

# Introduction à SysML

Une démarche d'architecture système



# Table des matières

- Introduction générale
- 1** Fondements de SysML
- 2** Structure d'un modèle système en SysML
- 3** Exercice de modélisation SysML
- 4** Ingénierie de la modélisation SysML
- Conclusion

# Table des matières

Introduction générale

**1** Fondements de SysML

**2** Structure d'un modèle système en SysML

**3** Exercice de modélisation SysML

**4** Ingénierie de la modélisation SysML

Conclusion

# Introduction

## PHILOSOPHIE



**SysML** n'est pas simplement un outil informatique ...

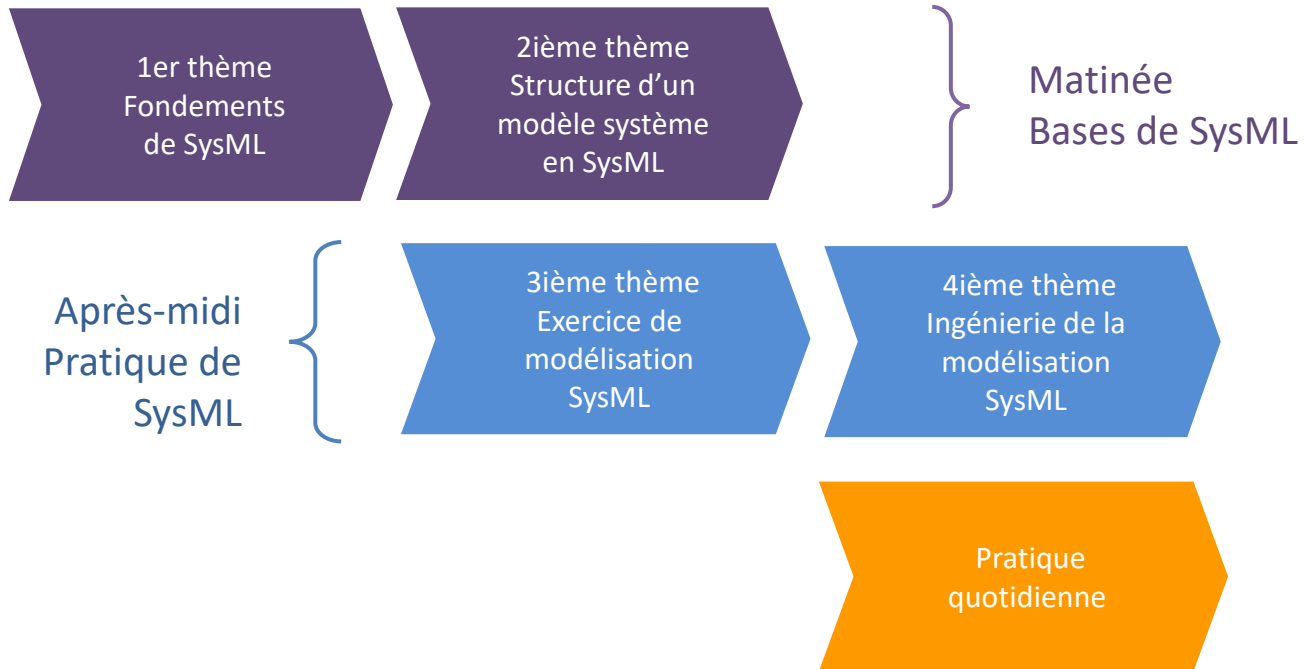


... mais d'abord un **standard international** conçu pour permettre de mettre concrètement en oeuvre une démarche d'**architecture système** !

Une formation **pragmatique** basé sur un mix de **concepts systèmes & SysML** et d'**études de cas**

# Introduction

## STRUCTURE



# Introduction

## OBJECTIFS

- **Pourquoi** une formation à SysML ?

Cette *formation* a pour but premier de donner rapidement une bonne vue d'ensemble du langage de l'architecture système qu'est SysML

- **De quoi** est il question ?

- Connaître les fondements de SysML tant au niveau des objets & des concepts de base que l'on manipule avec ce langage que du point de vue systémique (éléments d'architecture système)
- Comprendre comment organiser & structurer un modèle système en SysML (en s'appuyant notamment sur un cas tiré de votre métier)
- Mieux comprendre la manière dont on peut utiliser SysML au sein d'un processus intégré de modélisation système

- **De quoi** n'est il pas question ?

- Devenir un expert en SysML
- Maîtriser l'utilisation et le déploiement de SysML dans le cadre d'un processus industriel de conception

# Table des matières

■ Introduction générale

**1** Fondements de SysML

**2** Structure d'un modèle système en SysML

**3** Exercice de modélisation SysML

**4** Ingénierie de la modélisation SysML

■ Conclusion

# Table des matières

## 1 Fondements de SysML

*sous-thème* Introduction à l'approche système

*sous-thème* Historique & objectifs de SysML

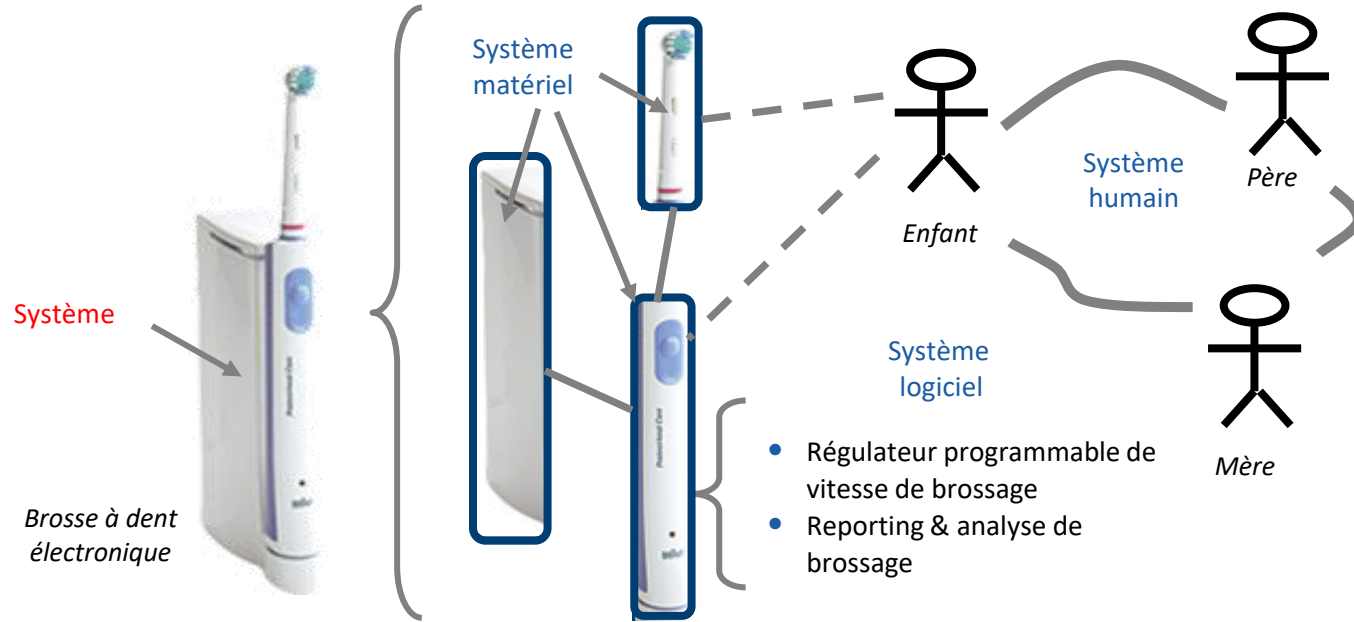
*sous-thème* Principaux objets & vues SysML

*sous-thème* Synthèse : pourquoi utiliser SysML ?



# Fondements de SysML – Système

## LA NOTION DE SYSTÈME



**Système** : un ensemble de **systèmes** (matériels, logiciels & humains) organisés de manière à ce que leur **intégration** permette d'accomplir – au sein d'un environnement donné – les **missions** pour lesquelles il a été conçu

# Fondements de SysML – Système

## LA NOTION DE POINT DE VUE

Perceptions  
des acteurs



Perception féminine ou  
maternelle



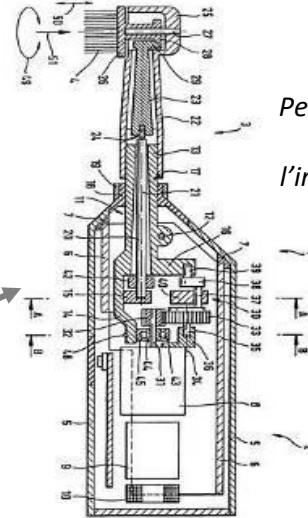
Perception du dentiste



Perception du businessman



Perception de l'utilisateur  
standard

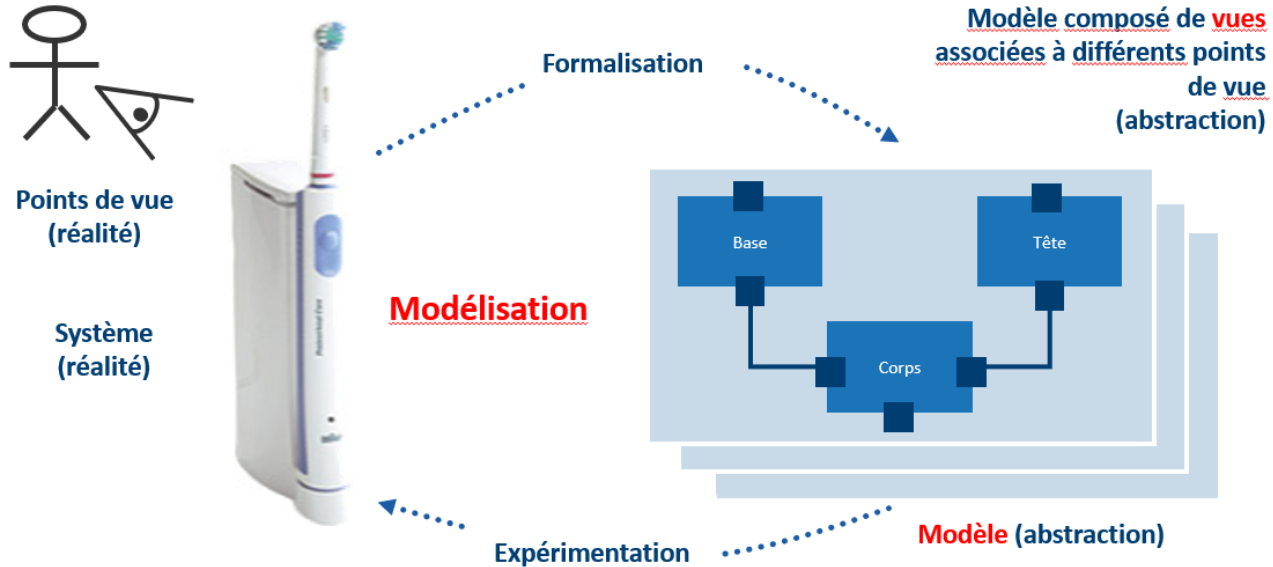


Perception  
de  
l'ingénieur

Point de vue : **perception** (nécessairement relative) d'un système par un **acteur**

# Fondements de SysML – Système

## LA NOTION DE MODÈLE SYSTÉMIQUE



Modèle systémique : une représentation formelle abstraite d'un système qui est souvent décomposée selon 3 grandes visions (présentées plus loin)

# Table des matières

## 1 Fondements de SysML

*sous-thème* Introduction à l'approche système

*sous-thème* Historique & objectifs de SysML

*sous-thème* Principaux objets & vues SysML

*sous-thème* Synthèse : pourquoi utiliser SysML ?

# Fondements de SysML – Objectifs

UN LANGAGE ORIENTÉ-MODÈLE ...

