PRÉPAS SCIENTIFIQUES

Alain Caignot • Vincent Crespel • Marc Derumaux • Christian Garreau
Patrick Kaszynski • Baudoin Martin • Sébastien Roux • Jean Saint-Pierre

Sciences industrielles de l'ingénieur

PTSI

Tout-en-un

Tout le cours avec :
- Objectifs-clés
- Démonstrations
- Conseils méthodologiques

Fiches de synthèse

Entraînement intensif :
- Vrai/faux
- Exercices d'application
- Exercices d'approfondissement
- Problèmes types concours

Tous les corrigés détaillés

Vuibert

+ EN LIGNE

+ de TP à télécharger

Sciences industrielles de l'ingénieur

PTSI

Tout-en-un

Alain Caignot est professeur en classe préparatoire scientifique au collège Stanislas à Paris.
Vincent Crespel est professeur en classe préparatoire scientifique au lycée Saint-Louis à Paris.
Marc Dérumaux est professeur en classe préparatoire scientifique au lycée Saint-Louis à Paris.
Christian Garreau est professeur en classe préparatoire scientifique au lycée Déodat-de-Séverac à Toulouse.
Patrick Kaszynski est professeur en classe préparatoire scientifique au lycée Marcelin-Berthelot à Saint-Maur-des-Fossés.
Baudouin Martin est professeur en classe préparatoire scientifique au lycée Descartes à Tours.
Sébastien Roux est professeur en classe préparatoire scientifique au lycée Baggio à Lille.
Jean Saint-Pierre est professeur en classe préparatoire au lycée Livet à Nantes.

Vuibert

Ressources numériques

+ de TP complémentaires à télécharger ici :

Ces activités sont également téléchargeables sur le site www.vuibert.fr à la page du livre.

ISBN : 978-2-311-40631-3

Conception de couverture : Hung Ho Thanh
Conception LaTeX et mise en page : Sébastien Mengin

La loi du 11 mars 1957 n'autorisant aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite » (alinéa 1er de l'article 40). Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal. Des photocopies payantes peuvent être réalisées avec l'accord de l'éditeur. S'adresser au Centre français d'exploitation du droit de copie : 20 rue des Grands Augustins, F-75006 Paris.

Tél. : 01 44 07 47 70

© Vuibert – juillet 2019 – 5 allée de la 2e DB, 75015 Paris

SOMMAIRE

Partie 1. Le langage SysML pour l'Ingénierie Système 9

Chapitre 1 Introduction aux concepts de l'Ingénierie Système (IS) 11

Cours 12
1. Introduction 12
2. Les Sciences Industrielles de l'Ingénieur dans la formation scientifique en CPGE 12
3. Le contexte de travail dans l'entreprise 13
4. L'Ingénierie Système (IS) 19

Chapitre 2 La modélisation des systèmes complexes par le langage SysML 25

Cours 26
1. Objectifs du langage SysML 26
2. Les diagrammes et leurs applications 28
3. Exemple de développement d'un modèle 32

Chapitre 3 Les outils de description d'un système automatisé 49

Cours 50
1. Introduction 50
2. Système automatisé 50
3. Outils de description d'un système automatisé 50

Fiche de synthèse 55
Exercices 57
Aide à la résolution des exercices 68
Corrigés 68

Partie 2. Analyse des systèmes asservis 75

Chapitre 4 Modélisation des systèmes asservis 77

Cours 78
1. Introduction à la commande des systèmes 78

2. Modélisation du comportement dynamique d'un système 87
3. Validation des performances du système 97
Fiche de synthèse 103
Exercices 105
Aide à la résolution des exercices 118
Corrigés 118

Chapitre 5 **Analyse temporelle des systèmes** **129**

Cours 130
1. Intérêt de l'analyse temporelle et démarche 130
2. Système à action proportionnelle 131
3. Système intégrateur 132
4. Système du premier ordre 132
5. Système du deuxième ordre 136
Fiche de synthèse 144
Exercices 146
Aide à la résolution des exercices 179
Corrigés 181

Chapitre 6 **Analyse fréquentielle des systèmes** **199**

Cours 200
1. Représentation fréquentielle des signaux 200
2. Représentation graphique du comportement fréquentiel des systèmes : diagrammes de Bode203
3. Exploitation des diagrammes en ingénierie 211
Fiche de synthèse 217
Exercices 219
Aide à la résolution des exercices 239
Corrigés 241

Chapitre 7 **Annexe technique : Éléments de technologie des systèmes mécaniques asservis** **255**

Annexe 256
1. Chaîne d'action 256
2. Chaîne de retour 259
3. Partie commande 262

Chapitre 8 **Annexe mathématique : Transformée de Laplace et décomposition en éléments simples** **265**

Annexe 266
1. Transformée de Laplace 266
2. Décomposition en éléments simples 270

Partie 3. Cinématique des systèmes de solides indéformables 273

Chapitre 9 Introduction au cours de cinématique des systèmes de solides indéformables 275

Cours . . . 276
1. Introduction . . . 276
2. Présentation des exemples d'illustration du cours . . . 276
3. Première approche de la schématisation . . . 279

Chapitre 10 Paramétrage et définitions des grandeurs cinématiques 281

Cours . . . 282
1. Définition de la cinématique . . . 282
2. Notions élémentaires de cinématique . . . 282
3. Calcul de la dérivée d'un vecteur . . . 288
Fiche de synthèse . . . 294
Exercices . . . 296
Aide à la résolution des exercices . . . 307
Corrigés . . . 307

Chapitre 11 Cinématique des systèmes de solides indéformables 317

Cours . . . 318
1. Paramétrage d'un solide indéformable . . . 318
2. Mouvements d'un solide par rapport à un référentiel . . . 323
3. Composition des mouvements . . . 338
Fiche de synthèse . . . 342
Exercices . . . 344
Aide à la résolution des exercices . . . 370
Corrigés . . . 370

Chapitre 12 Modélisation cinématique des mécanismes 381

Cours . . . 382
1. Modélisation des liaisons entre les solides . . . 382
2. Modélisation cinématique d'un mécanisme . . . 394
3. Étude géométrique et cinématique des chaînes de solides . . . 399
4. Mécanisme cinématiquement plan – Modélisation plane . . . 405
Fiche de synthèse . . . 410
Exercices . . . 412
Aide à la résolution des exercices . . . 439
Corrigés . . . 441

Chapitre 13 Compléments mathématiques 457

Annexe . . . 458
1. Vecteurs et opérations vectorielles . . . 458
2. Dérivation vectorielle — Vecteur vitesse angulaire . . . 466

Partie 4. Systèmes logiques et numériques 469

Chapitre 14 Introduction aux systèmes numériques 471

Cours . . . 472
1. Conception d'un système de commande . . . 472
2. Manipulation de l'information en binaire . . . 478
Fiche de synthèse . . . 484
Exercices . . . 486
Aide à la résolution des exercices . . . 499
Corrigés . . . 499

Chapitre 15 Commande numérique sur base micro-contrôleur 509

Cours . . . 510
1. Fonctionnement d'un micro-contrôleur . . . 510
2. Programmation des micro-contrôleurs . . . 514
Fiche de synthèse . . . 517
Exercices . . . 519
Aide à la résolution des exercices . . . 525
Corrigés . . . 525

Partie 5. Modélisation des actions mécaniques et statique des solides 533

Chapitre 16 Modélisation des actions mécaniques 535

Cours . . . 536
1. Modélisation locale des actions mécaniques . . . 536
2. Modélisation globale des actions mécaniques sur les solides . . . 543
3. Modélisation plane des actions mécaniques . . . 551
Fiche de synthèse . . . 553
Exercices . . . 555
Aide à la résolution des exercices . . . 566
Corrigés . . . 566

Chapitre 17 Principe fondamental de la statique 575
Cours . . . 576
1. Équilibre des systèmes de solides . . . 576
2. Illustration : micro-compresseur . . . 579
3. Quelques isolements particuliers . . . 584
4. Méthodes de résolution . . . 587
Fiche de synthèse . . . 590
Exercices . . . 591
Aide à la résolution des exercices . . . 610
Corrigés . . . 610

Chapitre 18 Modélisation des liaisons réelles 625
Cours . . . 626
1. Illustration : guindeau Lewmar . . . 626
2. Influence de la taille relative de la zone de contact . . . 628
3. Influence du jeu sur les actions transmissibles . . . 631
4. Influence du frottement dans les liaisons . . . 634
5. Liaisons par éléments d'interface – Roulements . . . 635

Partie 6. Circuits électriques et applications 639

Chapitre 19 Sources et circuits électriques 641
Cours . . . 642
1. Modélisation des sources et des circuits électriques . . . 642
2. Grandeurs électriques dans un circuit . . . 646
Exercices . . . 650
Aide à la résolution des exercices . . . 654
Corrigés . . . 654

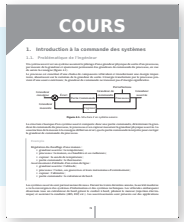
Chapitre 20 Modulation de l'énergie électrique 659
Cours . . . 660
1. Présentation . . . 660
2. Structures générales des convertisseurs statiques . . . 661
3. Les hacheurs . . . 670
4. Les onduleurs . . . 675
Exercices . . . 684
Aide à la résolution des exercices . . . 688
Corrigés . . . 688

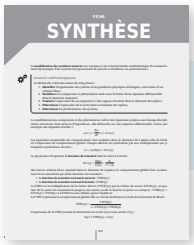
Mode d'emploi

Cet ouvrage, parfaitement conforme au programme de sciences industrielles de l'ingénieur en PTSI, vous propose les outils adaptés à la réussite de votre première année.

COURS

Tout le cours avec des rubriques claires pour un meilleur repérage des points importants à retenir, des conseils méthodologiques et de nombreux exemples et démonstrations





FICHES DE SYNTHÈSE

Des fiches de synthèse à la fin de chaque chapitre pour retenir l'essentiel

PICTOS DE REPÉRAGES

Pour un meilleur repérage des définitions

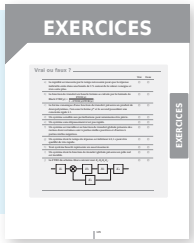
Conseils méthodologiques pour acquérir les bons réflexes et éviter les pièges

Attention ! pour mettre en valeur les points de vigilance

Objectif 2e année pour approfondir ses connaissances

ENTRAÎNEMENT

Un entraînement intensif avec quatre typologies d'exercices :
Vrai/faux, application, approfondissement et problèmes types concours.
Trois niveaux de difficulté clairement identifiés et un temps indicatif de réalisation de l'exercice ○○●, ○●● et ●●● → 25 min.



AIDE À LA RÉSOLUTION

Des aides à la résolution des exercices en cas de blocage à la lecture de l'énoncé

CORRIGÉS

Tous les corrigés détaillés pour comprendre chaque étape de résolution

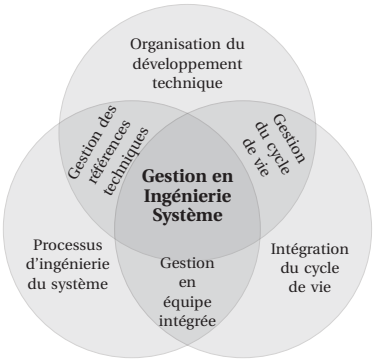
Partie 1

LE LANGAGE SYSML POUR L'INGÉNIERIE SYSTÈME

Chapitre 1

Introduction aux concepts de l'Ingénierie Système (IS)

1. Introduction — **2.** Les Sciences Industrielles de l'Ingénieur dans la formation scientifique en CPGE — **3.** Le contexte de travail en entreprise — **4.** L'Ingénierie Système (IS)



Problématique de l'ingénierie système

La conception d'un système industriel est une opération difficile et risquée pour l'entreprise : les évolutions continuelles des attentes des acheteurs et la concurrence accrue entre les différentes entreprises internationales expliquent en effet qu'une grande partie des projets soient abandonnés en cours de développement ou dépassent le budget initialement prévu.

Afin d'éviter d'identifier trop tardivement de telles issues, une organisation rigoureuse est indispensable : celle-ci doit être mise en œuvre depuis les prémices du projet jusqu'à la mise en vente finale du produit, et même prévoir son futur recyclage.

Cette démarche est relativement facile si les systèmes sont simples mais elle devient très difficile, et même souvent impossible, quand la complexité structurelle ou fonctionnelle du système augmente.

Une solution pour que le processus de création d'un objet technique aboutisse à une solution conforme aux évolutions des attentes du marché est d'adopter une démarche d'Ingénierie Système (*Systems Engineering* en anglais) dont quelques éléments sont présentés dans ce chapitre.

Cette démarche, en plein développement au niveau international, est très largement adoptée par les entreprises concernées par les évolutions technologiques et confrontées à une concurrence internationale accrue (aviation, automobile, téléphonie mobile, etc.).

COURS

1. Introduction

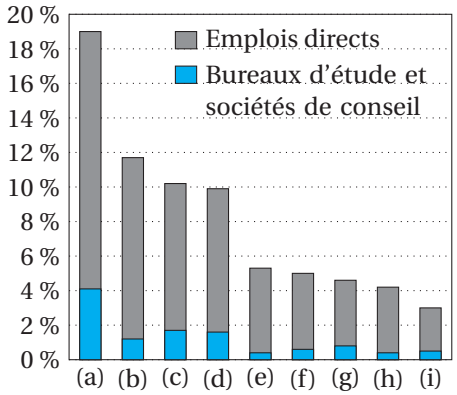
Quelques « points clés » sur l'ingénierie système, tant dans sa finalité que dans ses implications dans la formation en Sciences Industrielles de l'Ingénieur (SII), sont abordés dans ce chapitre.

