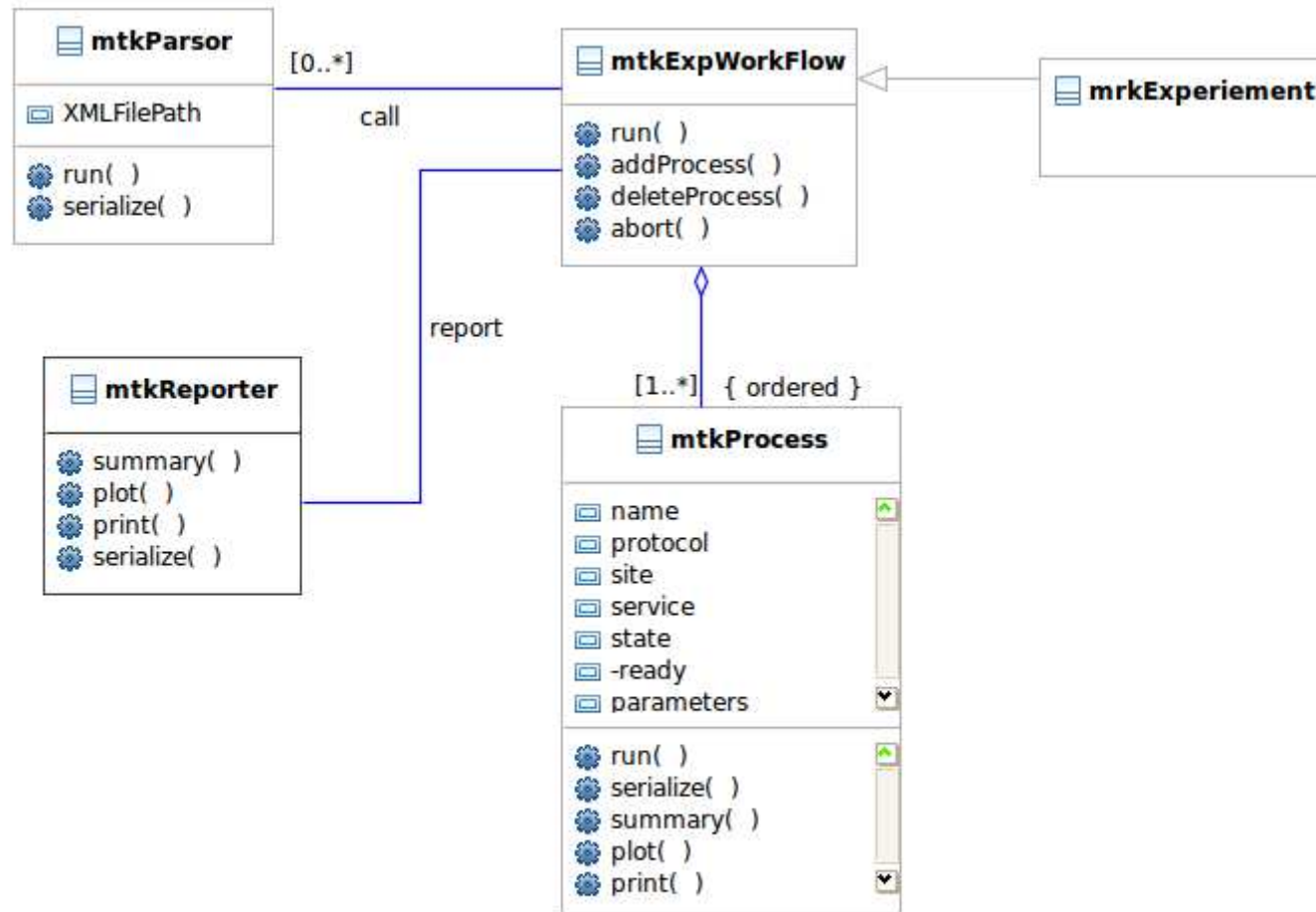
Architecture « mtk »