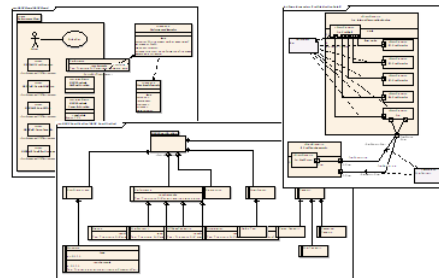
## Autrefois



*Ingénierie orientée-document*

- Spécifications systèmes
- Exigences systèmes
- Interfaces systèmes
- Architectures systèmes
- Vérification & validation
- Plans de test

## Demain



*Ingénierie orientée-modèle  
(Model Driven Engineering)*

**Génération automatique de documents**

SysML (System Modeling Language) est un **langage orienté-modèle** de la famille UML qui a été défini pour répondre aux **besoins des architectes systèmes** non couverts par UML 1

1 e.g. modélisation systémique, gestion des exigences, traçabilité

# Fondements de SysML – Objectifs

... COMMANDÉ PAR L'INCOSE

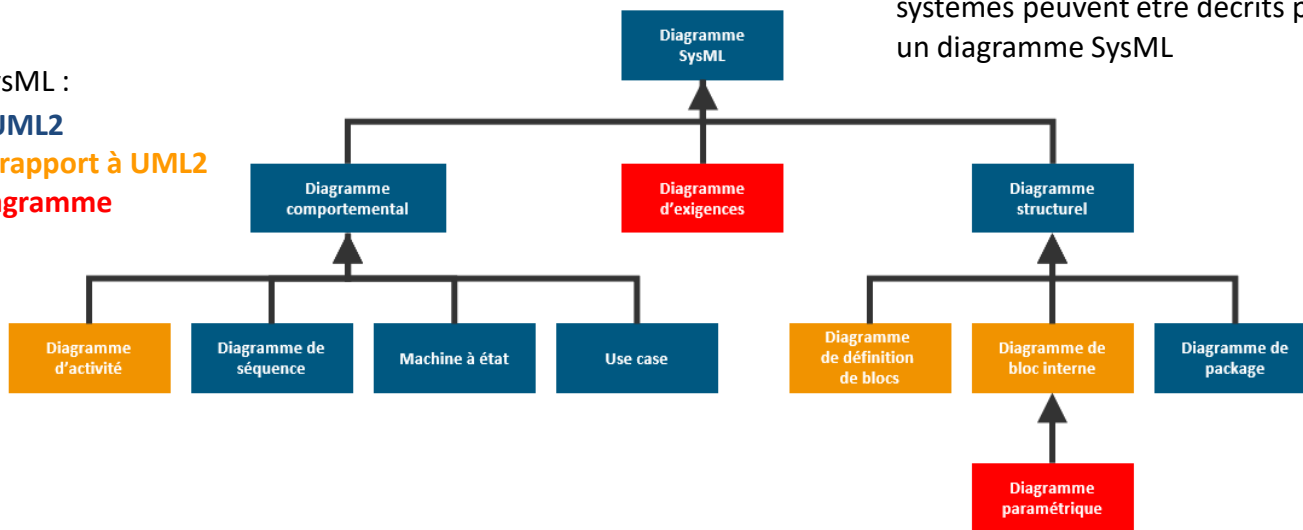
- **SysML** est né d'une **initiative de l'INCOSE** (International Council on Systems Engineering) :
  - Mars 2003 : “Request for Proposal” émise par l'OMG (Object Management Group) suite à une demande de l'INCOSE pour définir une version d'UML adaptée aux besoins de l'ingénierie système
  - Mai 2006 : publication officielle de la version 1.0 de SysML par l'OMG
  - Fin 2006 : premiers modeleurs SysML disponibles sur le marché
- **SysML** est une **réponse industrielle** conçue par un consortium formé de :
  - Systémiers : American Systems, BAE Systems, Boeing, Deere & Company, EADS-Astrium, Eurostep, Lockheed Martin, Motorola, NIST, Northrop Grumman, Oose, Raytheon, Thales
  - Editeurs : Artisan, Embedded plus, Gentleware, IBM, I-logix, Mentor graphics, PivotPoint, Sparx Systems, Telelogic, Vitech
  - Milieu académique : Georgia Institute of Technology

# Fondements de SysML – Objectifs

## LE PÉRIMÈTRE DE SYSML

Périmètre de SysML :

- Comme en UML2
- Modifié par rapport à UML2
- Nouveau diagramme

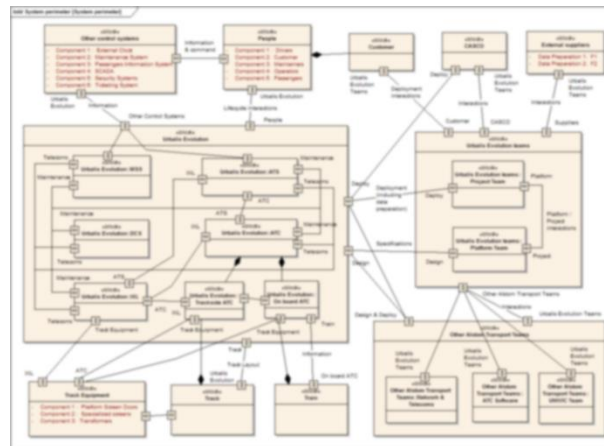


SysML est langage bien adapté aux **besoins de la modélisation système** : tous les concepts systèmes peuvent être décrits par un objet ou un diagramme SysML

- **SysML** est un **langage semi-formel** qui réduit l'ambiguïté de la description d'un système
  - Chaque objet SysML possède à la fois une sémantique (signification) & une syntaxe (mode de description) standardisée
- **SysML** est un **langage polyvalent intégré**
  - Il peut capturer les points de vue de toutes les parties prenantes d'un système
  - Il peut intégrer de manière cohérente les niveaux systèmes & sous-systèmes

# Fondements de SysML – Objectifs

LE BUT AVOUÉ : MAÎTRISER LA COMPLEXITÉ



Modèle de haut niveau d'un système de supervision et de son environnement systémique

Un modèle SysML permet de **maîtriser la complexité des architectures systèmes** en forçant l'ingénieur à **démêler leurs « spaghettis »** et en étant un support naturel au **partage** de ces architectures avec leurs parties prenantes



# Fondements de SysML – Objectifs

## CE QU'EST & N'EST PAS SYSML !

- **SysML** est un **langage**, pas un outil, ni une méthodologie
  - “Si je sais parler français, est ce que je sais écrire une pièce de théâtre ?”
- **SysML** est un **langage orienté système**, mais n’est bien sûr pas un langage “magique” qui garantit que vos systèmes seront “bien” spécifiés
  - On peut malheureusement faire du design horrible avec SysML !
- **SysML** est un **langage de modélisation systémique**, pas de simulation
  - Les interfaces entre les modeleurs SysML et les outils de simulation (et plus généralement tous les autres outils d’ingénierie) sont encore relativement limitées à date (bien que techniquement possibles)



- **Quelques autres limitations intrinsèques de SysML**
  - SysML est un langage qui fournit une syntaxe pour modéliser des systèmes. Mais la meilleure manière de s’en servir (la stratégie de modélisation) doit être adaptée aux besoins et aux contextes de chaque environnement industriel.
  - SysML ne peut pas être utilisé efficacement sans connaître les bases de l’approche et de l’architecture système

# Table des matières

## 1 Fondements de SysML

*sous-thème* Introduction à l'approche système

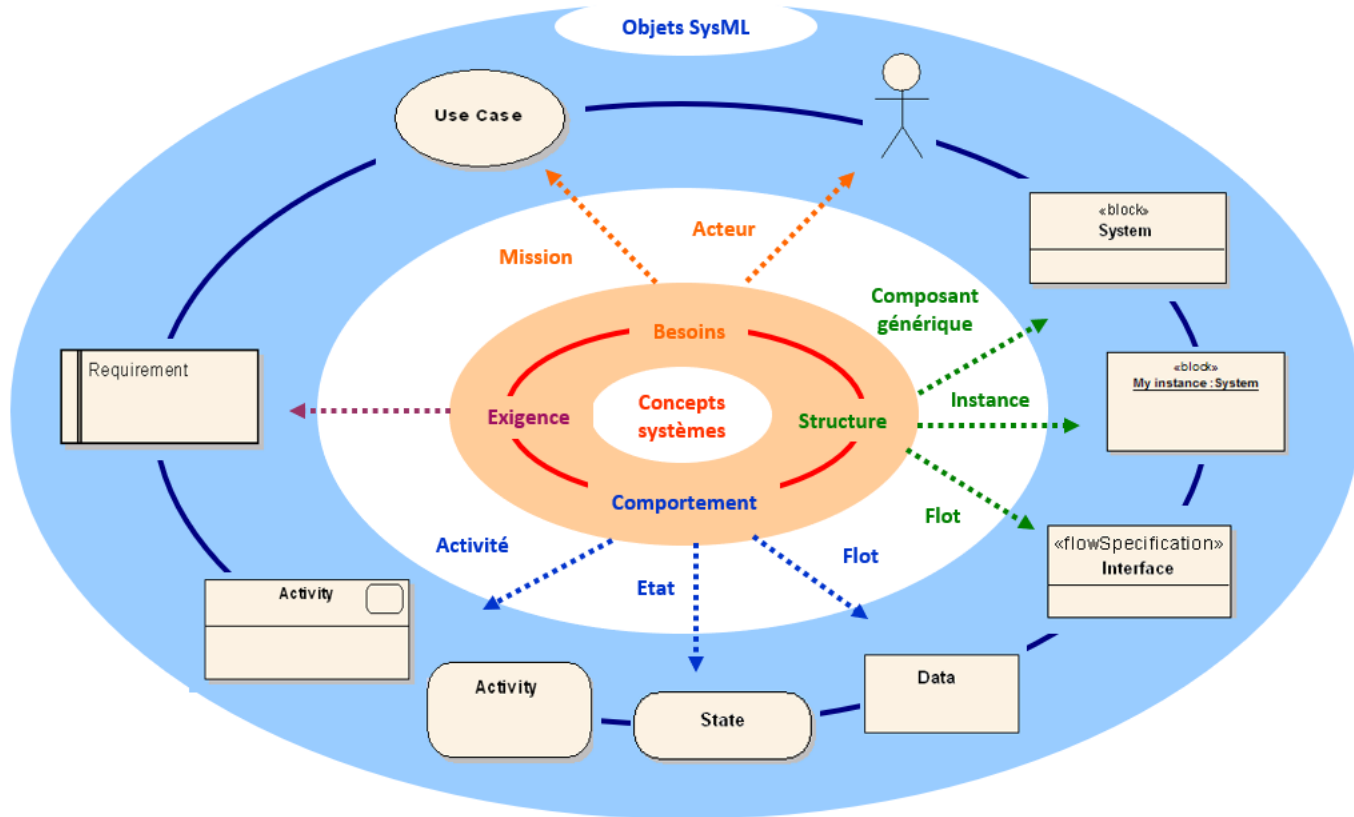
*sous-thème* Historique & objectifs de SysML

*sous-thème* Principaux objets & vues SysML

*sous-thème* Synthèse : pourquoi utiliser SysML ?