Lors de la découverte de la discipline au début de la première année, il est possible, et même conseillé, de passer ce chapitre qui aborde des concepts très généraux difficiles à comprendre sans une certaine culture technique. Par contre, en deuxième année lorsque le recul sur la démarche et sur les méthodes est suffisant, il peut être tout à fait profitable de prendre le temps de lire et analyser ce chapitre.

En conclusion, *seule une lecture du paragraphe 2 ci-après est réellement utile en première approche* car il permet de présenter le positionnement de cet enseignement dans la formation du futur ingénieur ou chercheur en sciences et techniques.

2. Les Sciences Industrielles de l'Ingénieur dans la formation scientifique en CPGE

Le choix d'une formation de l'enseignement supérieur est une étape importante pour l'avenir d'un étudiant. La plupart des étudiants en CPGE (classes préparatoires aux grandes écoles) intégreront une école d'ingénieurs [1] puis une entreprise, ce qui leur permettra de travailler dans une grande variété de secteurs économiques, comme le montre la figure 1.1.



(a) Industries automobile, aéronautique, navale et ferroviaire
(b) Bâtiment, travaux publics et construction
(c) Énergies (industries liées au pétrole, gaz, nucléaire, etc.)
(d) Technologies de l'information (service)
(e) Industries chimique et pharmaceutique
(f) Autres secteurs industriels
(g) Institutions financières, banque et assurance
(h) Industrie agroalimentaire (transformation)
(i) Agriculture, sylviculture et pêche

Figure 1.1. Secteurs et taux d'emploi de la promotion diplômée par les écoles d'ingénieurs française, en 2012 (source : Conférence des grandes Écoles).

[1] Deux autres orientations sont également possibles : les Écoles normales supérieures (Ulm, Lyon, Paris Saclay et Rennes) et les filières universitaires (structure Licence, Master et Doctorat). Ces deux formations préparent plus spécifiquement à la recherche et à l'enseignement même si une part croissante de ces étudiants travaillent également en entreprise.

La formation dispensée dans les CPGE puis dans les Grandes écoles d'ingénieurs doit être vue comme un *continuum* de cinq années débouchant sur un diplôme qualifiant et non comme la succession de deux années de préparation suivies de trois années de spécialisation sans lien l'une avec l'autre. En conséquence, une initiation à la démarche et aux outils de l'ingénieur doit être proposée dès le début de la formation scientifique d'un futur ingénieur, chercheur ou enseignant : les Sciences Industrielles de l'Ingénieur (SII ou S2I) répondent à cet objectif en initiant les étudiants aux méthodes de raisonnement et aux pratiques utilisées en entreprise.

Afin de rester au plus près des activités de l'ingénieur, les études en Sciences Industrielles de l'Ingénieur seront systématiquement réalisées sur des *systèmes industriels complexes*. La démarche adoptée est dite « descendante », d'un point de vue global à des points de vue locaux : elle est parfaitement complémentaire avec celles vues en mathématiques et en sciences physiques et chimiques.

Dans ce contexte, et afin de suivre au mieux l'augmentation importante de la complexité des systèmes depuis la fin des années 1990, une évolution importante est proposée sur le nouveau programme des CPGE : l'introduction d'une *initiation* à l'*Ingénierie Système* (en anglais : *Systems Engineering*), méthode d'analyse fondamentale pour les systèmes techniques industriels.

Pour aboutir à cette réflexion de l'ingénieur, il existe de nombreux outils, moyens et méthodes : en phase de standardisation et d'ores et déjà considéré comme une référence, le langage SysML (*Systems Modeling Language*) a été choisi pour la *modélisation* et/ou l'*analyse* de la *complexité des systèmes industriels*. Ce langage purement graphique est présenté dans le chapitre 2.

3. Le contexte de travail dans l'entreprise

3.1. Fonctionnement d'une entreprise

L'entreprise [2] constitue le cadre du travail de l'ingénieur.

Définition 1.1. Entreprise

Une entreprise est une association de personnes mettant en commun des ressources intellectuelles, financières et matérielles pour la conception, la réalisation, la commercialisation et le suivi d'un produit ou d'un service à destination d'usagers appelés « clients ».

Les personnes concernées par cette association sont celles qui travaillent physiquement dans l'entreprise (ouvriers, cadres, chercheurs dans la partie recherche et développement, etc.) mais également les investisseurs et les fournisseurs (cette liste n'est bien évidemment pas exhaustive).

Dans la phase de création d'un bien, plusieurs entités sont concernées :

- les employés de l'entreprise (ouvriers, techniciens, cadres, etc.) mettent à disposition leurs compétences et attendent en retour un salaire ;
- les fournisseurs et les sous-traitants, externes à l'entreprise, mettent à disposition leur savoir-faire et attendent en retour le règlement de leurs factures ;
- les investisseurs mettent à disposition leur capital et attendent en retour des dividendes.

Si le produit conçu, réalisé et mis en vente est acheté par le client, le « retour sur investissement » assure la pérennité de l'entreprise et lui permet d'innover en améliorant les caractéristiques et les performances

[2] Ce terme est à prendre dans son sens le plus large : il s'agit tout aussi bien des industries que des fournisseurs de services (banques par exemple), voire même, et de plus en plus avec les exigences d'efficacité et de concurrence internationale, les centres de recherche, qu'ils soient universitaires ou industriels.

des biens produits. Le retour sur investissement est alors en partie redistribué aux membres de l'entreprise [3] et sert à la faire fructifier en améliorant la qualité des biens disponibles sur le marché.

3.2. Le rôle du client dans le processus de conception

Le schéma présenté précédemment n'est bien entendu cohérent que si le client achète le produit !

Le plus souvent, plusieurs entreprises sont en concurrence et proposent des produits similaires et le client a donc le choix. Si une entreprise propose un produit en décalage avec ce que le client attend, celui-ci achètera un produit concurrent, ce qui peut entraîner la disparition de l'entreprise si la majorité des clients réagissent de la même façon.

Afin d'éviter cette situation critique, il est vital pour une entreprise :

- *en amont* de la mise sur le marché, d'*identifier* et même de *modéliser* (pour prédire les probables évolutions) les *attentes du client* ;
- *en aval* en cas de vente décevante, d'*analyser la chaîne de responsabilités* lors du processus allant de la *conception* à la *vente* du produit.

Les ouvriers, les investisseurs [4] et les fournisseurs ne peuvent être tenus comme responsables de l'échec commercial d'un produit : les seuls responsables sont les cadres (ingénieurs et commerciaux) dont le rôle est de définir, piloter et garantir la réussite d'un projet.

Les cadres de l'entreprise doivent donc mettre en œuvre des méthodes permettant de s'assurer que le client achètera le produit et en sera satisfait. Cette réflexion doit être initiée dès le début du projet puis entretenue tout au long du processus de conception et de réalisation, ce qui n'est pas simple, surtout si les enjeux économiques sont importants et les délais impartis courts.

La démarche de conception a pour objectif non de concevoir un produit en espérant un hypothétique client, mais de *satisfaire le client* en proposant un produit qui lui convient.

Les rôles sont inversés : *le client est au centre des priorités et le produit n'est qu'un moyen* [5].

3.3. Le cahier des charges fonctionnel

3.3.1. Positionnement du cahier des charges dans la conception d'un produit

Comme indiqué précédemment, *le client est obligatoirement au départ de la démarche de conception* : la première action consiste donc à définir précisément ce qu'il attend du produit afin que la solution technique conçue, produite et mise en vente corresponde au maximum à ses attentes.

La définition de l'attente du client est formalisée dans un document normalisé appelé *cahier des charges fonctionnel* (CDCF), qui « modélise » les attentes du client.

La conception du produit, objet réalisé et vendu au client, consiste ensuite à trouver une solution technique répondant scrupuleusement ou en très grande partie à toutes les requêtes du client, donc à tous les critères du cahier des charges [6].

[3] L'histoire montre qu'il peut y avoir matière à polémique sur les proportions redistribuées à chaque catégorie mais cela sort du cadre de cette présentation.

[4] Ce point est à atténuer si l'investissement financier est insuffisant ou décalé dans le temps ou si les attentes sur le retour sur investissements sont irréalistes.

[5] Le « processus de conception » devrait peut-être d'ailleurs s'appeler « processus de satisfaction du client ».

[6] L'ensemble des informations sur la structure du système lorsqu'il est finalement conçu à partir des attendus du cahier des charges est regroupé dans un document souvent appelé « dossier technique de conception » définissant l'architecture matérielle du produit et toutes ses caractéristiques techniques (plans, câblages, matériaux, etc.).

Définition 1.2. Cahier des charges fonctionnel (CDCF)

Le *cahier des charges fonctionnel* (CDCF) est un document ayant une structure normalisée et formalisant ce dont le client a besoin ainsi que l'ensemble de ses requêtes, le tout sans spécifier aucune *solution technique*.

Le processus de conception du produit technique est alors nécessairement itératif, à l'image du diagramme simplifié de la figure 1.2.

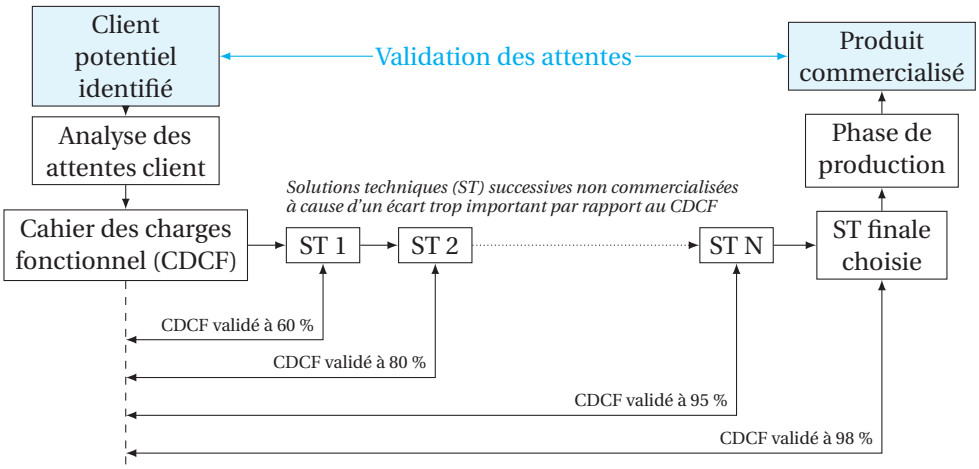


Figure 1.2. Processus de conception partant du besoin du client défini dans un cahier des charges fonctionnel (les valeurs des écarts par rapport au cahier des charges sont indicatives).

Une fois le cahier des charges décidé, une première solution technique est imaginée et ses performances sont estimées par un ensemble de calculs afin d'établir les écarts entre les spécifications du cahier des charges et les performances atteintes. Cette solution, largement imparfaite, sert de base pour une nouvelle où les défauts seront partiellement corrigés.

Après un certain nombre d'itérations, la solution doit s'approcher des demandes du cahier des charges : chaque étape de calcul des écarts est une comparaison avec le cahier des charges et donc avec le besoin du client, ce qui permet d'assurer une convergence vers un produit le satisfaisant au mieux.

À la fin, des compromis sont généralement adoptés sur les critères peu stratégiques du cahier des charges non validés de façon à limiter les délais de conception. Une fois la décision prise sur la solution technique finale qui sera proposée au client, celle-ci est fabriquée et devient un produit commercialisé.

En conclusion, *tous les travaux mis en œuvre dans l'entreprise* (calculs de dimensionnement, choix de conception, suivis de la qualité de la production, etc.) *se justifient par un ou plusieurs critères du cahier des charges* et, inversement, *tous les critères du cahier des charges font l'objet d'une ou plusieurs validations sur le produit réalisé.*

COMPLÉMENT CULTUREL

La norme NF X 50-151 propose un guide pour la rédaction d'un cahier des charges Fonctionnel (CDCF) en recommandant de le composer en quatre parties principales.

Partie 1 – Présentation générale du système

Cette partie importante est destinée à donner toutes les informations utiles concernant le produit : étude de ...

marché, contexte du projet, objectifs, énoncé du besoin, environnement du produit, etc. Il n'y a aucune recommandation sur le volume d'informations ou la forme adoptée.

Partie 2 – Expression fonctionnelle des besoins

Cette partie fondamentale décrit et définit les différentes fonctions de service du produit ainsi que les contraintes et les critères d'appréciation qui y sont associés. Il doit aussi apparaître, associées à ces critères, des spécifications permettant de fixer le niveau d'exigence requis, correspondant le plus souvent à une grandeur mesurable. Dans la mesure du possible, il est conseillé d'ajouter une indication de la flexibilité pour les niveaux d'exigence, soit sous une forme symbolique à niveaux (0 : impératif ; 1 : peu négociable, 2 : négociable, 3 : très négociable), soit sous une forme numérique ou explicite, avec des limites : les flexibilités permettent à l'ingénieur de créer un système moins contraint, donc moins cher.

Les informations sont le plus souvent réunies sous la forme d'un tableau tel que celui ci-après :

Fonctions de service	**Critères**	**Niveaux**	**Flexibilités**
FS1 : déplacer linéairement l'axe	Vitesse	$< 2\,\text{m s}^{-1}$	Impératif
	Amplitude	1 m	± 5 mm
	Masse entraînée	4 kg	Maximale
	Précision du positionnement	1 mm	Impératif
	Dépassement transitoire	< 10 %	Conseillé
	Stabilité (critère en gain)	MG > 10 dB	Conseillé
	Stabilité (critère en phase)	Mφ > 45 °	Impératif

Les critères indiquent sur quelle(s) donnée(s) physique(s) agir pour réaliser la fonction. À une fonction de service particulière peuvent être associés plusieurs critères, chacun étant accompagné d'un niveau qui indique la plage de valeur associée à la grandeur du critère avec une indication claire des unités.

