

# Projet Génie Logiciel

## **Partie 2 :** Extensions du modèle de processus

Philippe Leleux  
Jessica Hornik

# Tâche 1 : Métamodèle Simple PDL

## > Décomposition des activités :

- => Création SubWorkDefinition

- => Besoin de spécialiser les paramètres

  - => **ParameterWD** et **ParameterSWD**

- => Besoin d'accéder à une WD ou une SWD sans distinction

  - => Création d'**Activities** abstraite

- => idem pour les Parameter

  - => Création de **Parameter** abstraite

# Tâche 1 : Métamodèle PetriNet

=> Presque aucun changement

=> ajout de l'ArcKind : inhibitor

**APERCU DES MODELES**

# Tâche 2 : Projet GMF

- > **Parameter, Resource** : Node  
Lien entre la WD et le Parameter (- sale)
- > **Améliorations** : attributs en colonne
- > **Problèmes** : manip inconnues  
A chaque modif, tout recommencer

# Tâche 3 : xText

Syntaxe simple héritée de celle du TP

- > paramètres définis directement avec les Wds
- > ajout de lignes pour l'enregistrement des Resources

# Exemple synthaxe textuelle xText

```
process Developpement [20,50] {  
  wd Conception [10,16] concepteur:2 machine:2  
  wd RedactionDoc [8,12] redacteur:1 machine:1  
  wd Development [12,14] developpeur:2 machine:3  
  wd RedactionTest [10,12] testeur:1 machine:2  
  ws finishToFinish from Conception to RedactionDoc  
  ws startToStart from Conception to Development  
  ws startToStart from Conception to RedactionTest  
  ws finishToFinish from Development to RedactionTest  
  res concepteur contain 3  
  res redacteur contain 1  
  res developpeur contain 2  
  res testeur contain 2  
  res machine contain 4  
}
```

# Tâche 4 et 8 : Contraintes OCL

## > Contraintes normales :

Existence des attributs

Unicité des noms

ContainmentConsistency

Bon Process/PetriNet

## > Temps : $0 \leq \text{min\_time} \leq \text{max\_time} \leq p.\text{max\_time}$

Resource :  $\text{nomResource} = \text{nomParameter}$

$\text{Marking} \geq 0$

$\text{nbNeeds} \leq \text{marking Resource}$

## > Tests ?

# Tâches 5 : SimplePDL2PetriNet

- > **WorkDefinition** : normal + prise en compte du temps → réseau annexe + read\_arc
- > **Resource** : 1 place unique (init)
- > **Parameter** : 2 arcs pondérés



# Tâche 6 et 7 : Petri2Tina et Validation

- > Même chose qu'au TP
- > Long à coder
- > Validation grâce à PetriNet2Tina

**VALIDATION : 1°) 1Wd1Ressource  
2°) 2 étages  
3°) Développement**