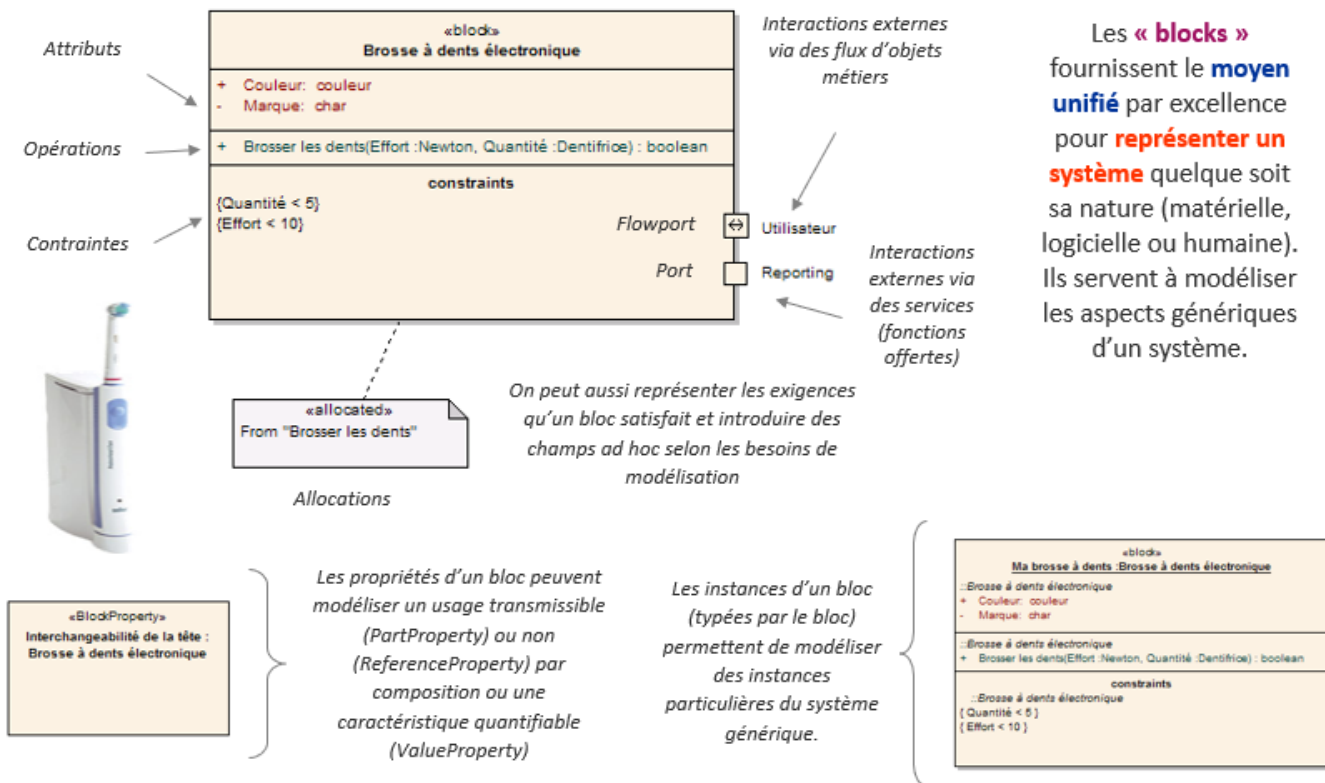
# Fondements de SysML – Objets

## PRINCIPAUX OBJETS SYSTM



# Fondements de SysML – Objets

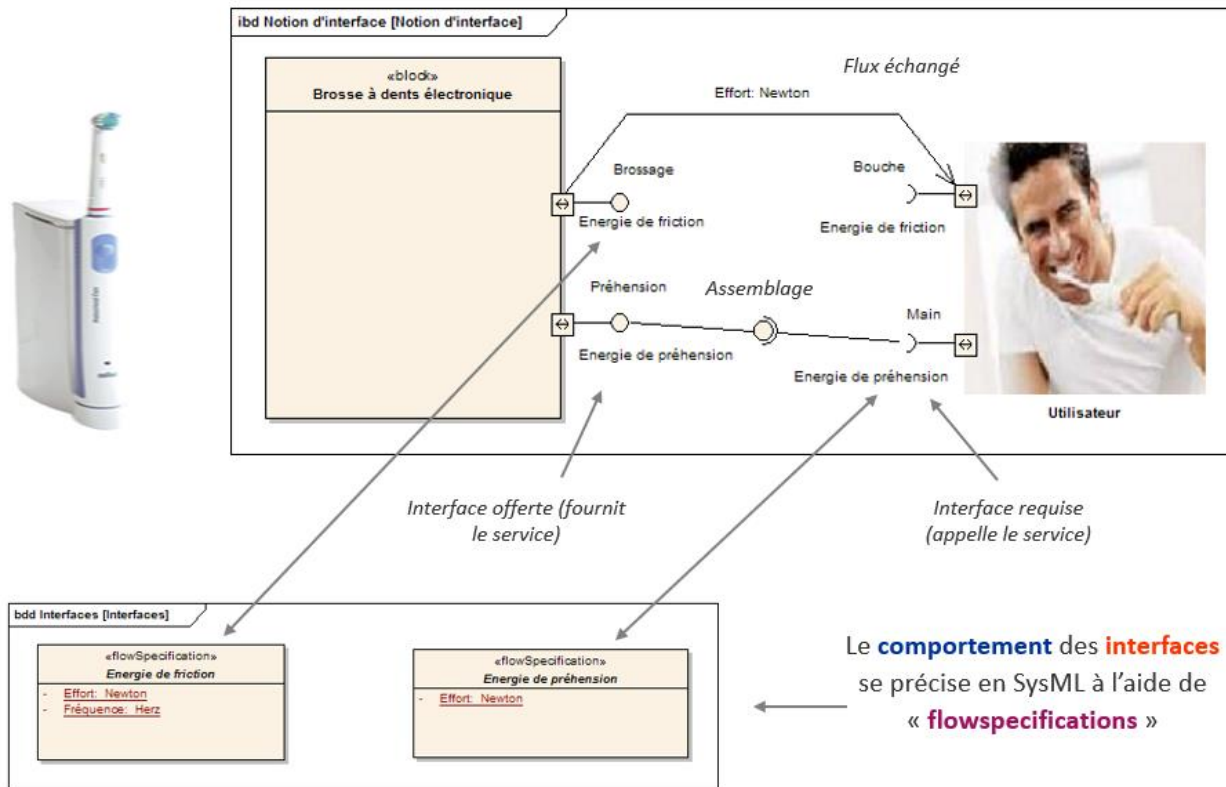
## L'OBJET FONDAMENTAL : LE BLOC SYSML



Les « **blocks** » fournissent le **moyen unifié** par excellence pour **représenter un système** quelque soit sa nature (matérielle, logicielle ou humaine). Ils servent à modéliser les aspects génériques d'un système.

# Fondements de SysML – Objets

## LA MODÉLISATION DES INTERFACES

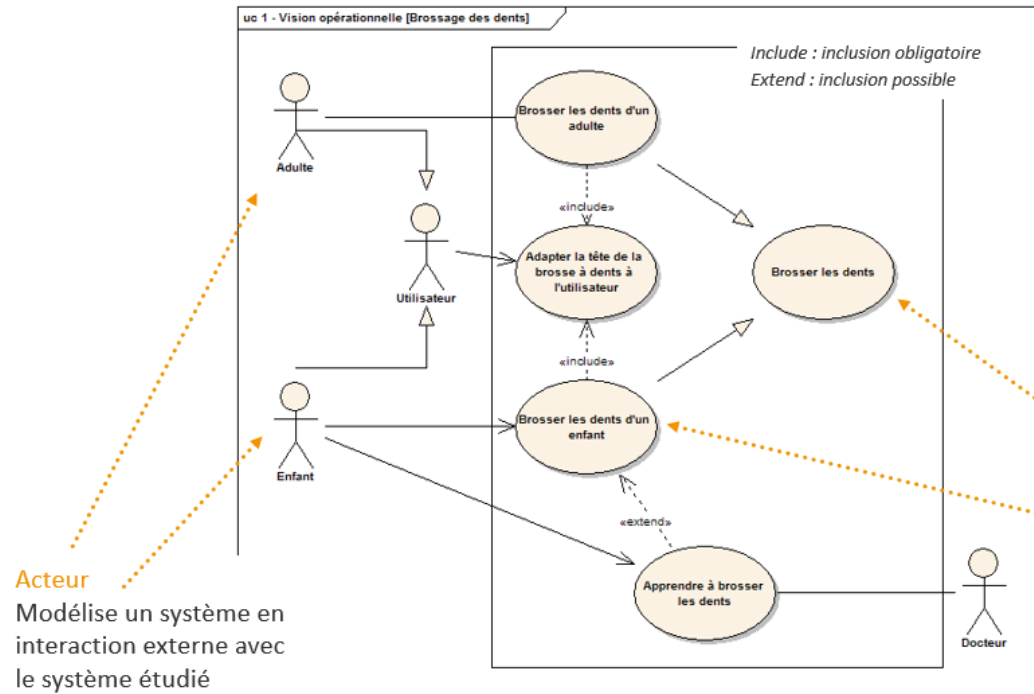


# Fondements de SysML – Vues

## ANALYSE DE BESOINS

### Use case

Modélise les missions et les contextes opérationnels d'un système



### Acteur

Modélise un système en interaction externe avec le système étudié

Use case  
Modélise un usage possible du système impliquant un ou plusieurs acteurs

# Fondements de SysML – Vues

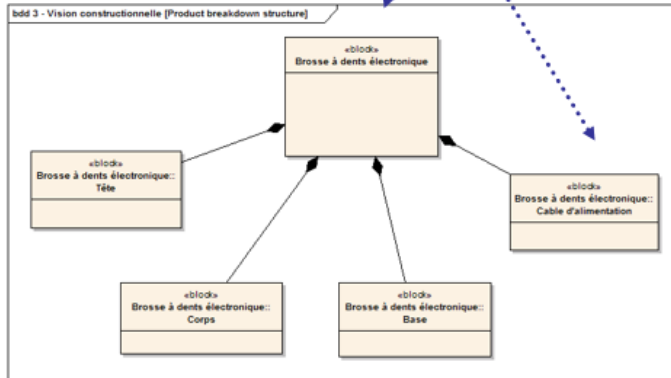
## VUES STRUCTURELLES

### Diagramme de définition de bloc

Modélise une organisation systémique  
(ici une décomposition organique)

Bloc

Modélise un composant  
systémique

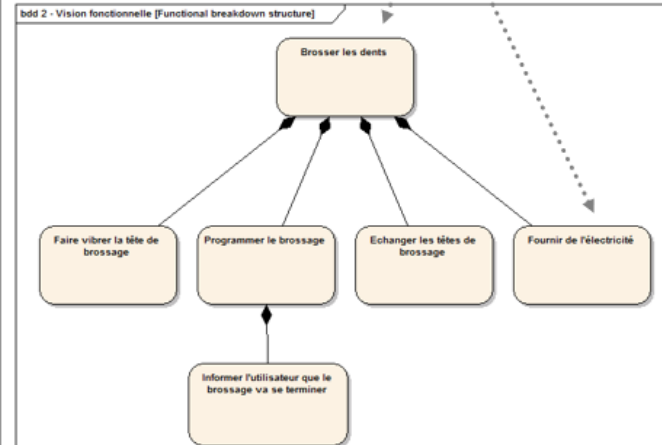


### Diagramme de définition de bloc

Modélise une organisation systémique  
(ici une décomposition fonctionnelle)

Activité

Modélise une fonction

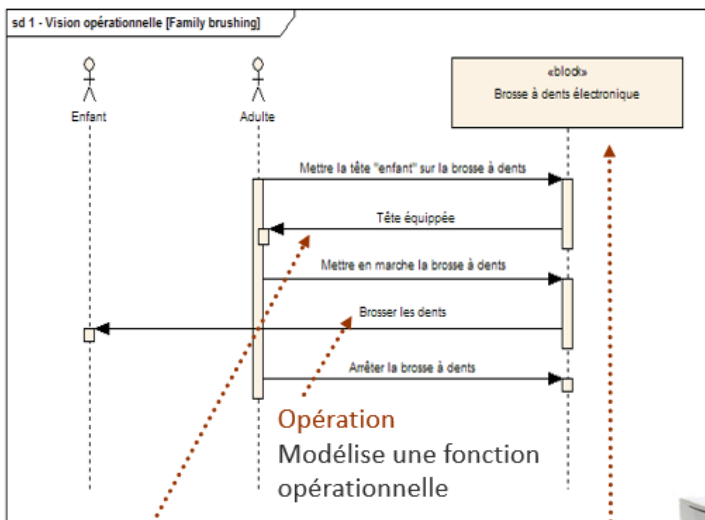


# Fondements de SysML – Vues

## VUES COMPORTEMENTALES (1)

### Diagramme de séquence

Modélise les interactions inter-composants



### Message

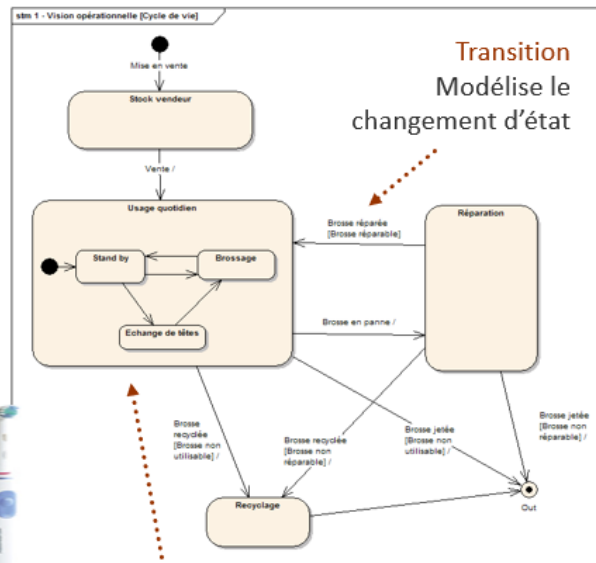
Modélise un échange d'énergie ou d'information entre deux composants systémiques

### Bloc

Modélise un composant systémique

### Diagramme d'état

Modélise l'évolution des états d'un système



Transition  
Modélise le  
changement d'état

### Etat

Modélise un contexte opérationnel, un mode fonctionnel ou une configuration



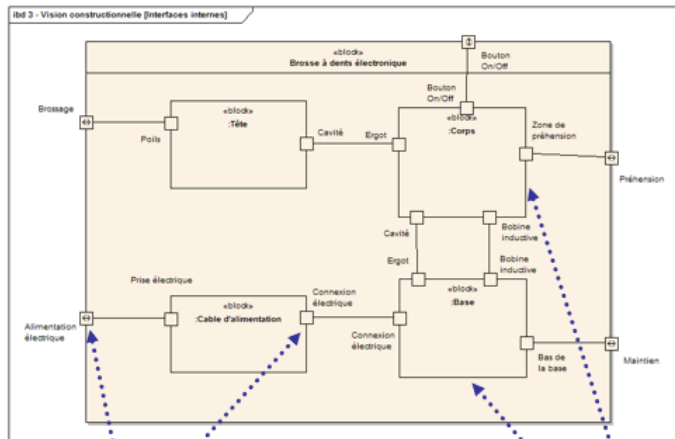


# Fondements de SysML – Vues

## VUES COMPORTEMENTALES (2)

### Diagramme de bloc interne

Modélise la dynamique interne d'un composant systémique (dynamique constructionnelle)

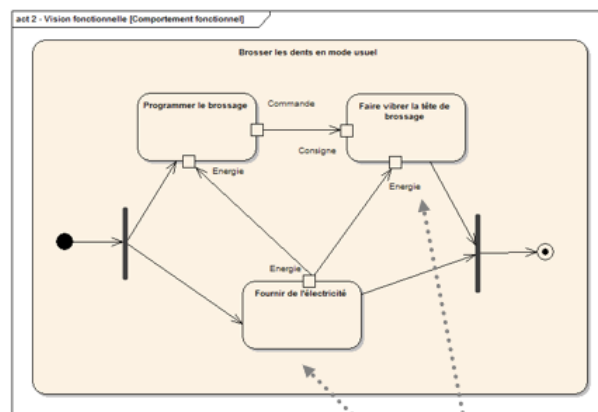


**Flow port**  
Modélise des points  
d'interaction

**Block instance**  
Instances de blocs  
modélisant des composants  
systémiques spécifiques

### Diagramme d'activité

Modélise un fonctionnement systémique (processus fonctionnel)