Partie 3 – Appel à des variantes / Partie 4 – Cadre de réponse

Cette deux parties sont optionnelles. L'appel à des variantes demande et fixe des limites à l'étude d'autres propositions ou d'autres solutions possibles pour réaliser le produit. Le cadre de réponse est destiné à simplifier et à codifier la façon de répondre (présentations, descriptions, etc.) pour faciliter les dépouillements et donc l'analyse des différentes idées.

Informations complémentaires

Le cahier des charges peut être utilisé pour les consultations, les appels d'offres, les adjudications, les marchés négociés entre partenaires (y compris entre services d'une même entreprise), la conception pour un coût objectif (CCO), etc. L'organisation d'un CDCF fait appel au demandeur (personne ou société responsable du financement), au décideur (responsable du projet ou personne qui suit le développement du produit), à l'animateur (responsable de l'élaboration du CDCF) et au concepteur-réalisateur (conception et fabrication du produit).

Attention !

La phase de production mise en évidence sur la figure 1.2 (page 15) est fondamentale et, comme la phase de conception, a une structure itérative.
La conception d'une ligne de production permettant de réaliser le produit « au plus près » de la solution technique choisie en minimisant et en maîtrisant les écarts entre le produit réalisé et la solution technique prévue sur le papier assure le développement et la réussite économique d'un produit. Le client achetant un produit « réel » et non un concept virtuel, les solutions techniques choisies dépendent des moyens de fabrication : il est donc nécessaire de vérifier la validation de la fabrication au fur et à mesure de la conception.
Un « bon » concepteur doit donc nécessairement avoir une connaissance minimale des processus de fabrication, des coûts associés ainsi que de la qualité pouvant être atteinte : cette partie est étudiée en filière PTSI / PT et elle n'est pas abordée dans le cadre de cet ouvrage.

3.3.2. Mise en œuvre du cahier des charges

Pour la mise en œuvre du CDCF, deux cas peuvent être distingués :

1. le client demande un produit « sur mesure » à une entreprise ;
2. l'entreprise souhaite mettre sur le marché un produit vendu aux clients sur des points de vente.

Dans le premier cas, le cahier des charges s'établit sous la forme d'un contrat spécifiant la fonction à assurer et les critères de performances chiffrés du produit. Ce premier cas correspond aux échanges entre entreprises [7] : le client et l'entreprise définissent ensemble le cahier des charges.

Le second cas correspond aux échanges entre les entreprises et les particuliers [8] : l'entreprise fixe elle-même le cahier des charges en déterminant les besoins des clients qu'elle cible, ce qui nécessite une étude commerciale préalable.

Le cahier des charges spécifie la ou les fonctions attendues du produit, les critères de performance permettant de juger si la fonction est correctement accomplie, les niveaux de performance associés et enfin la flexibilité de chaque critère (niveau de « négociabilité » du critère).

Attention !

Le cahier des charges doit être aussi exhaustif que possible. Pour cette raison, il est élaboré à partir de documents préparatoires listant, entre autres, les différentes phases de vie du produit et les différents « acteurs » interagissant avec le produit.

La première étape de mise en place d'un cahier des charges consiste donc à exprimer le besoin du client et spécifier les critères principaux du cahier des charges. Cette étape est critique car une erreur sur l'interprétation du besoin du client va conduire, au terme de tout le processus de conception et de fabrication, à un échec commercial du produit.

Les phases de vie rassemblent les différents cas d'utilisation du produit parmi lesquels les phases de réalisation, d'utilisation auprès du client [9], de maintenance et de recyclage.

Dans chacune des phases, différentes contraintes et « acteurs » apparaissent et apportent des critères au cahier des charges : leur liste exhaustive permet de ne pas oublier certaines contraintes. Ces travaux préparatoires sont prévus dans le cadre de la modélisation par le langage SysML (voir chapitre 2).

Exemple Véhicule automobile

Un constructeur automobile souhaite introduire sur le marché deux véhicules : un véhicule de type « cabriolet sportif individuel » (concurrence : BMW Z4, Audi TT, Peugeot RCZ ou Porsche Boxster) et un véhicule de type « berline monospace familiale » (concurrence : Citroën Picasso, Renault Scenic ou Volkswagen Touran).
Pour développer un tel produit technique, la formulation classique (« se déplacer », « effectuer des trajets courts ou longs », etc.) n'indique rien sur ce qui poussera le client à entrer chez un concessionnaire et acheter un de ces véhicules particuliers.
Il est donc nécessaire d'identifier le besoin principal et d'identifier les critères principaux pour les deux véhicules qui seraient, par exemple, ceux précisés sur le tableau suivant :

[7] Connus sous l'acronyme « B2B », comme *Business to Business*.
[8] Connus sous l'acronyme « B2C », comme *Business to Consumer*.
[9] Les phases d'utilisation sont souvent multiples donc difficiles à énumérer simplement.

Critères	Cabriolet sportif	Berline Monospace
Descripteurs	Luxe et image sociale	Confort et praticité
Passagers	2 à 4 personnes	5 à 7 personnes
Coffre	Critère non significatif	$\geqslant$ 210 litres
Trajets type annuels	Péri-urbains < 30 000 km	Inter-urbains > 80 000 km
Puissance	$\geqslant$ 180 cv	$\geqslant$ 120 cv
Motorisation	Essence uniquement	Essence ou diesel
Bruit moteur Niveau bruit à 50 m	Rauque et typé sport 70 dB < Bruit < 80 dB	Minimal et doux Quasi inaudible
Consommation	Non significatif	$\leqslant$ 6 litres / 100 km
Vitesse maximale	$\geqslant$ 220 km/h	$\geqslant$ 150 km/h
0 - 100 km/h	$\leqslant$ 8 s	$\leqslant$ 18 s
Conducteur potentiel (d'après une étude sociologique réalisée en 2005 pour PSA)	Homme célibataire aux hauts revenus en recherche de visibilité sociale, déplacements journaliers inter-urbains	Famille de deux à trois enfants, déplacements journaliers de grande distance et inter-régionaux aux vacances
Quelques caractéristiques incitatrices à l'achat du véhicule par le client	Tableau de bord en bois précieux, sièges baquet en cuir, design sportif, toit escamotable en option, etc.	Régulateur et limiteur de vitesse, sièges lavables, coupe classique, nombreuses options, fiabilité éprouvée, etc.
Éléments de sécurité du véhicule	Airbags frontaux conducteur et passager	Airbags frontaux et latéraux à l'avant et l'arrière
Prix selon options	35 k€ $\leqslant$ Prix $\leqslant$ 65 k€	10 k€ $\leqslant$ Prix $\leqslant$ 25 k€

Bien qu'il s'agisse de deux véhicules du même constructeur, les critères sont évidemment très différents : le prix (critère pratiquement toujours présent dans un cahier des charges) doit en particulier être suffisamment élevé pour le cabriolet pour satisfaire l'ego de l'utilisateur tout en restant accessible à la clientèle cible et suffisamment raisonnable avec possibilité de nombreuses options pour la berline familiale.
À partir du besoin exprimé, le cahier des charges des deux véhicules peut se mettre en place facilement : débuter la conception de tels véhicules sans avoir compris et identifié les besoins principaux des clients, c'est s'engager vers un échec à plus ou moins long terme.

Les critères du cahier des charges sont généralement très nombreux (plusieurs milliers sur une voiture) et le produit très complexe : il est donc impossible de proposer une solution technique répondant au cahier des charges dans une démarche purement déductive, à l'image d'une démonstration mathématique sous la forme « Client → Cahier des Charges → Solution technique → Produit ».

Exemple Définition d'un cahier des charges selon le type de produit

Afin d'aboutir à un produit correspondant aux attentes du client, la définition du cahier des charges fonctionnel d'une cafetière à capsules, des turbopropulseurs d'un avion de ligne et d'un véhicule de tourisme ne peut être réalisée par les mêmes groupes de personnes.
Une cafetière à capsules est un produit « marketing ». Lors de la conception de ce produit, des commerciaux ont mis en place des enquêtes pour définir une cible constituée d'un ensemble important de clients ayant un besoin très similaire. Le cahier des charges a été élaboré sur les

résultats de cette enquête et les ingénieurs ont ensuite trouvé une solution au problème posé.
La conception des turbo-propulseurs d'un avion de ligne fait appel à des critères techniquement très pointus de par le contexte même du produit. Lors de la conception de ce produit, seuls des ingénieurs ont participé à l'élaboration du cahier des charges.
Dans le cas d'un véhicule de tourisme (par exemple le modèle Clio de Renault), il y aura nécessairement un groupe de travail rassemblant commerciaux et ingénieurs pour imaginer un cahier des charges adapté à la cible et techniquement possible et innovant.

4. L'Ingénierie Système (IS)

Les éléments présentés dans ce chapitre sont en grande partie basés sur les documents diffusés par l'INCOSE ou l'AFIS [10], et plus particulièrement ceux réalisés par Dominique Luzeaux, Ingénieur Général de l'Armement, ancien président de l'AFIS et auteur, avec plusieurs collaborateurs, de nombreux ouvrages sur l'Ingénierie Système.

4.1. Qu'est-ce que l'Ingénierie Système ?

En 1996, le Standish Group [11] a publié des données révélatrices de la manière dont les projets industriels étaient gérés à cette époque :

- seuls 16 % des projets sont terminés dans le respect du cahier des charges ;
- 31 % des projets n'aboutissent pas ;
- 45 % des projets ont un dépassement budgétaire de plus de 50 % dont 11 % avec des dépassements supérieurs à 200 % (soit plus du triple du budget initialement alloué) ;
- 57 % des projets ont un retard de plus de 50 % dont 10 % avec un retard supérieur à 200 % (soit plus du triple du temps initialement alloué).

Ces chiffres posent un véritable problème de définition du projet, ce qui justifie la mise en œuvre d'une démarche de réflexion et d'étude plus rigoureuse : c'est l'objectif de l'Ingénierie Système.

Définition 1.3. Ingénierie Système

L'Ingénierie Système (en anglais *Systems Engineering*) est une approche scientifique interdisciplinaire dont le but est de formaliser et d'appréhender la conception de systèmes complexes avec succès. Le but de l'Ingénierie Système est donc l'analyse des échecs antérieurs afin d'apporter des solutions pour éviter qu'ils ne se reproduisent.

[10] L'**In**ternational **Co**uncil on **S**ystems **E**ngineering, dont le siège social est à San Diego (États-Unis), est un organisme non lucratif dont les objectifs sont la promotion, l'accompagnement par la formation, l'application et la diffusion de l'Ingénierie Système dans les entreprises au niveau international. L'INCOSE a mis en place un programme de certification pour reconnaître formellement la connaissance et l'expérience des ingénieurs dans le domaine de l'Ingénierie Système. L'**A**ssociation **F**rançaise d'**I**ngénierie **S**ystème, branche française de l'INCOSE, a les mêmes objectifs au niveau des entreprises françaises.

[11] Société créée en 1985 et basée à Boston (États-Unis) dont le but est la collecte d'informations sur les « flops » technologiques observés lors de la mise en place des systèmes d'information : à partir de l'analyse des données ainsi réunies, cette société prodigue des conseils adaptés aux entreprises afin qu'elles ne commettent pas les mêmes erreurs. La mission de ce groupe est donc de faire en sorte, par le conseil et l'analyse de solutions, que les projets industriels soient réussis et que les investissements soient pertinents. Depuis le début des années 2000, l'expertise a été étendue aux projets industriels pluritechniques.

En effet, si près d'un tiers des projets n'aboutissent pas, les causes des échecs sont diverses (les points couverts directement ou indirectement par l'Ingénierie Système sont indiqués par [IS]) :

- 12,8 % : manque de prise en compte des utilisateurs [IS] ;
- 12,5 % : exigences et spécifications incomplètes [IS] ;
- 11,8 % : changement des exigences et spécifications au cours de la conception [IS] ;
- 7,5 % : manque de soutien de la direction ;
- 7,0 % : incompétences sur les technologies [IS] ;
- 6,4 % : manque de ressources ;
- 5,9 % : attentes non réalistes ;
- 5,3 % : objectifs non clairement explicités [IS] ;
- 4,3 % : délais non réalistes ;
- 3,7 % : mauvaise maîtrise des nouvelles technologies ;
- 23 % : autres causes (marché mouvant, concurrence internationale, etc.).

Cette approche, très ancienne dans sa démarche, est devenue nécessaire avec l'accroissement de la complexité des systèmes. Elle a été formalisée de manière rigoureuse à partir de 2001 par les normes IEEE 1220, EIA 632 et ISO 15288.

4.2. But de l'Ingénierie Système

Afin d'éviter des conceptions insuffisantes voire erronées, il est nécessaire de *modéliser tout ce qui se rapporte au produit* : structure, comportement, attentes des clients, contexte économique, etc.

Cette démarche est connue sous l'acronyme *MBSE* (***M**odel **B**ased **S**ystem **E**ngineering*) et le langage SysML, présenté au chapitre 2, est l'outil privilégié pour son développement.

En effet, sans modélisation, les discussions entre parties prenantes (en anglais *stakeholders*) sont difficiles, chacun ayant sa propre culture interne : dans une entreprise, les outils comme les éléments de langage propres aux différents corps de métier peuvent induire de véritables incompréhensions.

L'évolution du point de vue (plus exactement du paradigme) d'une gestion de documents à une gestion de modèles est nécessaire car, entres autres :

- les modèles sont beaucoup plus formels, complets, riches au niveau sémantique et polyvalents que n'importe quel document, quelle que soit sa structure et son éventuelle exhaustivité ;
- par ses multiples points de vue d'un même modèle, chacun, quelles que soient ses compétences et connaissances, sera en mesure de l'utiliser et de le modifier à son meilleur niveau ;
- l'évolution du projet au cours du temps (versions successives d'un même système, modifications des exigences, etc.) est simplifiée et de multiples hypothèses peuvent être testées ;
- le partage d'une partie seulement du projet (sous la forme d'un paquetage) est possible avec un partenaire (un sous-traitant par exemple), sans lui donner accès à toutes les informations.

