



Le langage SysML

publié le 22/07/2016 - mis à jour le 22/08/2016

Descriptif :

Objectifs du langage SysML

Sommaire :

- Diagrammes SysML
 - Diagramme périphérique (non SysML)
-

Les **systèmes pluri-technologiques** actuels sont souvent d'une grande complexité tant au niveau **fonctionnel** que **structurel**. Leur **conception** est le fruit d'une collaboration entre différents acteurs (ingénieurs, techniciens, juristes, ergonomes, métiers du marketing, de la communication ...).

Afin de collaborer efficacement, l'ensemble de ces acteurs doit disposer d'un langage commun unique et fédérateur tel que le **langage SysML**.

A la fois simple à comprendre et suffisamment développé, le langage **SysML** est un langage graphique de modélisation spécifique au domaine de l'**ingénierie système** permettant de spécifier, d'analyser, de concevoir, de vérifier ... un **système pluri-technologique complexe**.

Véritable outil de communication et de suivi par toutes les **parties prenantes** du projet, le langage **SysML** permet de définir les **exigences**, le service rendu, l'aspect **comportemental** ainsi que **structurel** du système à concevoir. Il permettra en outre la traçabilité du projet tout au long de son avancement.

Son utilisation est également appropriée en phase de **rétro-ingénierie** pour reconcevoir un système existant. Dans une perspective d'innovation, il est d'autant plus important de disposer d'une solide connaissance du système afin d'éviter le "refaire plutôt que corriger" ainsi qu'une banque de modèles pour gagner en efficacité.

● Diagrammes SysML

Grâce à l'étendue de sa notation, le **langage SysML** permet de représenter graphiquement sous forme de diagrammes :

- Les **exigences** auxquelles le système doit satisfaire (point de vue transversal)
- Le **comportement attendu** du système, ses phases de fonctionnement et ses interactions avec l'environnement extérieur (point de vue comportemental)
- L'**architecture structurelle** du système (point de vue structurel)

Le **langage SysML** s'articule autour de neuf diagrammes selon ces trois points de vue.

○ Point de vue transversal

- Le diagramme d'exigence (Requirement Diagram) **req**

○ Point de vue comportemental

- Le diagramme des cas d'utilisation (Use Case Diagram) **ucd**
- Le diagramme de séquence (Sequence Diagram) **sd**

- Le diagramme d'états (State Machine Diagram) **stm**
- Le diagramme d'activité (Activity Diagram) **act**

○ Point de vue structurel

- Le diagramme de blocs (Block Definition Diagram) **bdd**
- Le diagramme de blocs internes (Internal Block Diagram) **ibd**
- Le diagramme paramétrique (Parametric Diagram) **par**
- Le diagramme de paquetage (Package Diagram) **pkg**

● Diagramme périphérique (non SysML)

Au delà de ces 9 diagrammes on s'attachera à réaliser un **diagramme de contexte** qui n'est pas normalisé **SysML**. Cependant il n'en est pas moins incontournable pour définir les **frontières** du système et pour préciser les différents acteurs ainsi que l'environnement dans lequel le système évolue.



**Académie
de Poitiers**

Avertissement : ce document est la reprise au format pdf d'un article proposé sur l'espace pédagogique de l'académie de Poitiers.

Il ne peut en aucun cas être proposé au téléchargement ou à la consultation depuis un autre site.