**Activité**  
Mode de modélisation  
d'une tâche élémentaire

# Fondements de SysML – Vues

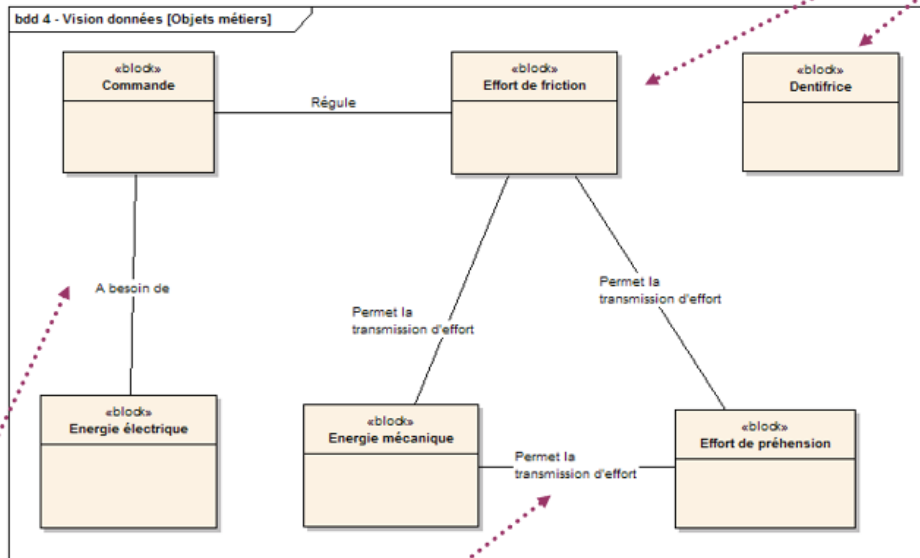
## OBJETS MÉTIERS

### Diagramme de bloc

Modélise les objets et les données métiers manipulés par le système

Bloc

Modélise un objet métier



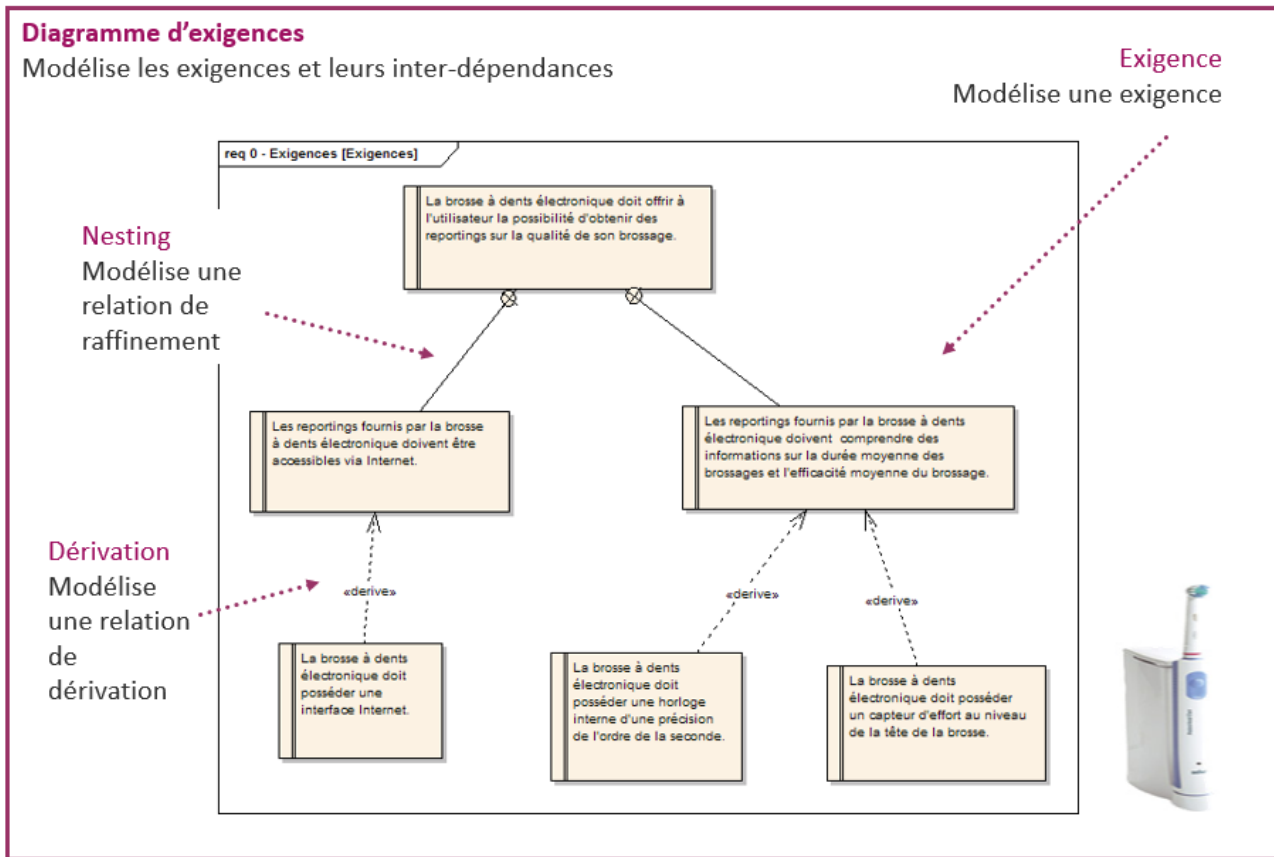
Association

Modélise une relation existant entre des objets métiers



# Fondements de SysML – Vues

## EXIGENCES

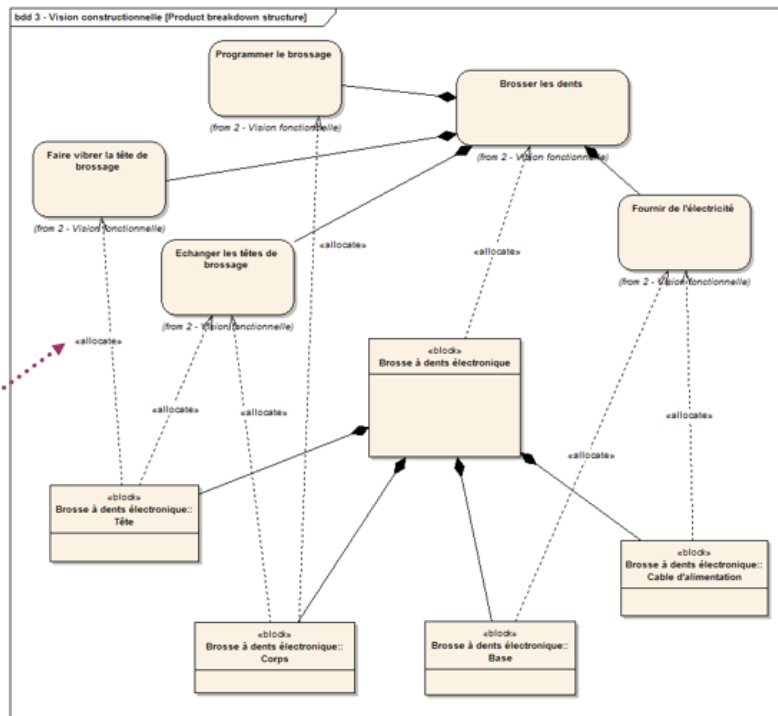


# Fondements de SysML – Vues

## TRAÇABILITÉ

### Vue de traçabilité (pas un diagramme stricto sensu)

Modélise les relations de traçabilité entre missions, fonctions et composants



**Allocation**  
Modélise une relation de traçabilité

# Table des matières

## 1 Fondements de SysML

*sous-thème* Introduction à l'approche système

*sous-thème* Historique & objectifs de SysML

*sous-thème* Principaux objets & vues SysML

*sous-thème* Synthèse : pourquoi utiliser SysML ?

# Fondements de SysML – Synthèse

## POURQUOI UTILISER SYSML ?

- **SysML** est le **seul langage réellement orienté-système** existant sur le marché :
  - La notion de **système** – considéré comme un mécanisme d'échanges de flots de grandeurs physiques ou informationnelles – est notamment au coeur de SysML où il est rendu par le concept de **“bloc”** (“block” en anglais)
  - Les concepts et les objets SysML permettent de rendre compte de la quasi totalité des notions clefs de l'**architecture système**
- **SysML** propose un **environnement unifié** qui implémente les fondamentaux du processus d'ingénierie système :
  - La **gestion des exigences** et de leurs relations avec les spécifications systémiques
  - La **traçabilité** des exigences et des différentes vues architecturales d'un système
- **SysML** est raisonnablement **intuitif** :
  - Il permet donc le **partage** entre architectes (qui doivent être des “écrivains” SysML) et non architectes (qui doivent devenir des “lecteurs” SysML)
- **SysML** a de bonnes chances de devenir un **standard système incontournable** :
  - De part le choix de se positionner dans le **monde UML** qui permet d'hériter de la force d'un standard industriel existant de tout premier poids

# Table des matières

- Introduction générale
- 1** Fondements de SysML
- 2** Structure d'un modèle système en SysML
- 3** Exercice de modélisation SysML
- 4** Ingénierie de la modélisation SysML
- Conclusion

# Table des matières

## **2** Structure d'un modèle système en SysML

*sous-thème* **Eléments d'architecture système**

*sous-thème* **Organisation d'un modèle SysML**

*sous-thème* **Bonnes et mauvaises pratiques**



# Modèle système – Visions

## LES TROIS GRANDES VISIONS SYSTÈMES



Brosse à dent électronique

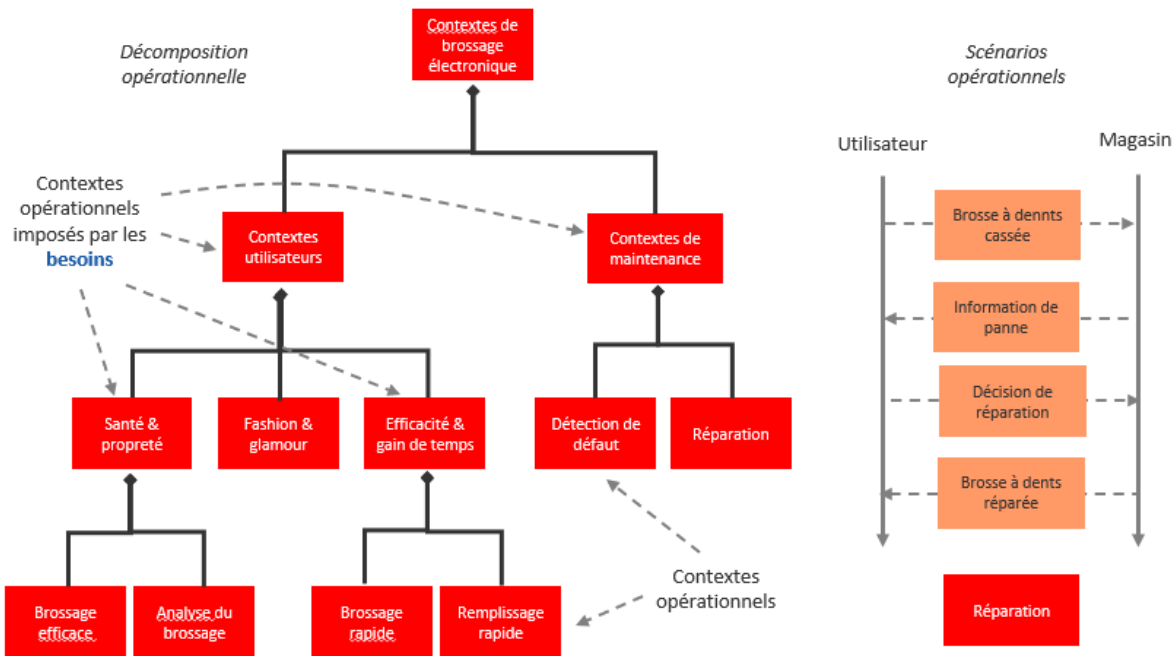
Points de vue	Répond à la question	Quelques mots clefs associés	Exemples (brosse à dent électronique)
<i>Opérationnelle</i>	Pourquoi ?	Contexte opérationnel, mission, cas d'utilisation	Dents propres et saines, gain de temps, salle de bain "fashion"
<i>Fonctionnelle</i>	Quoi ?	Fonction, tâche, opération & mode opératoire	Brossage, régulation de vitesse, programmation de la force de brossage
<i>Organique</i>	Comment ?	Composant, architecture & configuration technique	Tête, base, corps, régulateur de vitesse

# Modèle système – Visions

## VISION OPÉRATIONNELLE – POURQUOI ?



Brosse à dent électronique



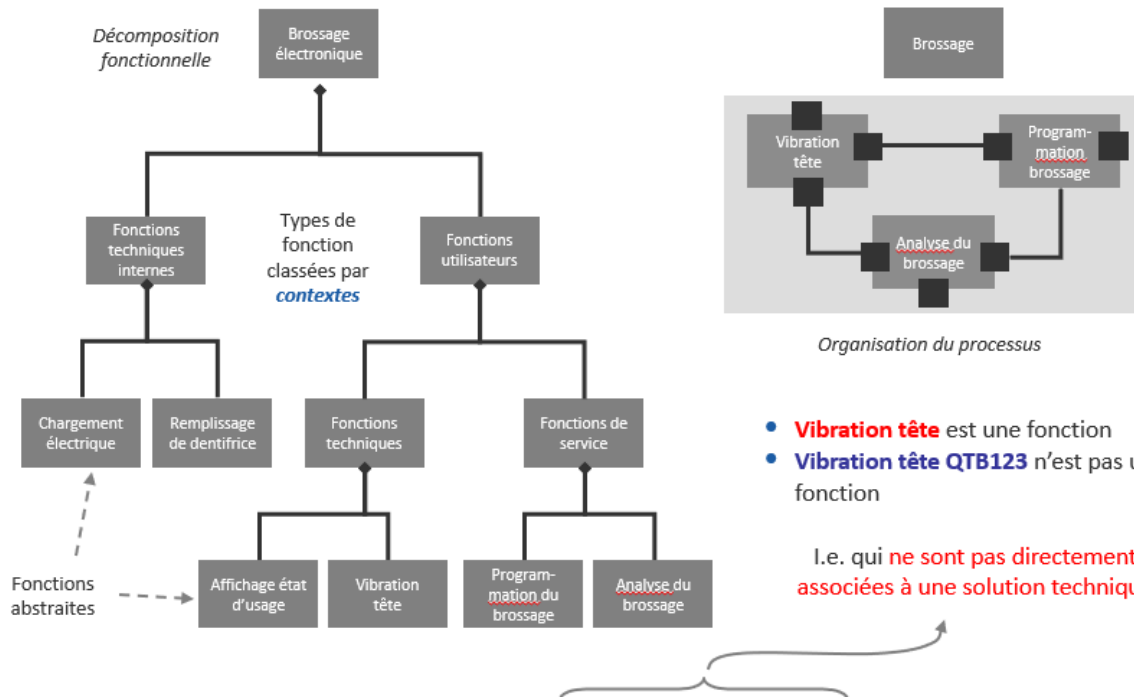
Vision opérationnelle : elle définit les **objectifs** & les **usages attendus** ainsi que les **processus opératoires** du système du point de vue des **systèmes externes** (clients, utilisateurs finaux, etc.)