En Ingénierie Système (IS), la définition du système comporte :

- celle de ses sous-systèmes et constituants (matériels, logiciels, organisations et compétences humaines) et de leurs interfaces, sièges des interactions recherchées ;
- celles des processus de leurs cycles de vie permettant de les concevoir, produire, vérifier, distribuer, déployer, exploiter, maintenir en condition opérationnelle et retirer du service, et donc des produits contributeurs nécessaires à ces processus.

Cette approche de la définition induit une démarche descendante d'ingénierie s'appuyant sur une décomposition itérative du système en blocs constitutifs. Les constituants sont alors définis avec leurs interfaces ainsi que les produits contributeurs à leur cycle de vie.

L'objet que constitue le système ne se conçoit qu'à travers la fonction ou la tâche accomplie. Une prise de recul est nécessaire pour appréhender le système d'un point de vue fonctionnel : le « que fait-il » doit prévaloir sur le « comment fonctionne-t-il ».

4.3. Processus de conception des produits complexes

4.3.1. Approche système des produits modernes

Notion de système

Les systèmes existent dans des domaines très variés (biologie, économie, linguistique, philosophie, management des organisations, etc.). Dans le cadre de la formation scientifique, les systèmes techniques seront particulièrement étudiés, mais les méthodes de raisonnement peuvent être transposées à tout système : en effet, les produits modernes nécessitant une ingénierie avancée sont bien souvent des *systèmes complexes* voire des *systèmes de systèmes*.

Définition 1.4. Notion de système (définition de l'AFIS à partir de la norme)

Un système est un ensemble d'éléments en interaction entre eux et avec l'environnement, intégrés pour rendre à son environnement les services correspondant à sa finalité.
Un système présente donc des propriétés nouvelles résultant des interactions entre ses constituants et est bien plus qu'un ensemble de composants : les flux d'information, d'énergie ou de matière échangés entre les composants sont essentiels dans le comportement global.

La majorité des systèmes actuels sont dits « complexes ».

Définition 1.5. Notion de complexité

Un système est dit complexe lorsque les interrelations entre les composants sont multiples et interdépendantes : le comportement global n'est donc pas directement prévisible à partir des comportements élémentaires des composants.

Il est important de ne pas confondre complexe et compliqué : certains systèmes très simples d'utilisation (par exemple les téléphones tactiles) ont en réalité une structure fonctionnelle très complexe.

Étude du comportement des systèmes complexes

Le comportement des systèmes complexes dépend souvent d'un grand nombre de paramètres et de conditions initiales influant fortement sur la réponse dynamique [12].

Un système complexe est bien souvent pluritechnique et son analyse nécessite la coopération de spécialistes de plusieurs disciplines. L'évolution des grandeurs au cours du temps n'est généralement accessible que par la simulation numérique ou la mesure expérimentale.

L'analyse des systèmes complexes requiert une organisation intellectuelle différente de celle des systèmes simples [13]. Un système complexe s'étudie.

- en triant les entrées et sorties significatives ;
- en hiérarchisant l'organisation interne en niveaux ;
- en cherchant des relations entre paramètres ;
- en identifiant des critères de comparaison et les interactions entre les grandeurs ;
- ou en proposant des solutions ou modèles partiellement valides.

[12] C'est le cas par exemple de la météorologie ou des organismes vivants, qui sont parmi les plus représentatifs de tels systèmes dont la compréhension est encore aujourd'hui très sommaire.

[13] Les systèmes peuvent s'étudier par un raisonnement déductif (hypothèses → modèle → résultat → conclusion), par isolements successifs des phénomènes élémentaires (et validations par des expériences) ou par relations de cause à effet.

4.3.2. Représentation des systèmes complexes

Dans un système complexe, les flux de matière, d'énergie ou d'information échangés entre les composants, les relations orientées ou non et les bouclages ne permettent pas de décrire un système simplement sous la forme d'un texte ou d'un discours et l'utilisation d'un support graphique devient rapidement indispensable. En conséquence, la *représentation* la *mieux adaptée* pour décrire un système complexe est *nécessairement graphique*.

Dans un système complexe, la présence de niveaux hiérarchiques nécessite souvent un assemblage de représentations graphiques organisées par niveaux et par points de vue : le langage SysML développé au chapitre 2 est un des outils adaptés à cette étude.

4.3.3. Démarche d'analyse des systèmes complexes

Une approche classique, dite académique, s'attache généralement à isoler les composants élémentaires d'un système, poser les propriétés ou le modèle de ces composants, puis assembler progressivement ces propriétés ou modèles pour en déduire le comportement global.

Ce type d'analyse est dite *ascendante* (ou *bottom-up*, du bas vers le haut) et part d'une description des phénomènes de base pour « remonter » au comportement global, ce qui ne peut raisonnablement se faire que si la complexité du système étudié reste limitée ou maîtrisée.

L'analyse d'un système complexe s'adapte mieux à une approche *descendante* (ou *top-down*, du haut vers le bas) où, à partir de la description globale d'un système sous forme d'une « boîte noire » liant les entrées et les sorties, l'architecture est progressivement détaillée par niveaux hiérarchiques pour aller jusqu'aux détails de conception. En phase de conception, cette approche est incontournable pour que des spécialistes puissent travailler en parallèle sur différentes parties en gardant une cohérence d'ensemble. Bien souvent, cette analyse descendante s'arrête à l'échelle des composants qui seront incorporés pour réaliser le produit [14].

> **Exemple** Analyse d'un *smartphone*
>
> L'approche descendante est très naturelle pour l'étude des systèmes techniques complexes tels que, par exemple, un nouveau modèle de smartphone où l'utilisateur appréhende globalement le produit à partir de tests divers pour en connaître le fonctionnement sans chercher à aucun moment à savoir comment le système est conçu au niveau électronique ou informatique.

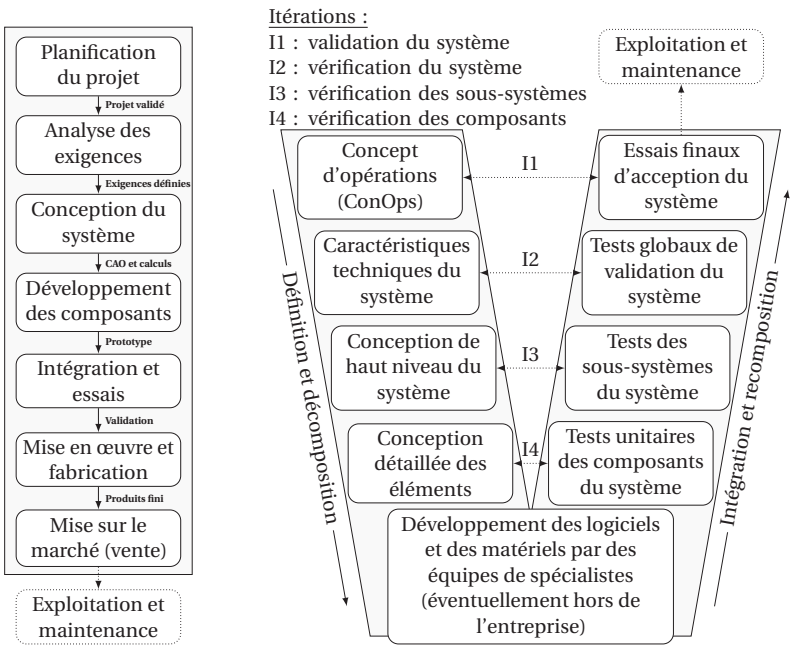
4.3.4. Méthodes de conception en Ingénierie Système (IS)

L'Ingénierie Système est la démarche de conception des systèmes complexes en entreprise. L'aspect pluritechnique de tels systèmes implique :

- La participation de spécialistes de cultures différentes : il faut donc des outils de communications communs et le langage SysML, présenté au chapitre 2, est un de ces outils.
- Les délais de conception courts nécessitent un travail parallèle des équipes, ce qui rend difficile la mise en place d'une organisation du travail efficace.
- Les interrelations entre les composants et les performances à atteindre nécessitent une adaptation permanente des paramètres.

[14] Ainsi, lors de la conception d'un véhicule, certains composants comme les moteurs d'essuie-glace, les amortisseurs, etc, sont achetés chez des fournisseurs : l'analyse descendante s'arrête au niveau de l'interface avec ces composants gérés par le sous-traitant. De même en informatique, la conception d'un logiciel va s'aider de bibliothèques de fonctions : l'analyse descendante s'arrête aux fonctions appelées, sans se soucier de la façon dont elles sont programmées.

De plus, les échanges avec le client en cours de conception conduisent parfois à modifier le cahier des charges et donc à réajuster les solutions techniques en cours d'élaboration [15].



(a) En cascade

(b) En « V » (Source : IBM)

Figure 1.3. Processus de conception « en cascade » et « en V ».

Ces contraintes propres au contexte de l'ingénierie ont conduit à remettre en cause le schéma de conception linéaire, de l'expression du besoin à la maintenance du produit en situation d'usage, dit « en cascade » (figure 1.3(a)). Ce schéma présume qu'une activité de conception ne débute que lorsque la précédente est définitivement terminée, ce qui limite son efficacité.

Une méthode classiquement répandue dans le milieu industriel est le « cycle en V » [16] (figure 1.3(b)).

Le cycle en V décline deux phases dans la conception :

- Dans la *phase descendante* (à gauche), le problème global est morcelé en sous-problèmes et des choix technologiques sont proposés, ce qui aboutit à la définition de chaque composant.
- Dans la *phase ascendante* (à droite), la solution technique précise est vérifiée à l'aide de calculs numériques ou d'essais expérimentaux, d'abord localement puis au sein d'ensembles plus globaux, jusqu'au produit final.

À chaque étape, si les tests de validation sont négatifs, il y a itération, c'est-à-dire modification des paramètres de la solution technique et test à nouveau.

[15] Ceci permet de suivre les innovations des concurrents industriels afin de ne pas mettre sur le marché un produit non conforme : le marché des smartphones, où la concurrence est très poussée, est révélateur de cette nécessité.

[16] Ce n'est pas la seule : il existe des méthodes beaucoup plus poussées et optimisées mais, dans le cadre d'une formation initiale à la démarche de l'ingénieur, le cycle en V a l'avantage d'une vision cohérente de la démarche de conception utilisée en entreprise. Le cycle en V et les autres méthodes (méthode AGILE, analyse PESTEL, cycle en spirale, etc.) constituent simplement un ensemble de « bonnes pratiques » pour organiser au mieux la démarche de conception.

Attention !

Cette démarche permet de diviser le système complexe en sous-composants, en définissant clairement le périmètre de chaque composant et ses contraintes vis-à-vis de son environnement. Il est ainsi possible aux équipes de travailler en parallèle au niveau inférieur du « V » et y assurer la cohérence dans les phases d'intégration des composants car les itérations aux niveaux hauts sont beaucoup plus coûteuses que les itérations aux niveaux bas.

Le cycle « en V » met en évidence que les *coûts effectifs* augmentent significativement en fin de projet, mais les *coûts engagés* sont pour l'essentiel fixés en tout début de la phase de conception :

- En début de projet, une petite équipe définit les grandes lignes de conception et les choix technologiques majeurs qui verrouillent une grande part des coûts futurs.
- Lors du développement, un grand nombre de personnes travaillent à la définition précise des solutions techniques mais les marges de manœuvre sont réduites.
- Lors de la fabrication, les coûts matériels deviennent prépondérants tandis que toutes les décisions ont été prises et toute modification a alors un impact financier très lourd.

Les étapes préliminaires du projet sont donc décisives : l'utilisation d'outils rigoureux et transversaux comme le langage de modélisation SysML, présenté au chapitre 2, permet d'organiser les phases de spécification et de conception du système, et donc de réduire l'augmentation des coûts engagés en permettant de retarder les choix définitifs de conception.

4.4. Bibliographie partielle

Les documents consacrés à l'Ingénierie Système sont nombreux mais le plus souvent adaptés aux métiers, donc difficiles d'accès au niveau des classes préparatoires. Les deux ouvrages ci-après permettent d'aller plus loin que les éléments présentés dans ce chapitre :

- *L'ingénierie système* (collection « 100 questions pour comprendre et agir », Afnor éditions) par Dominique LUZEAUX et Jean-René RUAULT : ouvrage concis et clair sur le positionnement de l'ingénierie système dans la formation d'un futur ingénieur.
- *Découvrir et comprendre l'ingénierie système* (collection « AFIS », Cépaduès éditions) par un groupe d'auteurs sous la direction de Serge FIORÈSE et Jean-Pierre MEINADIER : ouvrage d'un accès moins direct que le précédent, très intéressant en complément.

De nombreuses interventions de Dominique LUZEAUX et Jean-René RUAULT sont disponibles sur Internet en format PDF et constituent une source assez conséquente d'exemples intéressants car accessibles pour la plupart au niveau des CPGE. Il existe bien entendu de nombreuses autres sources d'information, parmi lesquelles la norme ISO15288 qui couvre les processus d'ingénierie système et les étapes du cycle de vie d'un système technique.

Chapitre 2

La modélisation des systèmes complexes par le langage SysML

1. Objectifs du langage SysML — **2.** Les diagrammes et leurs applications — **3.** Exemple de développement d'un modèle

Le langage de modélisation graphique SysML (*Systems Modeling Language*)
Afin d'adopter une démarche d'ingénierie système cohérente lors du développement d'un produit industriel, il est indispensable de modéliser, outre le système lui-même, tout ce qui lui est afférent : le marché, les fournisseurs, les sous-traitants, les lois et règlements, etc.
Cette modélisation, connue sous le nom de MBSE (*Model Based System Engineering*) peut se faire en utilisant de multiples outils, dont le langage de modélisation SysML présenté dans ce chapitre.
Ce langage de modélisation, développé depuis le début des années 2000 à partir du langage UML, a été formalisé en 2007 (version 1.0) et évolue encore : la version actuellement la plus utilisée est la 1.4. La version 2.0, éditée fin 2017, est en phase d'analyse et de validation par les spécialistes.