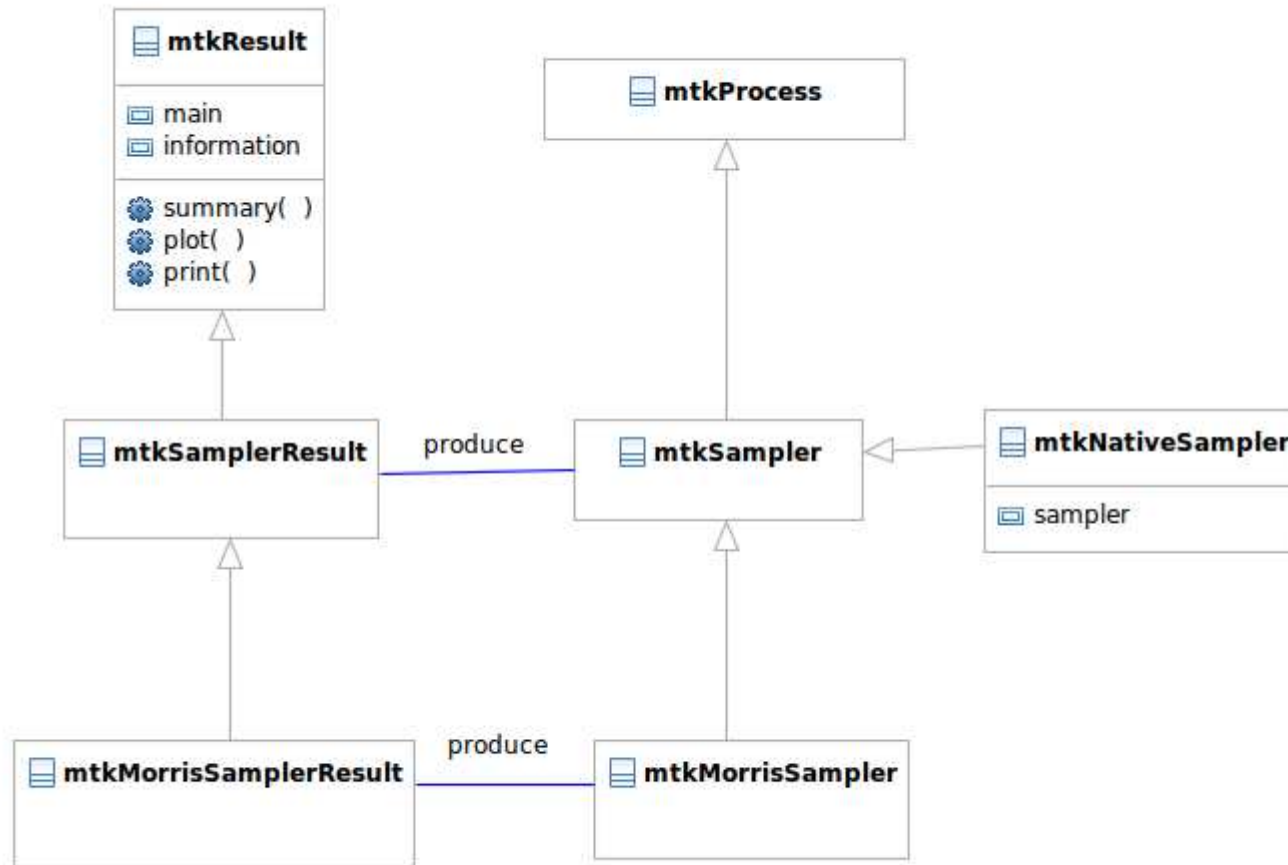
Conception (1)



Conception (2)



Conception (3)



Functionalités

- *Utilisation interactive au sein d'une session R*
- *A partir d'un fichier XML*
- *Intégration des contributions des tiers.*