# Modèle système – Visions

## VISION FONCTIONNELLE – QUOI ?



Brosse à dent électronique



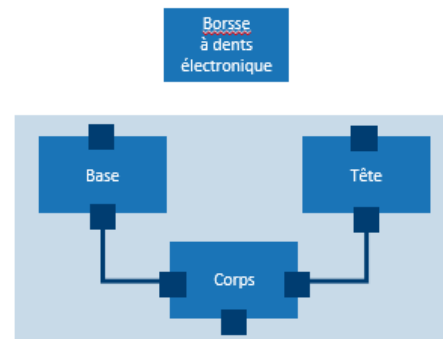
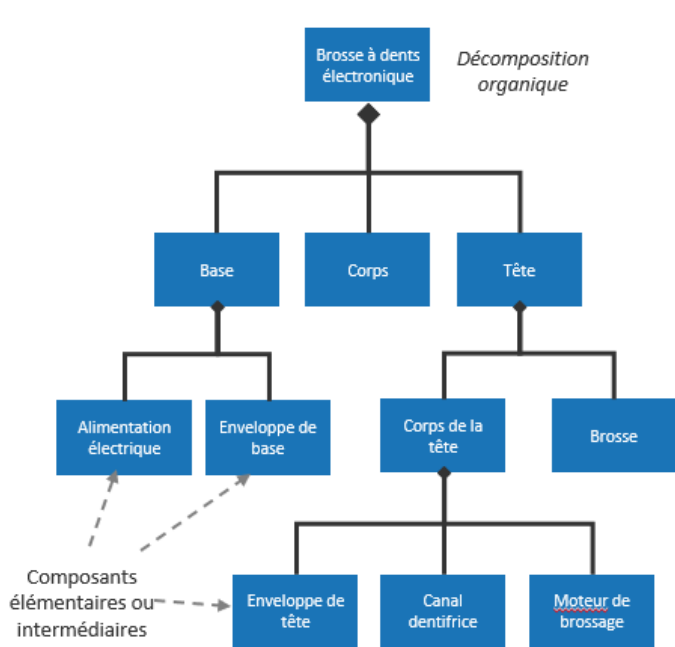
Vision fonctionnelle : elle définit les **fonctions abstraites** qui sont nécessaires pour réaliser les **missions** du système

# Modèle système – Visions

## VISION ORGANIQUE – COMMENT ?



Brosse à dent électronique



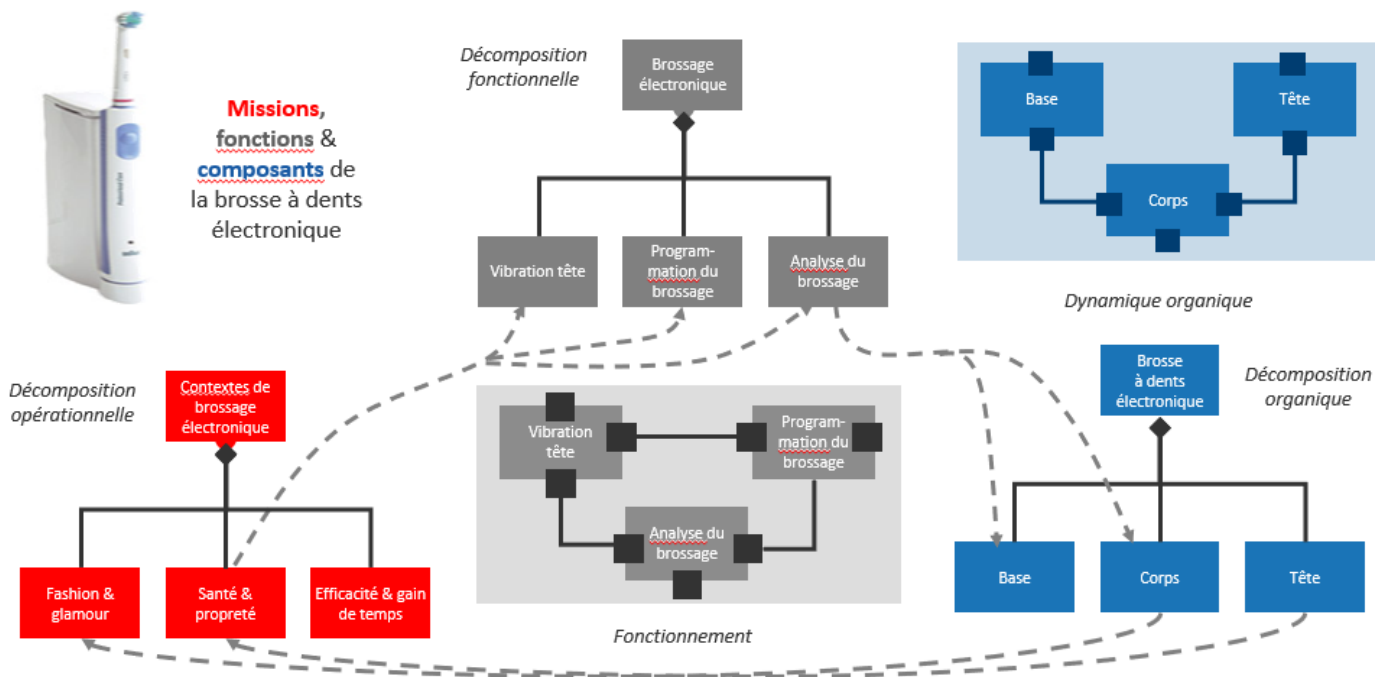
Organisation matérielle

- Composants **matériels**
- Composants **logiciels**
- **Hommes**

Vision organique : elle définit les **composants réels** qui vont implémenter les **fonctions abstraites**

# Modèle système – Visions

## CONSISTENCE D'UN MODÈLE



Consistance des points de vue : **l'organisation cohérente** des missions, fonctions & composants par rapport aux besoins

# Table des matières

## 2 Structure d'un modèle système en SysML

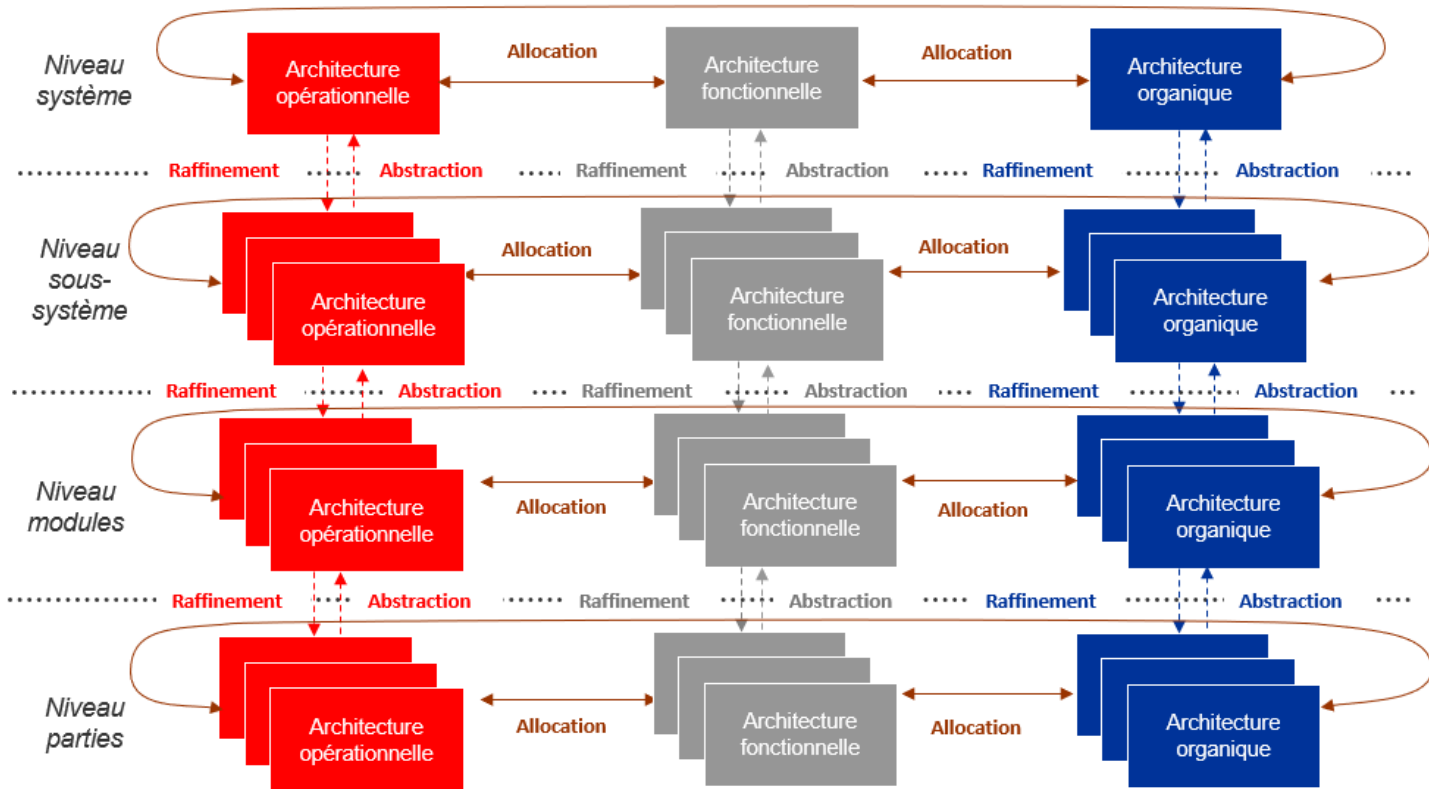
*sous-thème* Éléments d'architecture système

*sous-thème* Organisation d'un modèle SysML

*sous-thème* Bonnes et mauvaises pratiques

# Modèle système – Organisation

## VISION HIÉRARCHIQUE D'ENSEMBLE



# Modèle système – Organisation

## STRUCTURE D'UN MODÈLE SYSML

Visions architecturales	Exigences	→ Etats	→ Structure statique	→ Comportement dynamique	→ Données métiers
<i>Vision opérationnelle</i>	Exigences opérationnelles (Diagramme d'exigence)	Synthèse des contextes opérationnels (Machine à états)	Contextes opérationnels (Use cases)	Scénarios opérationnels (Diagramme de séquence pour chaque use case)	Données opérationnelles (Diagramme de bloc)
↓ <i>Vision fonctionnelle</i>	Exigences fonctionnelles (Diagramme d'exigence)	Synthèse des modes de fonctionnement (Machine à états)	Décomposition fonctionnelle & interactions fonctionnelles (Diagramme d'activité)	Fonctionnements (Diagramme de séquence associé à chaque fonction)	Données fonctionnelles (Diagramme de bloc)
↓ <i>Vision organique</i>	Exigences organiques (Diagramme d'exigence)	Synthèse des configurations (Machine à états)	Décomposition organique & interactions organiques (Diagramme de bloc et de bloc interne)	Fonctionnements organiques (Diagramme de séquence associé à chaque organe)	Données organiques (Diagramme de bloc)

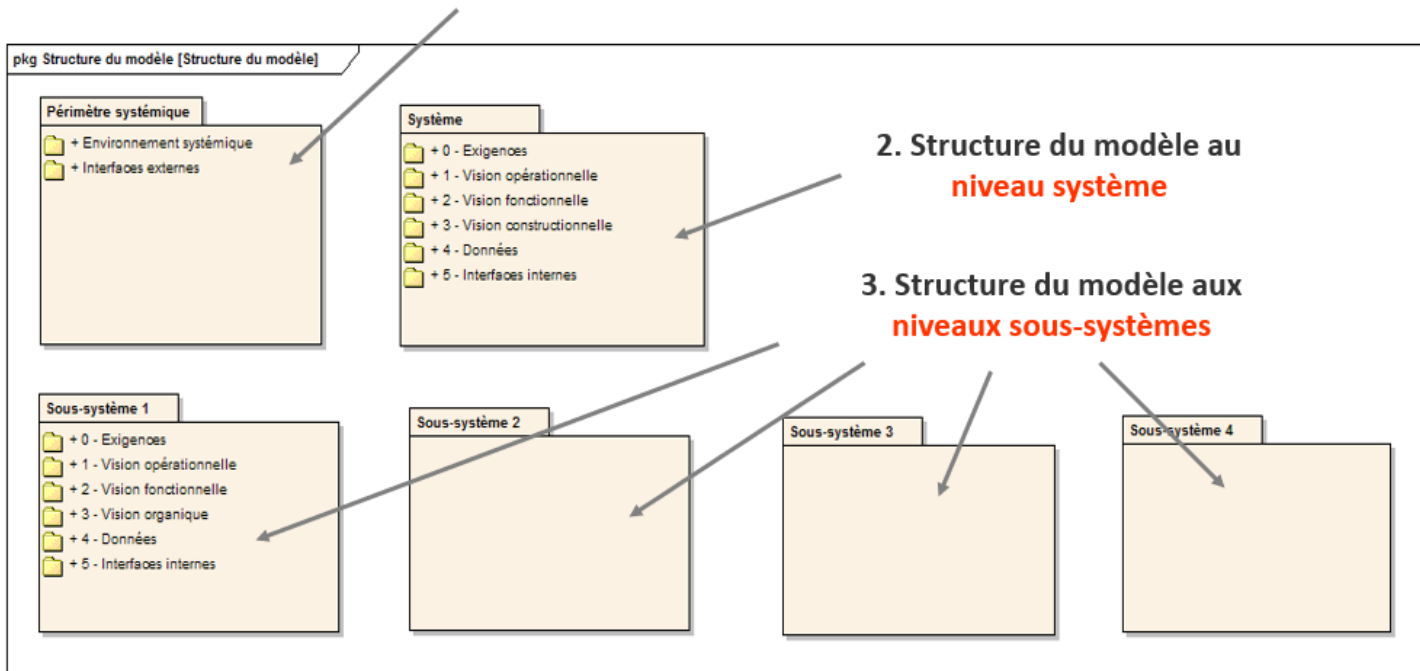
Structure d'un modèle SysML à un **niveau systémique donné**