COURS

1. Objectifs du langage SysML

1.1. Besoin d'un langage de modélisation universel

Les systèmes techniques actuels sont d'une très grande complexité, tant fonctionnelle que structurelle, incluant le plus souvent des parties mécaniques, électroniques, optiques, thermiques, etc. très performantes. Ils ne peuvent donc être conçus que par un groupe d'experts de spécialités différentes, parmi lesquels des ingénieurs (en acoustique, vibration, électronique, informatique, mécanique, etc.), mais également des économistes, des sociologues, des ergonomes, etc.

L'analyse fonctionnelle d'un système technique est traditionnellement mise en œuvre en utilisant différentes méthodes, auxquelles sont associés plusieurs outils adaptés pour les différents secteurs d'activité. Ces outils, bien que très performants chacun dans leur domaine, sont trop disparates pour donner une vision globale cohérente du système étudié, ce qui rend l'analyse très difficile. Par ailleurs, leur appropriation par des non-spécialistes est le plus souvent ardue, car nécessitant un socle de connaissances conséquent.

L'ensemble des acteurs intervenant dans les phases de vie d'un système technique [1] devant travailler en commun malgré leur culture initiale différente, il est devenu nécessaire de disposer d'un langage commun unique devant répondre à deux contraintes :

- Être *suffisamment simple* à comprendre pour que tout le monde puisse l'utiliser sans formation initiale particulière.
- Être *suffisamment développé* pour ne pas être un frein à la créativité et donc être utilisable dans une phase de développement par des spécialistes.

Afin de concilier ces deux contraintes *a priori* antinomiques, l'idée d'un unique langage graphique mis en œuvre par une interface logicielle permettant d'avoir, selon les besoins, une vision globale (utile pour les non-spécialistes) ou plus locale (nécessaire pour l'optimisation de certains points par les spécialistes), s'est développée au début des années 2000.

Cette réflexion a donné naissance au langage de modélisation des systèmes SysML (***Sys**tems **M**odeling **L**anguage*) qui a une structure standardisée depuis septembre 2007 (version 1.0a, la dernière en date étant la 1.4 d'août 2015 totalement compatible avec la version 1.3 de juin 2012 qui reste la plus utilisée).

1.2. Positionnement du langage SysML pour la modélisation

Définition 2.1. Langage SysML

Le langage SysML a pour objectif de formaliser, de manière graphique et indépendante de l'outil logiciel, les spécifications disparates associées à un système technique complexe.
Il permet, entre autres, de spécifier, concevoir, définir et analyser la structure d'un système, identifier les performances, les limites, l'environnement et les relations avec l'extérieur.
Il a donc avant tout un objectif de documentation de la modélisation adoptée et *n'est donc pas* une méthode d'étude, de réflexion ou de conception d'un produit industriel.

[1] Comme vu au premier chapitre, les étapes les plus souvent évoquées sont l'analyse du marché, la définition des objectifs, la conception, la fabrication, la diffusion, la vente, l'utilisation et enfin le recyclage.

La structure du langage SysML offre aux groupes de concepteurs une nouvelle manière de modéliser un système en leur permettant de construire un modèle global puis de le faire évoluer selon leurs besoins et ceux de toutes les parties prenantes au projet (marketing, vente, diffusion, maintenance, client, recyclage, etc.). Ce langage permet d'aborder un grand nombre de cas techniques et tout aussi bien d'analyser les besoins que de participer au développement de produits : voir figure 2.1.

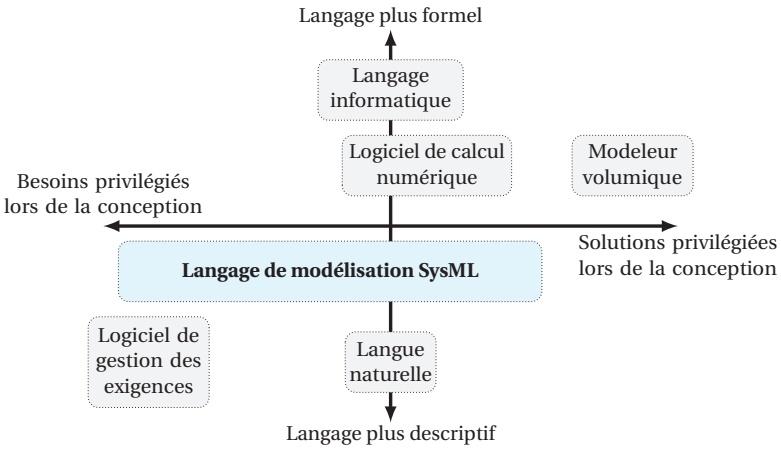


Figure 2.1. Carte de positionnement des outils d'étude des systèmes (d'après document VALEO).

Quelques exemples de logiciel parmi les plus répandus pour les autres langages de la figure 2.1 :

- Langages informatiques : C (tous types), Python ou Java.
- Logiciels de calcul numérique : Scilab ou Matlab avec les boîtes à outils « métiers » (*toolboxes*).
- Modeleurs volumiques : Catia ou SolidWorks avec les noyaux de calcul des mouvements, des vibrations, des déformations, du développement de processus de fabrication, etc.
- Logiciel de gestion des exigences : Doors ou Reqtify.

1.3. Structure du langage SysML

La richesse du langage SysML permet de décrire un système en phase de conception mais son utilisation est tout aussi pertinente pour générer de l'innovation et de la compétitivité à partir de la modélisation d'un système existant : cette phase de « reconception » est appelée « rétro-ingénierie » (en anglais : *Reverse Engineering*) et elle fournit une base solide pour éviter les erreurs liées à une connaissance insuffisante du système.

Grâce à sa structure purement graphique, le langage SysML permet :

- l'expression des besoins et des contraintes, quel que soit leur niveau ;
- la représentation de l'organisation structurée des composants et leur définition précise ;
- la représentation des modes de fonctionnement, des processus internes et externes au système ainsi que les interactions avec son environnement.

Sa structure autorise également des analyses très intéressantes pour les concepteurs telles que :

- la facilitation de la collaboration de tous les spécialistes des corps de métier concernés, en proposant un ensemble lié d'outils de représentation universels et expressifs ;
- la réalisation de la mise à jour, du stockage et du partage ainsi que l'interprétation des informations issues des analyses des travaux des différents intervenants ;
- l'intégration et la mise en relation des différentes composantes techniques, par exemple les liaisons entre un programme informatique et des actionneurs mécaniques ;

- la modélisation du système à toutes les étapes de son cycle de développement et dans sa phase de vie en représentant les éléments du modèle selon différents points de vue ;
- la validation des différentes solutions par une ou plusieurs simulations mises en œuvre à partir des diagrammes d'états, d'activités et paramétrique présentés dans la suite.

2. Les diagrammes et leurs applications

2.1. Points de vue pour l'analyse d'un système technique

Dans une phase de conception ou d'optimisation d'un système technique, il est courant d'aborder le problème selon les trois composantes suivantes :

- les *exigences* auxquelles doit répondre le système pour fonctionner selon les attentes des parties prenantes [2] ;
- le *comportement* attendu du système ou d'un de ses éléments au cours du temps ;
- la *structure du système* qui, selon les besoins de conception et les participants, peut être analysée de manière globale ou locale.

Le langage SysML propose, pour chacune de ces trois composantes, des outils spécifiques graphiques présentés : le langage est ainsi développé autour de neuf diagrammes permettant de représenter *divers points de vues* chacun étant une *partie du modèle complet.*

Cette multiplicité permet le développement de modèles à la fois complets et pertinents.

2.2. Typologies des diagrammes du langage SysML

Le langage SysML s'articule autour de neuf diagrammes, dédiés à la représentation des composantes d'un système : voir figure 2.2.

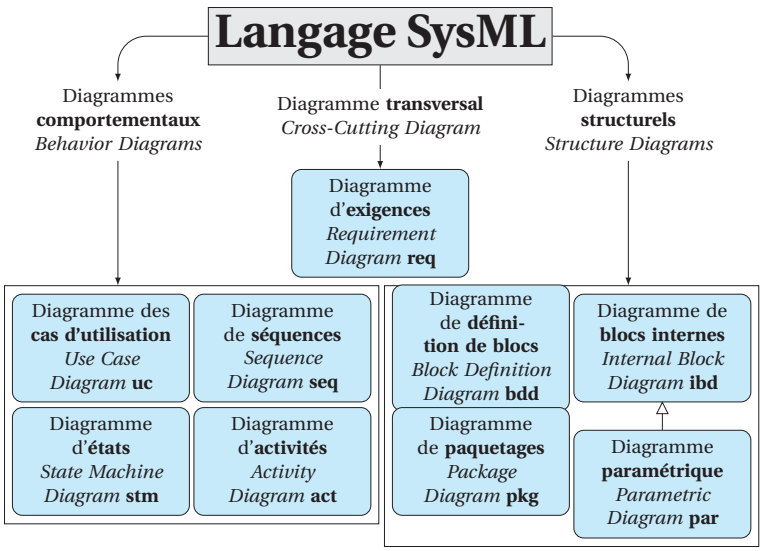


Figure 2.2. Positionnement des diagrammes dans le langage SysML ainsi que leurs identifiants.

[2] Une partie prenante d'un projet, appelée *stakeholder* ou *corporate* en anglais, est un acteur individuel ou collectif (groupe ou organisation) concerné de manière active ou passive par tout ou partie du projet.

Trois points de vue ont été privilégiés dans le langage SysML : un *point de vue comportemental*, un *point de vue structurel* et un *point de vue transversal*.

Quatre diagrammes sont associés au *point de vue comportemental*.

- Le diagramme des cas d'utilisation (*Use Case Diagram*, identifiant **uc**) décrit les objectifs poursuivis par l'acteur primaire à travers le système et donc les interactions de l'acteur primaire (typiquement l'utilisateur) avec le système étudié et d'éventuels acteurs secondaires.
- Le diagramme de séquence (*Sequence Diagram*, identifiant **seq**) décrit l'enchaînement des messages passés entre des instances de blocs en interaction [3] afin de documenter un cas d'utilisation.
- Le diagramme d'états (*State Machine Diagram*, identifiant **stm**) décrit les différents états du système ainsi que les transitions possibles entre ces différents états.
- Le diagramme d'activités (*Activity Diagram*, identifiant **act**) décrit l'enchaînement des tâches (flux « de contrôles ») et l'utilisation des données (flux « de données ») dans un processus.

Quatre diagrammes sont associés au *point de vue structurel*.

- Le diagramme de définition de blocs (*Block Definition Diagram*, identifiant **bdd**) décrit l'architecture matérielle, logicielle, etc., du système.
- Le diagramme de bloc interne (*Internal Block Diagram*, identifiant **ibd**) décrit l'organisation interne d'un élément et les interactions entre ses composants selon des flux qui pourront être de tout ou partie des trois types « énergie », « matière » et « information ».
- Le diagramme paramétrique (*Parametric Diagram*, identifiant **par**) est un cas particulier du diagramme de blocs internes qui décrit les contraintes internes du système à l'aide d'équations liant des propriétés des blocs.
- Le diagramme de paquetages (*Package Diagram*, identifiant **pkg**) décrit l'organisation logique du modèle et les relations entre paquetages en fonction des besoins (facteur d'échelle) afin de clarifier la lecture du modèle *via* des macro-éléments.

Un *point de vue transversal*, spécifique au langage SysML, a été rajouté : le diagramme d'exigences (*Requirement Diagram*, identifiant **req**) est associé à ce point de vue et il décrit ce que doit réaliser le système ainsi que les contraintes auxquelles il doit se soumettre.

2.3. Éléments graphiques des diagrammes

2.3.1. Cadre du diagramme

Chaque diagramme SysML représente un élément particulier du modèle selon un certain point de vue : afin de le repérer, chaque diagramme comporte un « cartouche » présenté figure 2.3, positionné sur la partie supérieure gauche du cadre.

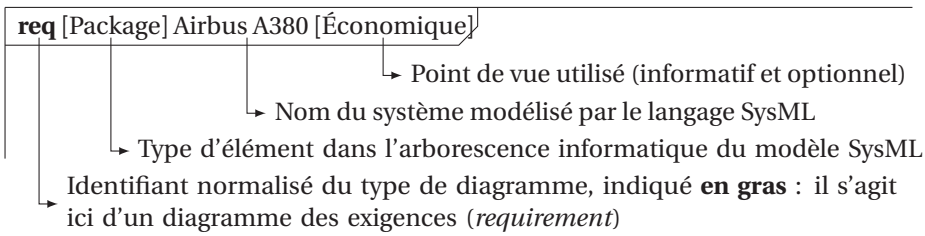


Figure 2.3. Structure du cadre de description des diagrammes SysML.

[3] Ce point sera précisé dans la suite.

Le type de diagramme, repéré par son identifiant (**bdd**, **ibd**, etc. : voir figure 2.2 page 28), est obligatoirement indiqué dans ce cartouche.
L'ajout des autres éléments est optionnel selon la norme mais les logiciels dédiés à la modélisation par ce langage (voir page 31) ajoutent par défaut le type d'élément (bloc ou paquet par exemple) et proposent d'indiquer le nom du modèle mis en place et le point de vue utilisé [4].

