

## C7 - UML Temps-Réel

### UML pour les systèmes embarqués et le temps réel

#### Objectifs

- Mettre en évidence les mécanismes de base liés aux systèmes embarqués et au temps réel
- Mettre en pratique le langage UML avec les extensions temps réel dans les différentes phases d'analyse, de conception et de codage d'une application industrielle

*Durant le cours, les stagiaires développeront un modèle complet sous Papyrus/l'outil UML gratuit, basé sur Eclipse ou, sur demande, Enterprise Architect Professional Edition à partir du cahier des charges d'un système embarqué réaliste.*

#### Matériel

- Un PC par binôme équipé d'un AGL
- Support de cours
- CDROM avec documentation et exercices corrigés

#### Pré-requis

- Connaissances de base des systèmes embarqués et temps-réel, de leur conception et de leur programmation.

#### Environnement du cours

- Cours théorique
  - Support de cours imprimé et au format PDF (en anglais).
  - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique.
- Au début de chaque demi-journée une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

#### Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus.

#### Modalités d'évaluation

- Les prérequis indiqués ci-dessus sont évalués avant la formation par l'encadrement technique du stagiaire dans son entreprise, ou par le stagiaire lui-même dans le cas exceptionnel d'un stagiaire individuel.
- Les progrès des stagiaires sont évalués par des quizz proposés en fin des sections pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, une attestation et un certificat attestant que le stagiaire a suivi le cours avec succès.
  - En cas de problème dû à un manque de prérequis de la part du stagiaire, constaté lors de la formation, une formation différente ou complémentaire lui est proposée, en général pour conforter ses prérequis, en accord avec son responsable en entreprise le cas échéant.

## Plan

### **Introduction au temps réel**

- concepts temps réel de base
- contraintes particulières du temps réel
- programmation structurée et objet
- apports des techniques objets

### **L'approche objet avec UML**

- genèse d'UML
- modèles UML standards
- cycle de développement Objet

### **L'approche temps réel avec UML**

- contraintes liées à l'interprétation des diagrammes
- interprétation des diagrammes
- définition de nouveaux diagrammes
- cycle de développement avec RT UML

## Modélisation

### **Le langage de modélisation UML**

- Modélisation statique
  - cas d'utilisation
  - modèles de classes
- Modélisation dynamique
  - diagrammes de Séquence
  - diagrammes de Collaboration
  - diagrammes Etats Transitions
- Etendre UML
  - Accéder au meta-modèle d'UML
  - Créer des stéréotypes
  - Créer des profils

### **Les extensions pour le temps réel**

- environnement / diagramme de contexte système
- contraintes / diagramme de contraintes
- comportement / diagramme d'états
- timings / diagramme de séquence étendu
- parallélisme / diagramme architecture logicielle
- architecture / diagramme architecture matérielle

### **Le profil MARTE**

- Présentation générale
- Les concepts de MARTE
- Gestion du temps avec MARTE

## **Spécification système**

### **L'aspect statique**

- spécification du diagramme de contexte
- formalisation des contraintes non fonctionnelles
- description des cas d'utilisation
- identification des classes de haut niveau
- ébauche du modèle de classes
- itération et affinage pour chaque cas d'utilisation

### **L'aspect dynamique**

- formalisation des cas d'utilisation par les "scénarios"
- ajout des aspects temporels dans le diagramme de séquence
- comportement du système et Diagramme Etat transition
- affinage du modèle objet avec les opérations
- ajout des objets d'interface dans les Diagrammes de séquences
- création du modèle Objet d'Interface

## **Conception système**

### **L'affinage des modèles de spécification**

- organisation du système en sous systèmes et packages
- description dynamique des classes
- diagramme de comportement de chaque classe
- affinage des diagrammes de séquence
- affinage du modèle de classes
- utilisation de bibliothèques d'Interface Homme Machine (GUI)
- intégration des Interfaces d'entrée/sortie
- prise en considération des objets de stockage

### **La gestion multiprocess et multitâches**

- différents types d'architectures logicielles
- identification des tâches
- allocation des sous-systèmes aux processeurs et aux tâches
- communication inter-process
- synchronisation
- élaboration du diagramme architecture logicielle

### **L'architecture système**

- partitionnement en sous système matériels (cartes, machines, CPU...)
- types d'architectures matérielles
- optimisation des choix d'architecture
- définition des Interfaces systèmes internes (bus, liaisons, protocoles...)
- modélisation du système à travers le diagramme d'architecture matérielle

### **L'environnement d'exécution des tâches**

- problèmes liés au choix de l'exécutif
- interprétation des mécanismes temps réel en fonction du type d'architecture

## Conception détaillée et Codage

### La préparation au codage

- description détaillée des méthodes et attributs
- critères d'optimisation
- affinage de l'héritage
- classes abstraites, template...
- associations et pointeurs

### Le codage incrémental

- implémentation des objets de stockage
- UML et les langages de programmation
  - passage UML -> C
  - passage UML -> C++
  - passage UML -> Java
- codage d'un incrément
- test de l'incrément
- validation de l'incrément
- livraison

## Renseignements pratiques

**Durée : 4 jours**  
**Prix : 2490 € HT**