# Modèle système – Organisation

## ORGANISATION D'UN MODÈLE SYSML

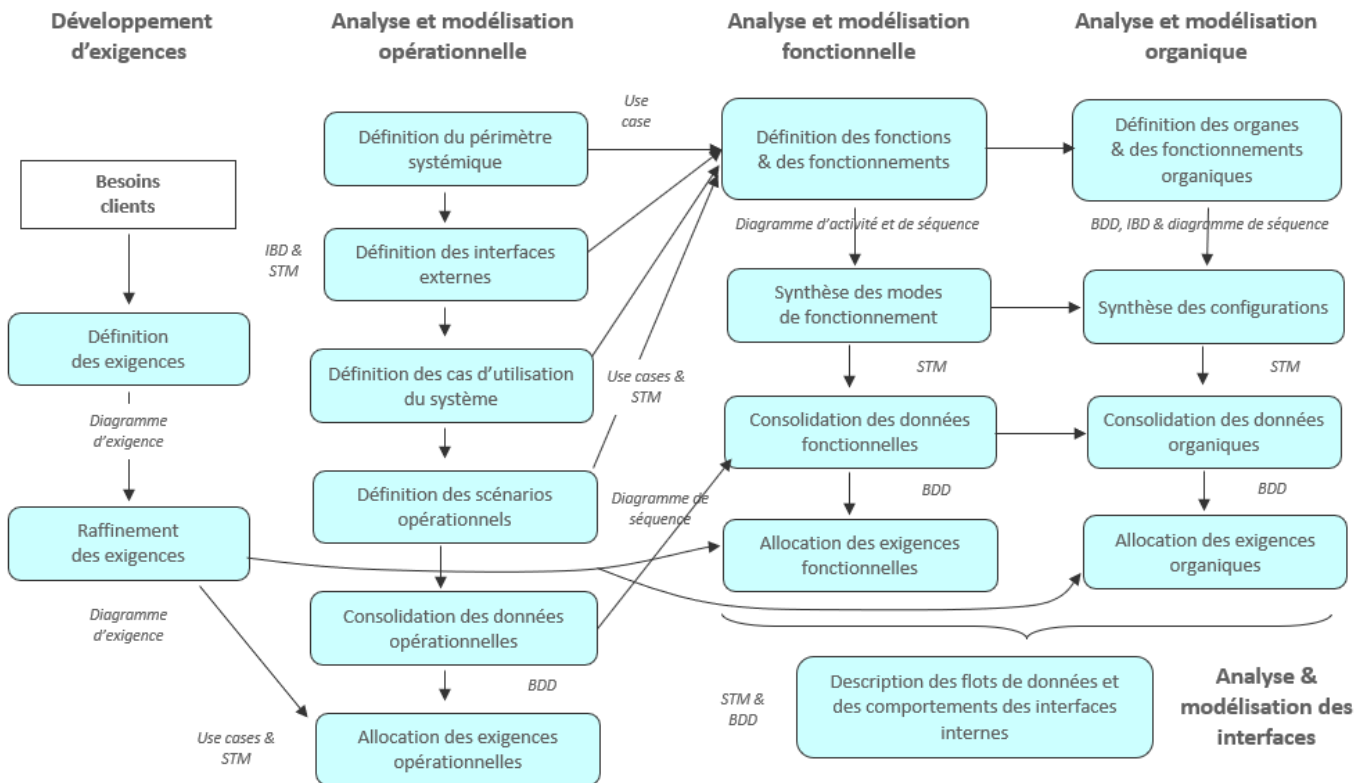
### 1. Description de l'environnement du système



L'accostage d'un niveau systémique à un sous-niveau systémique nécessite de spécifier l'interface correspondante (qui se décrit comme un système)

# Modèle système – Organisation

## LE PROCESSUS DE MODÉLISATION SysML



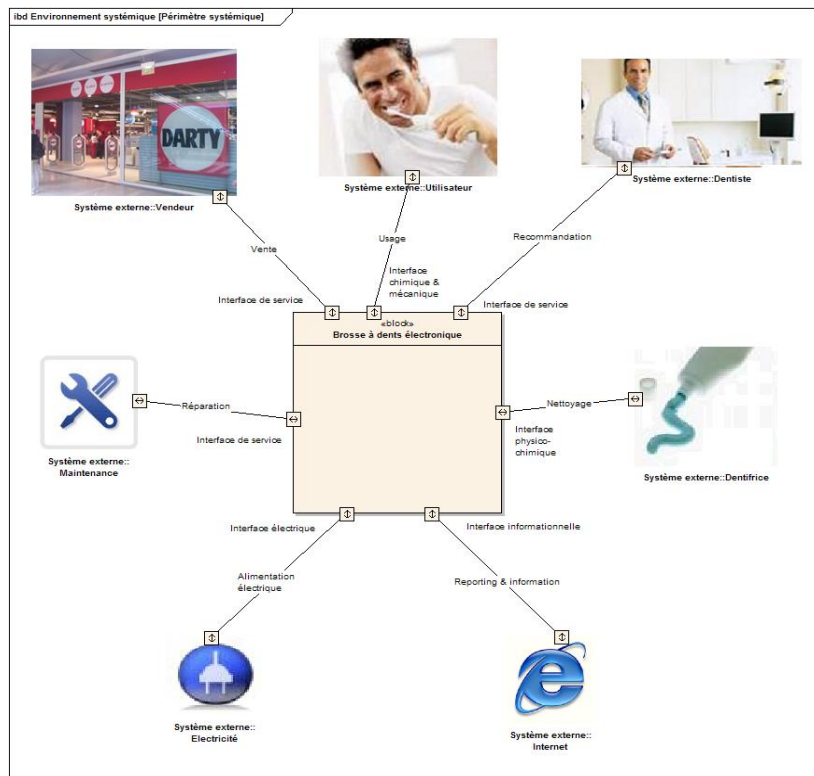
# Modèle système – Organisation

## EXEMPLE DE MODÈLE SYSML (1)



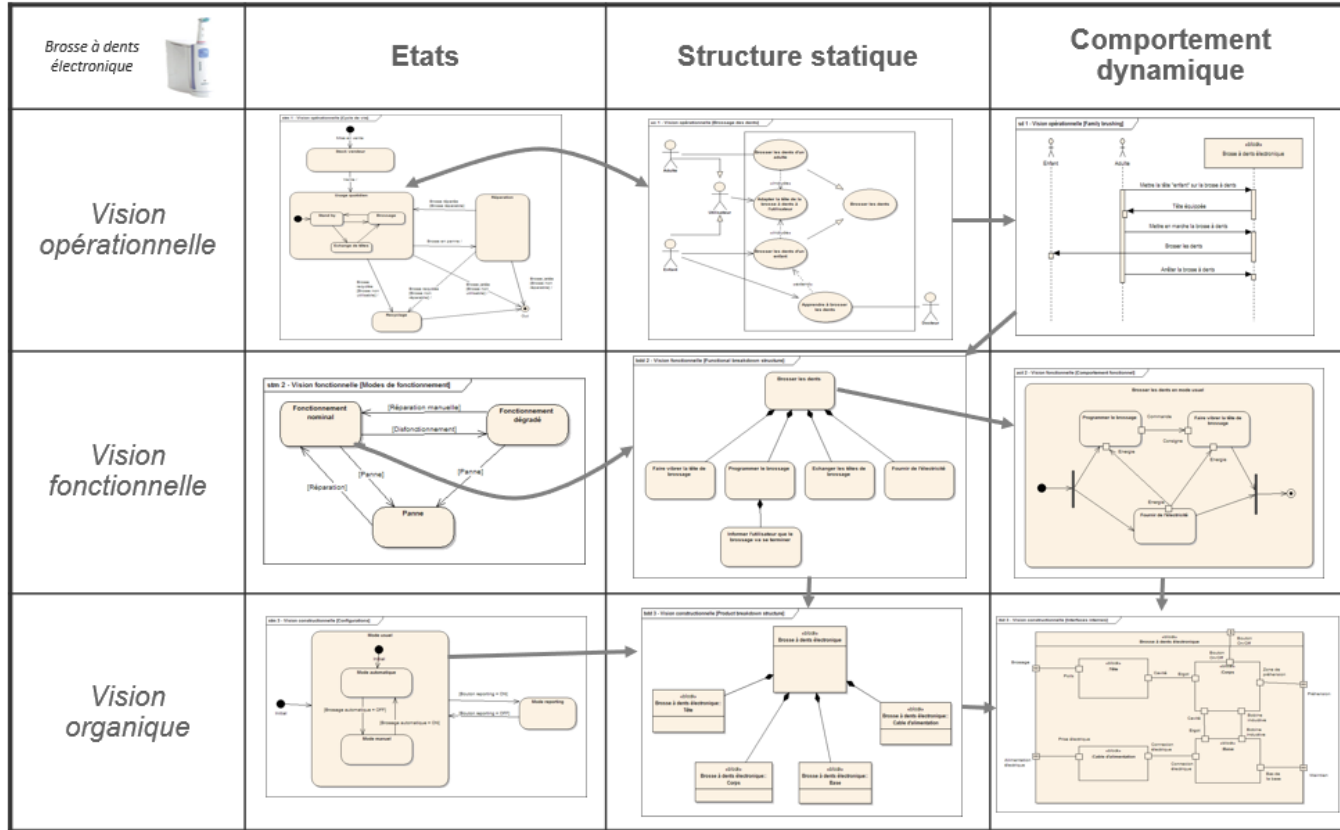
Brosse à dent  
électronique

Le point de départ : l'analyse  
et la définition du **périmètre**  
**systémique** du système étudié



# Modèle système – Organisation

## EXEMPLE DE MODÈLE SYSML (2)



# Table des matières

## **2** Structure d'un modèle système en SysML

*sous-thème* Éléments d'architecture système

*sous-thème* Organisation d'un modèle SysML

*sous-thème* Bonnes et mauvaises pratiques

# Modèle système – Pratiques

## LES BONNES PRATIQUES SysML

- Un **modèle SysML** doit toujours tendre à être à la fois :
  - **complet** : tout composant opérationnel (resp. fonctionnel ou organique) du modèle doit être lié à un composant fonctionnel et/ou organique (resp. opérationnel et/ou organique ou opérationnel et/ou fonctionnel)
  - **cohérent** : le modèle ne doit contenir aucune contradiction logique interne, chaque composant du modèle devant notamment être totalement indépendant du chemin de dérivation que l'on suit pour y accéder
- Un **modèle SysML** doit toujours être organisé de manière à assurer un **“isomorphisme”** et une **subsidiarité** entre **l'architecture des exigences et celle du modèle** :
  - Chaque exigence ne peut notamment couvrir qu'un élément et un seul du modèle
  - Un élément du modèle peut être couvert par plusieurs exigences, mais qui expriment uniquement des propriétés de **sécurité** (le système vérifie toujours une propriété), de **vivacité** (le système vérifiera toujours une propriété) ou d'**équité** (si le système vérifie une propriété, il existera un moment où le système ne vérifiera pas cette propriété)
- **SysML** peut outiller les **processus de vérification & de validation** :
  - Il est notamment recommandé d'associer systématiquement à chaque diagramme et à chaque exigence d'un modèle un **cas de test** que l'on peut décrire typiquement sous la forme d'un diagramme de séquence spécifiant le protocole de test à suivre

# Modèle système – Pratiques

## LES MAUVAISES PRATIQUES SYSML

- **Ne pas organiser** un modèle **SysML** débouche nécessairement sur le chaos !
  - **Structurez votre modèle** selon **la règle des 7 x 7 x 7** : pas plus de 7 packages de premier niveau, eux mêmes contenant récursivement au plus 7 packages de second niveau (avec un maximum de 3 niveaux au total)
- **Ne pas suffisamment abstraire** les éléments d'un modèle **SysML** conduit toujours à des architectures trop complexes et peu lisibles !
  - **Organisez chaque diagramme** selon la règle des 7 x 7 x 7 : pas plus de 7 éléments de premier niveau, eux mêmes contenant récursivement au plus 7 composants de second niveau (avec un maximum de 3 niveaux au total)
  - **Abstrayez en permanence** les éléments bruts issus de vos analyses pour éviter de rester “coincé” dans le niveau organique et pouvoir imaginer de nouvelles solutions
- **Utiliser** des **niveaux de grain non homogènes** au sein d'un modèle **SysML** est une erreur architecturale !
  - Veillez toujours à ce que chaque élément du modèle soit de **nature** et de **structure comparable à celle des autres éléments** du modèle du même niveau systémique
- **Construire** un modèle **SysML en “chambre”** empêche toute convergence des parties prenantes du système et fragilise donc votre travail de clarification !
  - Pensez toujours à **partager votre modèle** avec toutes les parties prenantes concernées au cours d'**ateliers de travail collaboratif**

# Table des matières

- Introduction générale
- 1** Fondements de SysML
- 2** Structure d'un modèle système en SysML
- 3** Exercice de modélisation SysML
- 4** Ingénierie de la modélisation SysML
- Conclusion

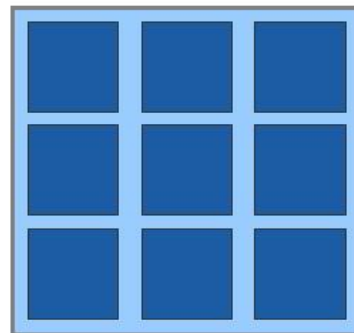


# Exercice de modélisation SysML

## CONSTRUCTION D'UN MODÈLE SYSML



Grille d'analyse systémique



Systeme

Construire une analyse systémique en SysML  
du **systeme de votre choix**

# Table des matières

- Introduction générale
- 1** Fondements de SysML
- 2** Structure d'un modèle système en SysML
- 3** Exercice de modélisation SysML
- 4** Ingénierie de la modélisation SysML
- Conclusion

# Table des matières

## 4 Ingénierie de la modélisation SysML

*sous-thème* Les modeleurs SysML

*sous-thème* Le processus d'ingénierie avec SysML

*sous-thème* Dangers & recommandations

# Modélisation SysML – Modeleurs

## FONCTIONNEMENT D'UN MODELEUR SYSMML

The screenshot illustrates the Enterprise Architect interface for SysML modeling. The main workspace shows a diagram titled "Boîte à outils générique" (Generic toolbox) containing several use case elements like "Brosser les dents d'un adulte" (Brushing an adult's teeth) and "Adapter la tête de la brosse à dents à l'utilisateur" (Adapting the toothbrush head to the user). A "Boîte à outils SysML" (SysML toolbox) is located on the left side of the interface. A "Use Case" dialog box is open, displaying the "Spécification d'un composant élémentaire du modèle" (Specification of an elementary component of the model) for the selected use case. On the right, the "Project Browser" window shows the "Définition de la structure du modèle" (Definition of the model structure) as a hierarchical tree of components and diagrams, including "Toothbrush model", "0 - Exigences", "1 - Vision opérationnelle", "2 - Vision fonctionnelle", "3 - Vision constructionnelle", "4 - Données", and "5 - Interfaces internes".