2.3.2. Formes géométriques et liens

Les neuf diagrammes du langage SysML (voir figure 2.2 page 28) sont composés des mêmes types de formes géométriques : des rectangles à coins droits ou arrondis, des ellipses et des lignes. Selon les diagrammes, tout ou partie de ces formes géométriques seront utilisées.
Plusieurs types de relations peuvent être rencontrés entre les formes géométriques dans les diagrammes SysML : le tableau de la figure 2.4 regroupe les liens les plus classiques.

Lien	Signification et commentaires
⊕	La relation de **contenance** (aussi appelée inclusion) est représentée par une ligne continue terminée par un cercle contenant une croix du côté du conteneur : elle permet de décomposer une exigence en plusieurs autres plus faciles ensuite à identifier lors de la mise en place du système ou des tests.
→	La relation d'**association** relie deux éléments considérés d'égale importance et elle indique qu'ils sont en lien sans en indiquer la nature. Cette relation peut être unidirectionnelle (dans ce cas elle prend une flèche pour indiquer le sens) ou bidirectionnelle (dans ce cas il n'y a pas de flèche).
- - - - →	Les relations d'**inclusion**, d'**extension**, de **raffinement** ou de **dérivation** d'un cas d'utilisation ou d'une exigence dans un(e) autre sont représentées par une flèche pointillée à pointe ouverte. La flèche est orientée : • du cas d'utilisation global vers un cas d'utilisation partiel inclus avec le mot clé *include* pour l'inclusion ; • du cas d'utilisation partiel vers le cas d'utilisation global avec le mot clé *extend* pour l'extension ; • de l'exigence partielle vers l'exigence globale avec le mot clé *refine* pour l'ajout de précisions, par exemple des données quantitatives, pour le raffinement ; • de l'exigence partielle vers l'exigence globale avec le mot clé *deriveReqt* pour relier de manière dérivée des exigences de niveaux différents, par exemple entre un système et certains de ses sous-systèmes.
▷	La relation de **généralisation** (ou de spécialisation) indique une spécialisation d'un élément (cas d'utilisation, bloc, etc) : elle est représentée par une flèche continue dont la pointe blanche est orientée vers l'élément plus général.
◆	La relation de **composition** permet de relier deux blocs et elle indique qu'un élément est structurellement indispensable à l'autre ; elle est représenté par une flèche dont le losange plein est du côté du composé (ou système principal), l'autre extrémité du côté du composant.
◇	La relation d'**agrégation** a le même rôle que la relation de composition mais elle a un sens moins fort : en général, elle indique que le composant est présent de manière optionnelle ; sa représentation est identique à la composition, mais avec un losange vide du côté du composé (ou système principal), l'autre extrémité du côté du composant.

Figure 2.4. Les principaux liens graphiques du langage SysML.

2.3.3. Commentaires

Dans tous les diagrammes du langage SysML, il est possible d'utiliser des notes graphiques sous la forme de « Post-it® » avec éventuellement les deux mots clés *problem* (problème) pour indiquer les problèmes

[4] Ces indications permettent d'assurer un suivi historique de la mise en œuvre des modèles, surtout si le modèle est partagé entre différents corps de métier.

encore à résoudre et *rationale* (raison) pour justifier certains choix : ces notes permettent de commenter n'importe quel élément d'un diagramme (forme graphique ou lien) et elles seront donc utilisées pour expliciter un point qui pourrait être difficile à comprendre ou modéliser par le lecteur.

2.3.4. Relations entre l'UML et le SysML

Afin de profiter des réflexions initiées au milieu des années 1990 par les concepteurs de logiciels, la structure du langage SysML s'est mise en place à partir du langage UML (***U**nified **M**odeling **L**anguage*), communément utilisé dans ce secteur et standardisé par l'OMG [5].

Un certain nombre de concepts ont été repris de l'UML, d'autres ont été adaptés ou créés afin de répondre aux problématiques spécifiques rencontrées lors de la conception de produits techniques : les langages UML et SysML ont donc des points communs mais sont développés dans un but différent.

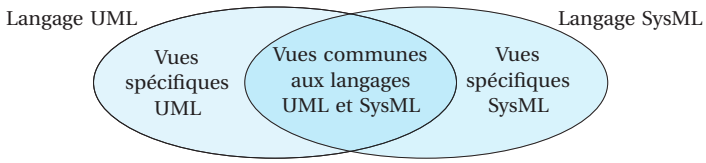


Figure 2.5. Positionnement relatif de l'UML et du SysML.

Les liens entre les deux langages ont un avantage très important : les outils de développement informatiques sont transversaux et les améliorations ou précisions apportées dans un langage peuvent rapidement être adoptées, après adaptation, par l'autre langage.

2.4. Mise en œuvre pratique de ces diagrammes

La mise en œuvre de ces diagrammes se fait *via* des logiciels adaptés parmi lesquels MagicDraw, Artisan Studio, Rhapsody, Modelio ou Visual Paradigm (propriétaires), TopCased et Papyrus (libres).

Ces logiciels, outre une interface de dessin adaptée, permettent, entre autres, de « naviguer » d'un diagramme à un autre et de créer des liens multiples et de différentes natures (site, fichier, autre logiciel, etc.), ceci afin de multiplier les points de vue. Le développement d'un modèle sur ce type de logiciel permet d'éviter des erreurs car, par défaut, seuls les formes graphiques et les liens acceptés par le diagramme en cours seront proposés.

Conseils méthodologiques

Pour information, il est également possible d'utiliser des logiciels de dessin plus universels (parmi lesquels MicroSoft Visio, OmniGraffle, Dia ou Inkscape qui tous proposent des extensions graphiques UML-SysML), mais une grosse partie de l'intérêt de l'utilisation de l'outil informatique pour développer des modèles dans ce langage graphique disparaît, à savoir la navigation entre les différents diagrammes donc entre les différents points de vue d'un même modèle.

La maîtrise de ces logiciels n'est absolument pas attendue par le programme mais leur utilisation comme aide à la réflexion lors des TIPE ou des travaux pratiques peut être très formatrice.

[5] L'**O**bject **M**anagement **G**roup est une association créée en 1989 aux États-Unis et à but non lucratif dont l'objectif est de standardiser et de promouvoir le « modèle objet » sous toutes ses formes.

2.5. Bibliographie partielle

Ce langage est standardisé et dans une phase de développement actif. Les références ne sont pas encore nombreuses et pas toujours adaptées au niveau des CPGE.

La bibliographie qui suit permet donc essentiellement d'aller plus loin dans l'analyse de ce langage, dont on rappelle que *seul le niveau de lecture est attendu dans le cadre du programme* :

- Le site de l'association SysML France, en particulier les onglets *Documents* et *Outils* : `http://www.sysml-france.fr`.
- Le site de l'association française pour l'ingénierie système : `http://www.afis.fr`.
- Une présentation des principales différences entre le langage UML et le langage SysML : `http://www.uml-sysml.org`.
- Le livre de référence de Pascal ROQUES : *SysML par l'exemple - Un langage de modélisation pour systèmes complexes* (disponible en « e-book » PDF sur le site de l'éditeur Eyrolles).

3. Exemple de développement d'un modèle

Attention !

Au niveau de la formation en CPGE, les diagrammes sont proposés uniquement à la lecture : la maîtrise de l'écriture des diagrammes n'est donc pas attendue par le programme.

3.1. Introduction

Ce chapitre présente les neuf diagrammes du langage SysML sur l'exemple particulier d'un drone [6] multi-rotors utilisé pour la prise de vue aérienne lors de la réalisation de films ou de reportages.

Le niveau de description choisi pour les diagrammes est sciemment élémentaire car largement suffisant pour les concours où ils ne sont utilisés qu'en présentation de la problématique.

3.2. Système étudié

3.2.1. Présentation du système

Afin de réaliser des prises de vue aériennes en haute définition lors de la réalisation de reportages ou de films, des drones sont fréquemment utilisés.

Ces appareils sont loués par la production à des entreprises spécialisées qui fournissent, outre le matériel, un pilote maîtrisant parfaitement le vol de son engin et pouvant donc répondre à toutes les demandes du réalisateur et de son chef opérateur [7], le tout pour un prix très raisonnable par rapport aux solutions classiques utilisant des avions ou des hélicoptères : le prix horaire d'au moins 2 000€ HT pour un avion ou un hélicoptère (donc environ 12 000€ HT la journée de location) est réduit à un prix journalier de 1 000 à 3 000€ HT selon les prestations attendues pour le drone.

[6] Un drone (« faux bourdon » en anglais), encore appelé UAV (*Unmanned Aerial Vehicle*), est un aérodyne sans pilote à voilure fixe ou tournante emportant une charge utile et destiné à des missions de surveillance, de renseignement, de combat ou de transport. Les drones civils tels que celui étudié dans ce chapitre sont en plein développement et leurs performances en termes de capacité de vol (durée, maniabilité, etc.) ou de charge transportée progressent continuellement.

[7] Le chef opérateur, aussi appelé « directeur de la photo », est la personne responsable de la prise de vue sur un tournage.

3.2.2. Présentation du besoin client

Cette présentation est issue d'une interview de M. Emmanuel LAURENT, président de la société *Films à Trois*, producteur du documentaire *Le Port englouti de Constantinople* réalisé par Hannes SCHULER et diffusé sur la chaîne ARTE le samedi 28 janvier 2012.

Synopsis du documentaire (présentation du contexte)

À Istanbul, la construction d'une énorme station de métro a révélé, au cœur de la ville moderne, les vestiges de l'ancien port construit sous Théodose II, en usage du IV^e^ au X^e^ siècle de notre ère.

Les travaux du réseau ferroviaire ont été suspendus et la course contre la montre est désormais lancée pour les archéologues qui fouillent sans relâche ce gigantesque site. Ils y ont découvert, conservés dans la vase non oxygénée, des épaves de navires en bois, des objets en tous genres, et des centaines de squelettes. Le conflit entre les ingénieurs des BTP pressés et la lenteur nécessaire des recherches archéologiques est au cœur du documentaire.

Besoin client

Lors de la réalisation de ce documentaire, la production a souhaité qu'une prise de vue aérienne soit réalisée afin de mettre en évidence la configuration et la structure des fouilles réalisées par les archéologues. Ce point n'ayant pas été prévu initialement, la solution d'un survol par hélicoptère a rapidement été écartée, d'une part à cause du prix (dépassant les 3 500 € HT l'heure avec les autorisations de vol) et d'autre part à cause de la configuration du site, entouré d'habitations.

L'utilisation d'un drone s'est révélée être la seule solution selon les besoins définis par le réalisateur :

- Image en taille 16/9 en très haute définition, dite 4K.
- Durée de la séquence après montage : 2 minutes maximum (durée précise non définie lors de la réalisation mais durée effective de 17 s lors de la diffusion du documentaire).
- Dimension de la zone à filmer : 300 m × 300 m.
- Vitesse de déplacement : maximum 1 m s^{-1} en vitesse stabilisée à ± 0,2 m s^{-1} pour que les détails restent visibles.
- Altitude de la prise de vue : inférieure à 60 m.
- Adaptation de la caméra : fixation universelle adaptable à toute caméra ; capacité de charge d'au moins 1 kg (caméra + batterie + système de communication), idéalement de 3 kg.
- Communication au sol : retour en temps réel de l'image au sol en version FWVGA au minimum (format 16/9, résolution 854 × 480).
- Stabilisation en vol : visée d'une cible sans bouger pendant au moins 5 s.
- Résistance à l'environnement : capacité à résister à des rafales de vent de 20 km/h (valeur constatée les jours précédents) sans bouger de plus de 0,5 cm dans chaque direction ni pivoter de plus de 1,5 ° selon les trois axes de roulis, tangage et lacet [8].
- Capacité de vol : au moins 8 h (prévoir plusieurs batteries chargées).
- Durée de la mission : une journée, déplacement et deux nuits d'hôtel pris en charge par la production ; dépassements à la charge de l'entreprise qui loue le drone.
- Prix total maximum pour l'entreprise qui emporte le marché : 4 000 € TTC.

Typologie des drones de cinéma

Pour remplacer la capture par hélicoptère ou avion, deux technologies existent pour la conception de drones utilisés pour la prise de vue aérienne pour les films ou les documentaires : sustentation par aile + entraînement par hélice (comme sur un avion) ou sustentation et entraînement par plusieurs hélices, (comme dans un hélicoptère).

[8] Cette contrainte correspond à la limite de la capacité de traitement de l'image par l'autofocus puis par un traitement informatique en post-production *via* le logiciel Adobe After Effect CS6® très largement utilisé par les professionnels.

La technologie à sustentation par aile et entraînement par hélice est peu onéreuse (quelques centaines d'euros, même pour les plus grands modèles permettant d'embarquer jusqu'à trois caméras à plusieurs focales), a une grande autonomie (jusqu'à une heure voire plus), peut aisément être pilotée depuis le sol grâce à sa grande stabilité naturelle et sa large amplitude de déplacement permet la réalisation de longs travelings. Elle est par contre inutilisable pour la réalisation de plans fixes et le suivi de cibles au sol (acteur ou véhicule en mouvement).

La technologie à sustentation et entraînement par plusieurs hélices permet un pointage et un suivi très précis des cibles au sol, dès lors que leur déplacement reste limité en vitesse et en amplitude. Elle est par contre très onéreuse (entre 10 000 et 80 000€), a une autonomie limitée (moins de vingt minutes sans charge et temps calme), est très sensible à l'environnement et demande une grande technicité pour le pilotage depuis le sol à cause de son instabilité naturelle. La capacité de charge est par ailleurs limitée, ce qui a longtemps été un frein à l'utilisation de cette technologie : dorénavant, cependant, les caméras HD ont des dimensions et des masses compatibles avec les capacités de ces drones à rotors.

Dans le cadre d'étude, seule cette seconde technologie répond à l'appel d'offres.