Utilisation interactive

- *Spécifier les facteurs*
 - *nom, nature, etc*
 - *domaine d'incertitude : nominal, loi de distribution, etc.*
- *Former les processus qui réalisent les traitements : design, evaluation, analysis*
 - *design -----> sampler -----> experiment design*
 - *evaluation -----> simulator -----> model output*
 - *analysis -----> analyser -----> sensitivity index*
- *Construire le workflow*
 - *former une chaîne de traitement à partir des processus définis.*
- *Exécuter le workflow et reporter les résultats*
 - *run*
 - *summary, print, plot, show, report.*

Utilisation interactive

- *Spécifier les facteurs*

```
x1 <- make.mtkFactor(name="x1", distribName="unif",  
  distribPara=list(min=-pi, max=pi))  
x2 <- make.mtkFactor(name="x2", distribName="unif",  
  distribPara=list(min=-pi, max=pi))  
x3 <- make.mtkFactor(name="x3", distribName="unif",  
  distribPara=list(min=-pi, max=pi))  
ishi.factors <- mtkExpFactors(list(x1,x2,x3))
```

- *Former les processus qui réalisent les traitements : design, evaluation, analysis*

```
echantillonneur <- mtkNativeSampler("BasicMonteCarlo",  
  information=list(size=20) )  
simulateur <- mtkNativeSimulator("Ishigami" )  
analyseur <- mtkNativeAnalysor("Regression",  
  information=list(nboot=20) )
```


A partir d'un fichier XML

- *Générer le fichier XML selon le schéma « expDesign.xsd »*
 - *par une plate-forme*
 - *MED (Mexico EDitor)*
 - *à la main*
- *Construire le workflow à partir du fichier XML*
 - *parser le fichier XML afin d'extraire les informations sur les traitements à appliquer.*
 - *construire les processus de traitement*
 - *former le workflow*
- *Exécuter le workflow et reporter les résultats*
 - *run*
 - *summary, print, plot, show, report.*

Un fichier XML

Voir Fichier Ishigami_moris.xml

A partir d'un fichier XML

- *Construire le workflow*

```
exp <- mtkExpWorkflow(xmlFilePath =  
                        "./ishigami_morris.xml")
```

- *Exécuter le workflow et reporter les résultats*

```
run(exp)
```

```
summary(exp)
```

Contribution des tiers

- *Fonctions R*

- *Utilitaires permettant de transformer, en classe « mtk », les nouvelles méthodes programmées sous formes de fonction R*

*mtk.samplerAddons(), mtk.simulatorAddons(),
mtk.analysorAddons()*

- *fonction R doit respecter certaines règles dans sa définition*

- *Méthodes publiées sous forme de Web Services*

- *à faire*

mtk.samplerAddons()

function(factors, distribNames, distribParameters, ...)

- the parameter "factors" is a number or a list managing the names of the factors, the parameter "distribNames" is a list managing the names of the distributions of the factors, and the parameter "distribParameters" is a list of parameters associated with the distributions.
- the function returns a named list with two elements: «main » and « information ». The element "main" is a data.frame holding the results of the experiment design and the element "information" is a named list holding complementary information about the design.

mtk.samplerAddons()

- Définition :
 - `mtk.samplerAddons = function(where=NULL, library=NULL, authors="", name="", main="", summary=NULL, plot=NULL, print=NULL)`
- Exemple :
 - `mtk.samplerAddons(where="samplerSobol.R", authors="H. Monod, INRA-MIA Jouy en Josas", name="HMSobol", main="sampler.sobol", plot="plot.sobol", summary="summary.sobol")`
 - `sampleur <- mtkNativeSampler("HMSobol", information=list(N=200, shrink=0.8))`

Conclusion

- Projet en cours :
 - Toutes les contributions et critiques sont les bienvenues.
- *La 1ère version est disponible aux adresses :*

*[https://mulcyber.toulouse.inra.fr/frs/?
groupe_id=105&release_id=487](https://mulcyber.toulouse.inra.fr/frs/?groupe_id=105&release_id=487)*

www.reseau-mexico.fr/MTK