Principales fonctionnalités d'un modeleur SysML (exemple d'Enterprise Architect)

# Modélisation SysML – Modeleurs

## COMPARAISON DE MODELEURS SYSML

- **Artisan Studio** – Artisan Software
- **Entreprise Architect** – Sparx Systems
- **Rhapsody** – Telelogic
- **Objecteering** – Objecteering software
- **Rational Systems Developer** – IBM

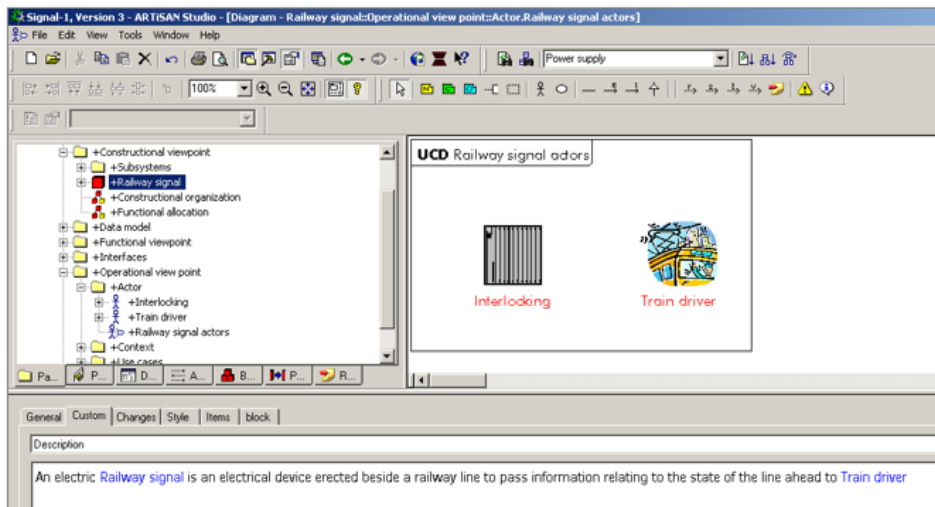
Modeleurs SysML  
les plus évolués à  
date

Critères  
d'évaluation et  
de comparaison

Critères généraux	Critères techniques
Solidité de l'éditeur	Respect des standards SysML
Qualité de l'offre de maintenance	Génération de code
Facilité d'installation	Travail collaboratif
Prix	Génération de documentation
Performance	Lien avec d'autres outils

# Modélisation SysML – Modeleurs

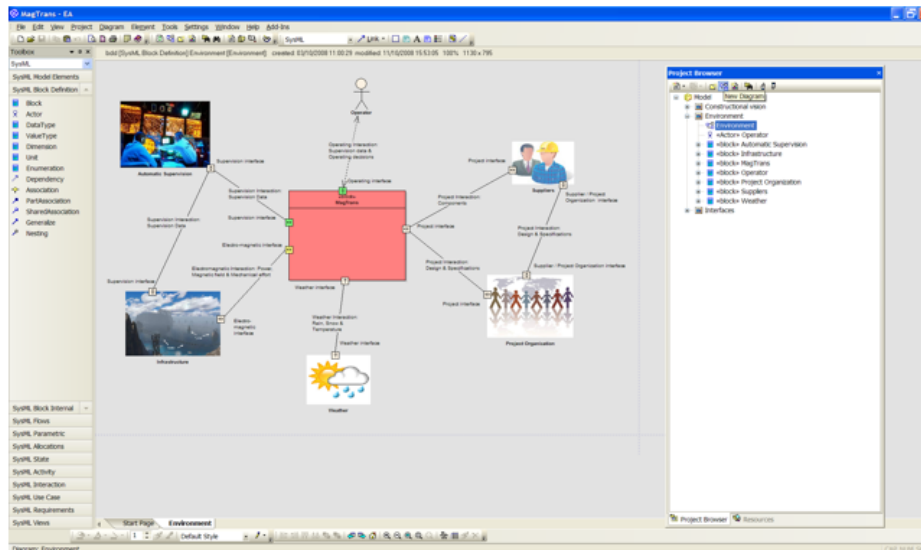
## ARTISAN STUDIO



Points forts	Points faibles	Points bloquants
<ul style="list-style-type: none"><li>• Outil industriel &amp; robuste</li><li>• Diffusion &amp; présence mondiale</li><li>• Interface avec Reqify &amp; DOORS</li><li>• Vérification automatique du modèle</li><li>• Maturité et respect des standards SysML</li></ul>	<ul style="list-style-type: none"><li>• Coût de maintenance élevé (obligation d'achat de 3 ans au moins de maintenance, coût de maintenance égal à 30 % du coût d'acquisition / an)</li><li>• Petite société (30 personnes)</li></ul>	<p>Aucun</p>

# Modélisation SysML – Modeleurs

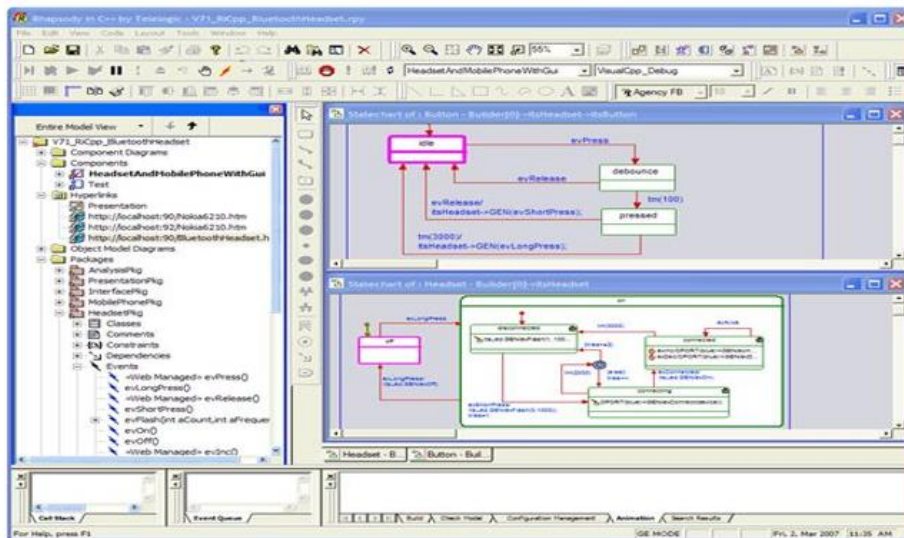
## ENTERPRISE ARCHITECT



Points forts	Points faibles	Points bloquants
<ul style="list-style-type: none"><li>• <b>Prix très bas</b></li><li>• <b>Diffusion élevée au niveau mondial</b></li></ul>	<ul style="list-style-type: none"><li>• <b>Aucun mécanisme de vérification automatique du modèle</b></li><li>• <b>Ergonomie d'utilisation faible</b></li><li>• <b>Pas d'interface avec d'autres outils de modélisation système</b></li></ul>	<p><b>Pas de respect des standards SysML</b></p>

# Modélisation SysML – Modeleurs

## RHAPSODY

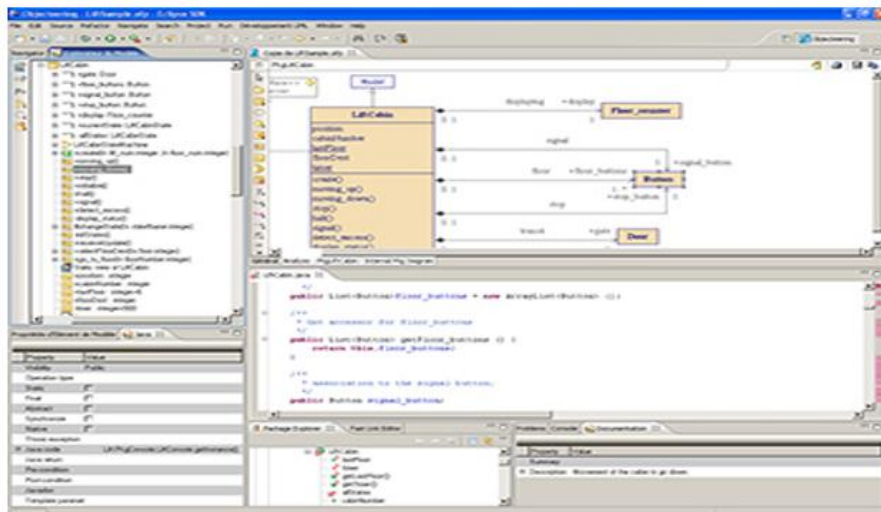


Points forts	Points faibles	Points bloquants
<ul style="list-style-type: none"><li>• Outil industriel</li><li>• Part de marché très élevée</li><li>• Interface avec SCADA &amp; Simulink</li><li>• Interface avec Reqify &amp; DOORS</li></ul>	<ul style="list-style-type: none"><li>• Manque de maturité SysML</li><li>• Avenir de l'éditeur depuis son rachat par IBM</li></ul>	Faible respect des standards SysML



# Modélisation SysML – Modeleurs

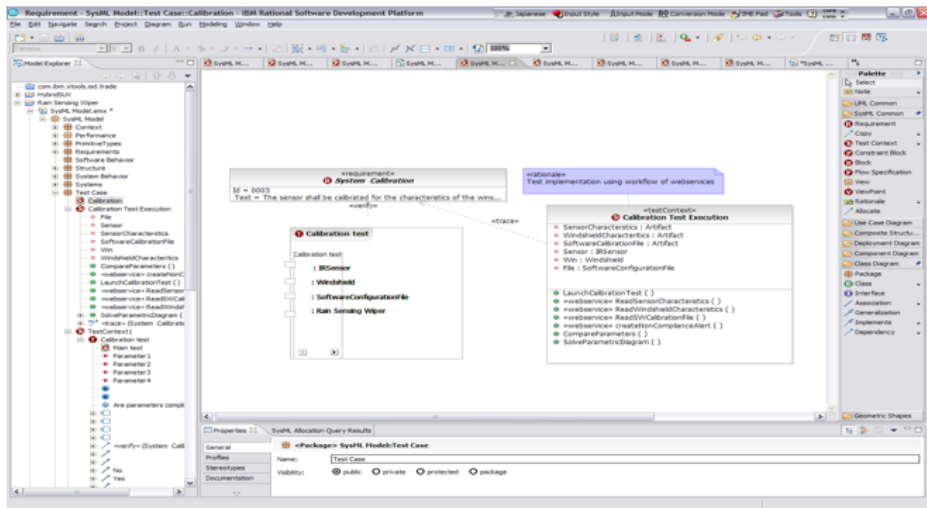
## OBJECTEERING



Points forts	Points faibles	Points bloquants
<ul style="list-style-type: none"><li>• <b>Outil industriel</b></li><li>• <b>Ergonomie d'utilisation forte</b></li><li>• <b>Analyse automatique en temps réel de la consistance du modèle</b></li><li>• <b>Interface avec DOORS</b></li></ul>	<ul style="list-style-type: none"><li>• <b>Faible viabilité de l'éditeur</b></li><li>• <b>Faible maturité SysML</b></li><li>• <b>Interface avec Reqify annoncée, mais pas encore existante</b></li></ul>	<p><b>Faible respect des standards SysML</b></p>

# Modélisation SysML – Modeleurs

## RATIONAL SYSTEMS DEVELOPER



Points forts	Points faibles	Points bloquants
<ul style="list-style-type: none"><li>• Force de l'éditeur (IBM)</li><li>• Intégration avec Eclipse</li><li>• Migration sans coût à RSD si l'on possède une licence Rose</li><li>• Intégration avec Doors</li></ul>	<ul style="list-style-type: none"><li>• Stratégie non évidente de l'éditeur depuis l'acquisition de Rhapsody</li><li>• Complexité dû au fait que 3 vendeurs (IBM, Eclipse, Embedded plus) sont impliqués dans l'achat</li><li>• Interface avec Reqify inexistante</li></ul>	<p>Aucun</p>