Caractéristiques du système choisi

La présentation à suivre est issue d'une interview de deux employés de l'entreprise Ciné-Drone (`http://cine-drone.fr`), qui conçoit et exploite des drones pour le cinéma. Des vidéos montrant les capacités de vol et de prise de vue de leur appareil sont disponibles sur le site de l'entreprise.

L'entreprise Ciné-Drone est un acteur important au niveau mondial pour la réalisation de prises de vue par drone. Elle est implantée en France, au Costa Rica, aux États-Unis et en Chine et propose ses services dans le monde entier. Afin de répondre à toutes les demandes des réalisateurs, cette société dispose de trois types de drones à rotors équilibrés dynamiquement à structure carbone et nacelle stabilisée par gyromètre (voir figure 2.6).

Figure 2.6. Drone Ciné-Drone, modèles S et L.

Les principales différences entre les trois modèles S (small), L (large) et XL (extra large) sont fournies sur le tableau de la figure 2.7.

Le tarif est dégressif en fonction de la durée de location et, pour tous les modèles :

- Les pieds sont repliables en vol, ce qui permet à la caméra de viser toutes les directions.
- Le drone peut s'orienter sur ± 180 ° en lacet (rotation autour de l'axe vertical) et ± 45 ° en roulis et tangage (rotations autour de l'axe longitudinal et de l'axe transversal).
- La nacelle supportant la caméra peut s'orienter ± 180°en lacet (rotation autour de l'axe vertical), ± 90 ° en roulis (rotation autour de l'axe longitudinal) et ± 135 ° en tangage (rotation autour de l'axe orthogonal).
- Transmission vidéo au niveau normalisé de 5,8 GHz - 10 mW.
- Conformément à la législation, deux « télépilotes » sont présents pour l'utilisation du drone : le

Caractéristiques	Modèle S	Modèle L	Modèle XL
Moteurs	6 rotors 15 A - 12 V	8 rotors 30 A - 12 V	8 rotors 60 A - 12 V
Charge embarquée	jusqu'à 1,5 kg	jusqu'à 2,3 kg	jusqu'à 5 kg
Charge maximale	3 kg	5 kg	14 kg
Exemple de caméra	Go-Pro 2	Black Magic	Red Epic
Vent maximal	25 km h^{-1}	35 km h^{-1}	65 km h^{-1}
Durée de vol maximale	12 min	8 min	5 min

Figure 2.7. Caractéristiques principales des drones de l'entreprise Ciné-Drone.

premier gère l'attitude[9] de la structure volante du drone par rapport au sol selon les attentes du réalisateur ; le second gère le positionnement angulaire de la nacelle par rapport au drone selon les attentes du chef opérateur.

La qualité de la prise de vue (profondeur de champ, zoom, etc.) est gérée par le chef opérateur grâce au retour vidéo en temps réel. La coordination avec l'équipe de télépilotes permet d'obtenir des images conformes à celles réalisées par hélicoptère.

Le drone est livré sans caméra mais intègre :

- une nacelle alvéolée qui permet de fixer tous les appareils de prise de vue (appareils photo numériques et caméras) existants ;
- un système de communication vidéo avec une prise de connexion universelle pour les caméras professionnelles fournies par l'équipe de tournage.

Les caméras utilisées dans les prises de vue haute définition disposent d'une mise au point autofocus et d'un système de stabilisation optique permettant d'obtenir une image nette malgré les vibrations de l'appareil dès lors que les mouvements globaux (translations et rotations selon les trois directions de l'espace) restent assez lents : le pilotage du drone par un professionnel permet donc d'obtenir une image nette dans la très grande majorité des configurations de vol ce qui réduit la durée, et donc le prix, de la phase de traitement informatique de l'image.

Les drones de cinéma de l'entreprise Ciné-Drone sont par ailleurs équipés d'une gestion spécifique « anti-crash » avec deux niveaux de sécurité :

- Dès que le niveau de batterie devient inférieur au niveau de sécurité (niveau 1), le transfert de flux vidéo est coupé et le drone descend automatiquement au sol en un point défini par GPS : le pilote peut accompagner cet atterrissage mais ne peut pas forcer le maintien en vol.
- Si le système n'a plus les capacités de redescendre au sol à cause d'un niveau insuffisant de batterie (niveau 2), le drone sort ses pieds, coupe les moteurs et déclenche son parachute, ce qui assure un retour au sol sans casse.

Cette double sécurité, définie, conçue et implantée par l'entreprise Ciné-Drone n'est absolument pas obligatoire mais elle permet de préserver l'intégrité des équipements de prise de vue en cas de panne de batterie, ce qui assure la grande renommée de cette entreprise auprès des réalisateurs.

Législation

La France a adopté une des toutes premières réglementations au monde sur l'utilisation des drones civils dans l'espace aérien. Mise en place par la direction générale de l'aviation civile (DGAC), elle permet

[9] L'attitude correspond aux six paramètres permettant de définir la position et l'orientation du drone dans l'espace : déplacements en abscisse, ordonnée et altitude, angles de roulis, tangage et lacet.

l'utilisation d'avions et d'hélicoptères sans pilote pour des applications non militaires ou de sécurité (par exemple agriculture, audiovisuel, inspection de zones ou lutte anti-incendie).

Les télépilotes de drones doivent suivre une formation équivalente en niveau à un brevet de pilotage d'aéronef et demander une autorisation de vol à la préfecture. Le seul scénario qui n'exige pas d'autorisation préalable suppose que l'engin évolue en dehors des zones peuplées, qu'il ne s'éloigne pas de plus de 100 m du télépilote et que celui-ci puisse le suivre à l'œil nu.

Cette réglementation, actuellement la plus restrictive au monde, n'est pas obligatoire en dehors du sol français : la société Ciné-Drone a cependant décidé de l'appliquer, en la renforçant, dans toutes ses interventions dans le monde.

3.3. Diagramme des exigences

3.3.1. Présentation

Définition 2.2. Diagramme des exigences (req)

Le diagramme des exigences, appelé *Requirement Diagram* (identifiant **req**) dans le langage SysML, est le seul diagramme transversal du langage SysML (voir figure 2.2 page 28).
L'objectif de ce diagramme est de modéliser les exigences devant être vérifiées par le système en liant les solutions mises en œuvre sur le système avec les besoins définis dans le cahier des charges. Ce diagramme traduit, par des fonctionnalités ou des contraintes, ce qui doit être satisfait par le système. De nombreux domaines peuvent être couverts, les plus classiques étant les exigences environnementales, économiques, fonctionnelles ou techniques.

Les éléments graphiques utilisés dans ce diagramme sont principalement des rectangles avec un titre représentant les exigences, un identifiant sous forme de numéro et une description textuelle libre mais concise [10]. Il est possible, mais non obligatoire, de relier les exigences entre elles [11] par des liens tels que ceux présentés sur le tableau 2.4 de la page 30.

La nature des liens entre les rectangles modélisant les exigences exprime la manière dont elles sont reliées, il existe donc différentes syntaxes de flèches pour exprimer la nature de la liaison.

Conseils méthodologiques

Ce diagramme devra être le plus simple possible pour rester lisible : il est donc inutile de poser toutes les exigences, sauf cas très particuliers. De plus, pour améliorer sensiblement la compréhension de la problématique, il est possible de réaliser plusieurs diagrammes d'exigences si nécessaire en regroupant par exemple les exigences par secteur.

Remarque

Il est possible d'associer aux exigences des propriétés telles que :

- une priorité, par exemple haute, moyenne ou basse ;
- une indication de la « source », par exemple client, législation ou concurrence ;
- un statut, par exemple proposé, validé, implanté ;
- ou, de manière générale, toute donnée pouvant se rapporter à une exigence devant être validée à un niveau du cycle de vie du produit.

[10] Les logiciels proposent des intitulés « types » tels que, par exemple, « Le système doit … », les pointillés devant être complétés.

[11] Une exigence non connectée à d'autres exigences exprime qu'elle se suffit à elle-même et que sa description indique complètement ce qui est souhaité.

3.3.2. Illustration

Pour le système étudié, il est possible de mettre en place le diagramme présenté sur la figure 2.8 dans lequel les exigences de la production apparaissent comme étant déclinées en trois exigences particulières sur les aspects économique, de norme et technique :

- Économie : le prix maximum admissible pour la réalisation de la séquence est indiqué.
- Norme : le drone doit respecter les normes les plus restrictives, ceci afin d'éviter tout problème avec la législation en cours.
- Aspect technique sur la vidéo et la capacité de vol : ces exigences sont « raffinées » (flèche en pointillés avec le stéréotype « *refine* ») par une précision sur des données numériques attendues en fonction des caméras dont dispose la production.

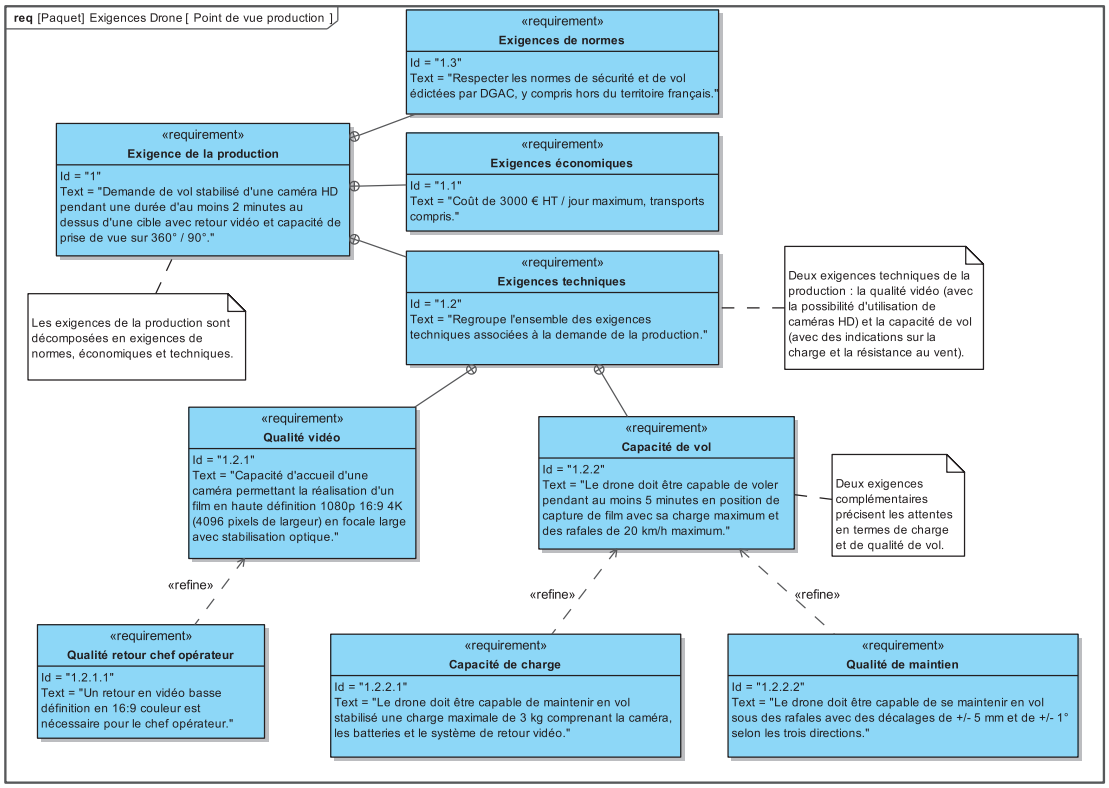


Figure 2.8. Un exemple de diagramme des exigences pour le drone de cinéma.

3.4. Diagramme des cas d'utilisation

3.4.1. Présentation

Définition 2.3. Diagramme des cas d'utilisation (uc)

Le diagramme des cas d'utilisation est un diagramme comportemental, appelé *Use Case Diagram* (identifiant **uc**) dans le langage SysML.
L'objectif de ce diagramme est de montrer les fonctionnalités offertes par un système en identifiant les services qu'il rend : il permet donc de modéliser les exigences selon un point de vue complémen-

taire à celui exposé par le diagramme des exigences (voir page 36).
L'énoncé d'un cas d'utilisation doit se faire *hors technologie*, puisqu'il est défini en termes de résultats attendus.

Les éléments graphiques utilisés dans ce diagramme sont principalement :

- Les acteurs, entités extérieures au système et en interaction avec lui, sont représentés par le pictogramme « bonhomme bâton » et sont reliés à un ou plusieurs cas d'utilisation par une ligne simple appelée *association*.
- Les cas d'utilisation sont représentés sous forme d'ovales. Ils donnent les fonctionnalités du système et sont énoncés du point de vue de l'acteur.
- La frontière du système permet de symboliser les limites du modèle et est représentée par un simple rectangle englobant les cas d'utilisation, les acteurs étant à l'extérieur, à gauche s'ils sont considérés comme « principaux », à droite si ils sont considérés comme « secondaires ».

Conseils méthodologiques

Les fonctionnalités d'un système correspondent à des cas d'utilisation, c'est-à-dire à des services rendus par le système. Il n'apparaîtra donc pas ce qui ne peut être fait par des acteurs extérieurs : ainsi, par exemple, le lavage, la recharge, le recyclage, la réparation, etc. ne doivent pas apparaître si le système n'a pas été développé expressément pour cela.

Remarque

Pour des systèmes simples tels que ceux étudiés en CPGE, ce diagramme le sera également : par conséquent, le diagramme de cas d'utilisation d'un système de laboratoire de CPGE ou d'un sujet de concours comportera typiquement trois à quatre cas d'utilisation au maximum.

3.4.2. Illustration

Pour le système étudié, il est possible de mettre en place le diagramme présenté sur la figure 2.9 où se retrouvent les éléments énoncés précédemment, à savoir :

- Les trois acteurs qui interagissent avec le système (liens avec les cas d'utilisation) et qui pilotent le drone, la nacelle et la caméra.
- Les cas d'utilisation reliés aux acteurs et rédigés selon le point de vue de ces derniers.
- La frontière du système qui contient tous les éléments permettant d'atteindre les objectifs terminaux.