# Modélisation SysML – Modeleurs

## COMPARAISON DE MODELEURS SYSML

	Critères	Poids	Artisan Studio		Entreprise Architect		Objectteering		Rhapsody		IBM RSD	
			Note	Valeur	Note	Valeur	Note	Valeur	Note	Valeur	Note	Valeur
	<b>Critères génériques</b>											
1	Solidité de l'éditeur	10		45		40		35		40		40
2	Qualité de la maintenance	14		59		58		53		63		61
3	Facilité d'installation	14		64		53		70		60		60
4	Prix	10		30		45		25		45		35
5	Performances	4		20		20		20		20		12
	<b>Critères techniques liés à l'outil</b>					<b>Critères bloquants</b>						
6	Respect des standards SysML	36		160		121		153		136		135
7	Génération de code	4		13		9		9		12		11
8	Travail collaboratif	15		75		56		75		73		73
9	Gestion de documentation	16		56		65		49		67		53
10	Lien avec d'autres outils	10		43		27		24		44		43
	<b>Note finale</b>			564		494		513		560		523



**Artisan Studio** semble être le meilleur choix à date en matière de modeleur SysML

# Table des matières

## 4 Ingénierie de la modélisation SysML

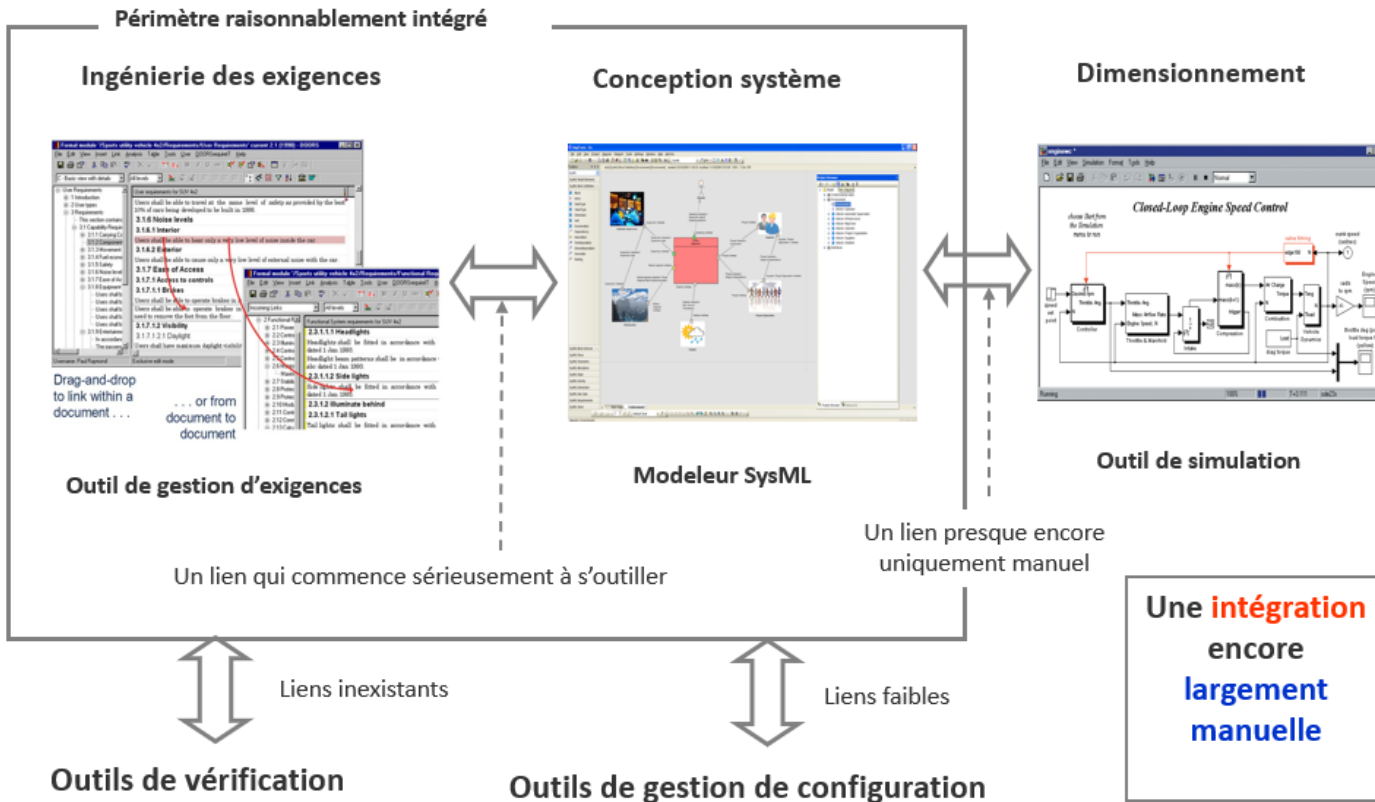
*sous-thème* Les modeleurs SysML

*sous-thème* Le processus d'ingénierie avec SysML

*sous-thème* Dangers & recommandations

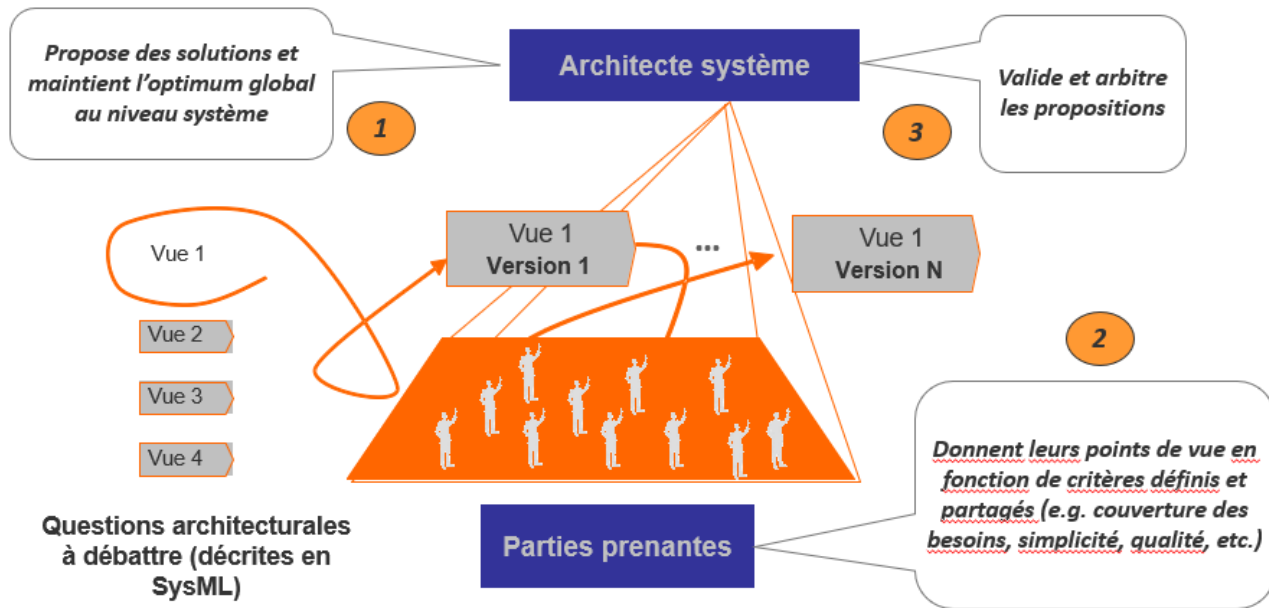
# Modélisation SysML – Ingénierie

## LE PÉRIMÈTRE DE L'ARCHITECTURE SYSTÈME



# Modélisation SysML – Ingénierie

## UN SUPPORT À LA CONVERGENCE



SysML est un langage commun de description système qui permet de **partager facilement des visions systèmes** entre leurs parties prenantes au sein **d'ateliers d'architecture collaborative** qui servent à converger sur un modèle en un même lieu et instant

# Table des matières

## 4 Ingénierie de la modélisation SysML

*sous-thème* Les modeleurs SysML

*sous-thème* Le processus d'ingénierie avec SysML

*sous-thème* Dangers & recommandations

# Modélisation SysML – Dangers

## LES CHAUSSE-TRAPPES CLASSIQUES

- Utiliser **SysML** ne peut se faire sans avoir **défini au préalable** :
  - la structure d'un **processus de modélisation** adapté à ses besoins et à ses pratiques
  - des règles communes de **travail collaboratif & de partage** d'un modèle
- Utiliser **SysML** impacte l'ensemble de **l'ingénierie** bien au delà de la conception système proprement dite et tout particulièrement :
  - le processus d'**ingénierie des exigences**
  - le processus de **dimensionnement** d'un système
  - le processus de **vérification, validation & qualification**

Toutes les **parties prenantes** de l'ingénierie d'un système technique doivent donc être au moins capable de lire des spécifications SysML (à défaut de savoir les écrire).

- Utiliser **SysML** reste **difficile** car la méthodologie sous-entend que ses utilisateurs aient :
  - la maîtrise des **méthodes de l'architecture système**
  - de fortes capacités d'**abstraction** (au sens de l'interprétation abstraite)
- Mettre en place **SysML** ne revient pas simplement à donner un modéleur aux ingénieurs, mais nécessite une réelle prise en compte de la **conduite du changement** sous-jacente
  - Aider les **architectes systèmes** à s'approprier la démarche
  - Accompagner les **utilisateurs de SysML non architectes** (clients, qualité, etc.)



# Table des matières

- Introduction générale
- 1** Fondements de SysML
- 2** Structure d'un modèle système en SysML
- 3** Exercice de modélisation SysML
- 4** Ingénierie de la modélisation SysML
- Conclusion

# Conclusion

## LES MESSAGES CLEFS À RETENIR

Thèmes	Messages clefs
<i>Fondements de SysML</i>	<b>SysML</b> est le langage par excellence de <b>l'architecture système</b> dont il permet d'instancier tous les concepts clefs. Les <b>objets</b> et les <b>diagrammes SysML</b> permettent notamment de modéliser les <b>systèmes</b> en décrivant leurs <b>interfaces</b> , leurs <b>structures</b> , leurs <b>comportements</b> et leurs <b>exigences</b> tout en garantissant la <b>traçabilité</b> entre ces différentes caractéristiques d'un système.
<i>Structure d'un modèle système en SysML</i>	Un <b>modèle SysML</b> doit se structurer selon un cadre d'architecture système autour de deux axes clefs d'organisation, à savoir d'une part les <b>visions opérationnelles, fonctionnelles et organiques</b> et d'autre part les <b>niveaux systémiques</b> (niveau système, niveau sous-système, etc.) d'un système. Le processus de <b>modélisation SysML</b> consiste à modéliser un système par rapport à ces différents axes d'analyse.
<i>Ingénierie de la modélisation SysML</i>	<b>SysML</b> est une méthodologie d'architecture système qui <b>commence à être outillée</b> , même si les outils de modélisation SysML sont encore <b>partiellement déconnectés</b> des autres outils d'ingénierie système. Mettre en place <b>SysML</b> au sein d'un processus d'ingénierie <b>reste cependant difficile</b> et nécessite une réelle démarche de <b>conduite du changement</b> technique, organisationnel et humain qui en résulte.

# Conclusion

## BIBLIOGRAPHIE RAISONNÉE

### • Ressources bibliographiques

- Systems Engineering with SysML/UML, *Tim Weilkiens*, Morgan Kaufmann OMG Press, 2007 (version anglaise du premier ouvrage sur SysML paru initialement en allemand)
  - Commentaire : ouvrage assez facile d'accès et très pédagogique, même s'il ne couvre pas toutes les dimensions de la norme SysML – un bon point d'entrée pour ceux qui veulent approfondir leurs connaissances après avoir commencé à utiliser SysML
- A practical Guide to SysML, The SysML Modeling Language, *Sanford Friedenthal, Alan Moore, Rick Steiner*, Morgan Kaufmann OMG Press, 2008
  - Commentaire : ouvrage très complet, dont l'un des auteurs est un des grands experts mondiaux de SysML, mais pas toujours très pédagogique – il n'est pas recommandé pour une première approche de SysML, mais sera clairement très utile à l'expert SysML à qui il pourra servir de livre de références

### • Ressources numériques disponibles sur Internet

- <http://www.omgsysml.org/>
  - Commentaire : le site Web de l'OMG dédié à SysML – il contient une base d'informations à jour (notamment en matière de standard) et un portail d'accès au monde SysML
- [http://www.omgsysml.org/SysML-Tutorial-Baseline-to-INC05E-060524-low\\_res.pdf](http://www.omgsysml.org/SysML-Tutorial-Baseline-to-INC05E-060524-low_res.pdf)
  - Commentaire : le premier tutorial sur SysML, présenté lors de la conférence mondiale de l'INC05E de Juillet 2006, reste une excellente référence et un incontournable du petit monde SysML – à lire à plusieurs reprises tant il y a d'informations dans ce document

# Conclusion

## VOTRE OPINION SUR LES OBJECTIFS INITIAUX

Objectifs initiaux	+	=	-
Connaître les fondements de <b>SysML</b>			
Comprendre comment <b>organiser &amp; structurer un modèle système</b> en SysML			
Etudier un <b>exemple concret</b> de modélisation <b>SysML (-like)</b> sur un cas d'étude tiré de votre pratique métier			
Mieux comprendre la manière dont on peut utiliser SysML dans le cadre d'un <b>processus de modélisation système</b>			

Merci de votre vote ...

# Contact

CESAMES

Adresse : 15, rue La Fayette – 75009 Paris – France

Téléphone : +33 (0)1 40 15 54 20