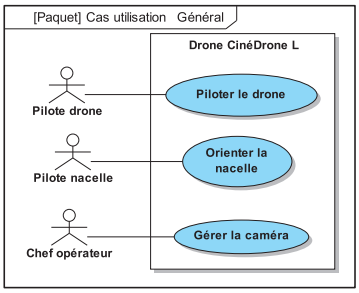


Figure 2.9. Un exemple de diagramme des cas d'utilisation pour le drone de cinéma.

Même si ce diagramme semble très simple voire basique au premier abord, le travail de réflexion pour aboutir à sa représentation l'est moins. Cette simplicité est en général atteinte après plusieurs itérations afin de bien cerner les objectifs visés grâce au système.

Remarque

Il aurait ainsi été possible de rajouter des sous-cas d'utilisation ou des éléments graphiques complémentaires avec, par exemple, des raffinements (« *refine* ») ou des extensions (« *extend* »), etc. : cela n'aurait pas été pertinent car ce diagramme est d'autant plus intéressant qu'il reste simple et compréhensible immédiatement.

3.5. Diagramme de séquence

3.5.1. Présentation

Définition 2.4. Diagramme de séquence (seq)

Le diagramme de séquence est un diagramme comportemental appelé *Sequence Diagram* (identifiant **seq**) dans le langage SysML.
L'objectif de ce diagramme est de décrire les interactions existant entre plusieurs entités, celles-ci pouvant être des acteurs, le système ou ses sous-systèmes. Le diagramme ne montre donc que l'enchaînement séquentiel des différentes interactions.
Un diagramme de séquence est rattaché à un cas d'utilisation et décrit ce dernier en entier ou en partie, ce qui correspond à un scénario de fonctionnement possible, défini dans un cadre précis : il peut donc aboutir tout aussi bien à des évolutions positives (fonctionnement normal) ou négatives (gestion des problèmes), en particulier dans la phase de démarrage avant le fonctionnement normal.

Les éléments graphiques utilisés dans ce diagramme sont principalement :

- Des traits verticaux en pointillés appelés « lignes de vie » avec l'indication des propriétaires (en général des acteurs, le système et tout ou partie de ses sous-systèmes) sur la partie supérieure. Le temps se déroule du haut vers le bas, sans échelle particulière.
- Des flèches horizontales, avec différentes syntaxes, indiquant l'envoi et la réception de « messages », notion qui doit être prise au sens large car ils matérialisent les interactions qu'il peut y avoir entre un émetteur et un récepteur. Par exemple, l'appui sur un bouton peut être considéré comme le message envoyé (représentant dans ce cas un événement) et l'affichage d'une image sur un écran comme la réponse à cette sollicitation.

Conseils méthodologiques

S'il y a plusieurs scénarios possibles, plutôt que de créer un diagramme global dont la lecture serait difficile, il est conseillé de créer un diagramme par scénario.

Remarque

Ce diagramme permet de montrer les interactions entre les différentes parties non visibles dans un diagramme de cas d'utilisation présenté dans le paragraphe 3.4. page 37 qui n'indique que l'association entre l'acteur et un cas d'utilisation.

3.5.2. Illustration

Pour le système étudié, il est possible de mettre en place le diagramme présenté sur la figure 2.10 qui décrit l'évolution séquentielle des échanges entre les deux techniciens de vol (le pilote du drone et le

pilote de la nacelle supportant la caméra) lors de la phase de préparation au vol : cette description répond à une exigence portant sur la norme et ne représente donc pas un scénario complet car seule la phase de test préalable à l'utilisation du système est représentée, ce qui correspond à une partie des cas d'utilisation « Piloter le drone » et « Orienter la nacelle ».

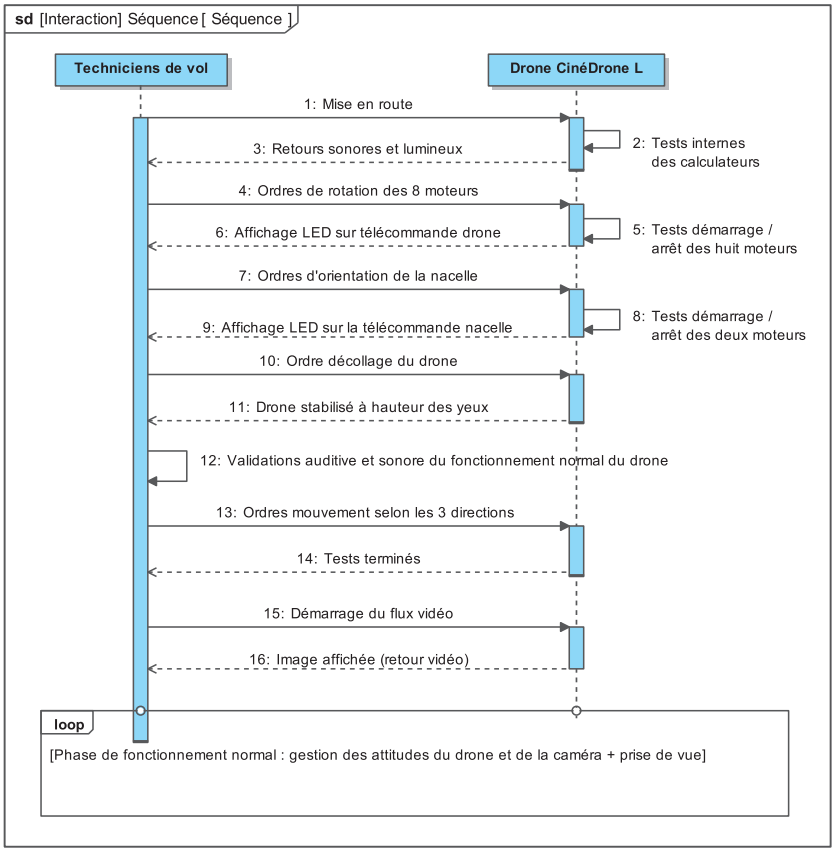


Figure 2.10. Un exemple de diagramme de séquence pour le drone de cinéma.

3.6. Diagramme de définition de blocs

3.6.1. Présentation

Définition 2.5. Diagramme de définition de blocs (bdd)

Le diagramme de définition de blocs est un diagramme structurel appelé *Block Definition Diagram* (identifiant **bdd**) dans le langage SysML.
L'objectif de ce diagramme est de décrire le système *via* des blocs (*blocks* dans le langage SysML) représentant des éléments matériels (cas le plus fréquent) mais également des entités abstraites (regroupement logique d'éléments) ou des logiciels.
Ce diagramme représente les caractéristiques principales de chaque bloc ainsi que les liens entre eux : il permet donc une modélisation de l'architecture du système.

Graphiquement, un bloc est représenté par un rectangle avec le stéréotype « *block* » comprenant un titre et des compartiments étagés regroupant des propriétés particulières : voir figure 2.11.
Il est ensuite possible de relier les blocs au moyen de liens dont la sémantique dépend de la nature particulière de la relation : en règle générale, le lien utilisé correspond à une composition, parfois à une agrégation (trait avec losange plein ou vide : voir tableau page 30), indiquant que l'élément supérieur possède un élément inférieur. Sur ces liens, il est possible de préciser la multiplicité d'un bloc en plaçant une valeur au bout du lien (voir exemple du drone).

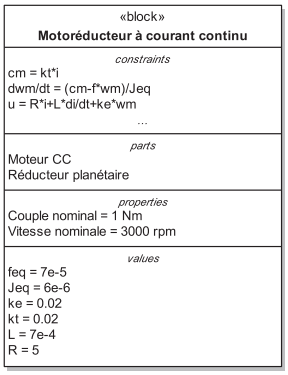


Figure 2.11. Constitution très complète d'un bloc dans le langage SysML.

Les blocs peuvent être caractérisés par des propriétés : il en existe de plusieurs sortes, comme indiqué sur la figure 2.11, mais les plus significatives sont :

- La propriété de type *value* permet d'exprimer une caractéristique quantifiable : pour un moteur par exemple, cela pourrait être son couple nominal, sa puissance maximale, etc.
- La propriété de type *part* permet de représenter ce qui compose le bloc. Elle est équivalente à un lien de composition (simple trait).

3.6.2. Illustration

Pour le système étudié, il est possible de mettre en place le diagramme de définition de blocs présenté figure 2.12 qui fait apparaître une hiérarchie de blocs, indiquant ce dont chaque bloc est composé, ainsi que les relations entre les blocs (voir tableau de la page 30) telles que :

- Relations de composition (trait avec losange plein) : le système de sustentation en vol est composé d'une batterie, d'un calculateur, d'un récepteur pour télécommande et de huit moteurs (le nombre à l'extrémité du lien indique la multiplicité). Cette relation sera celle la plus souvent rencontrée au niveau des CPGE.
- Relation d'agrégation (trait avec losange vide) : le système de sécurité est optionnel et non imposé par la norme rédigée par la DGAC.
- Relation d'association : le système de sustentation et la nacelle sont associés à un niveau d'importance équivalent (absence de flèche).

3.7. Diagramme de blocs internes

3.7.1. Présentation

Définition 2.6. Diagramme de blocs internes - ibd

Le diagramme de blocs internes est un diagramme structurel appelé *Internal Block Diagram* (identifiant **ibd**) dans le langage SysML.
Le diagramme de blocs internes est rattaché à un bloc issu du diagramme de définition de blocs présenté page 40, le cadre du diagramme représentant la frontière d'un bloc.
Le diagramme de définition de blocs introduit la notion fondamentale de « port » qui correspond à un point d'interaction avec l'extérieur du bloc.
Les connecteurs (traits) entre les ports indiquent soit les associations soit les flux de matière, d'énergie et d'information entre les différents blocs.

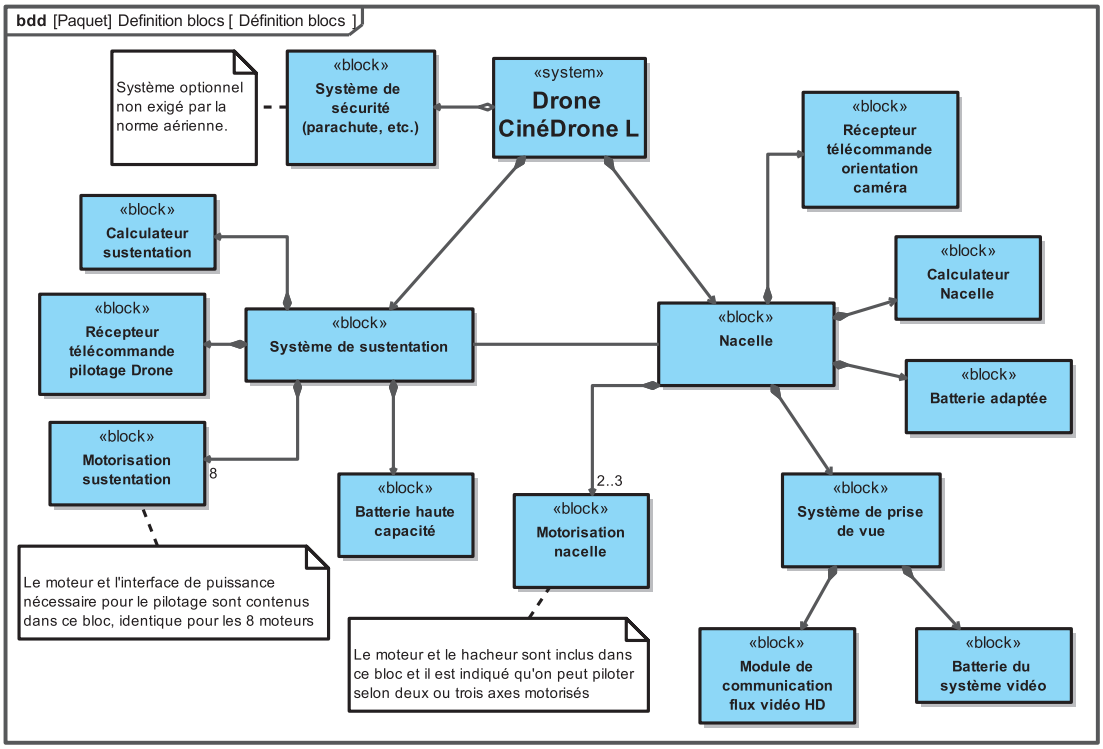


Figure 2.12. Un exemple de diagramme de définition de blocs pour le drone de cinéma.

La représentation graphique des ports est un carré placé sur le contour du bloc :

- Les *ports de flux* indiquent les échanges de matière, d'énergie et d'information entre blocs : ce type de port contient une flèche dont le sens (entrante, sortante ou bidirectionnelle) indique celui du flux.
- Les *ports standards* indiquent la logique de commande et les interfaces d'un bloc : ce type de port ne contient pas d'indication particulière.

Attention !

Ce diagramme montre les relations entre blocs de même niveau *via* les ports : il apporte donc des informations complémentaires à celles fournies dans le diagramme de définition de blocs, ce qui rend ces deux diagrammes complémentaires et non interchangeables.

Depuis la norme 1.4, en phase de large diffusion, cette notion de port a évolué légèrement, sans en changer globalement le principe :

- les ports *complets* (*Full Ports*) sont utilisés pour la connexion entre parties physiques du système (il s'agit donc d'une structure identique aux ports de flux ou standards précédents) ;
- les ports de *procuration* (*Proxy Ports*) précisent une interface avec une fonction du système ;
- les ports *encapsulés* (*Nested Ports*) permettent une combinaison de plusieurs ports dans un seul « macro-port » géré de manière globale.

3.7.2. Illustration

Pour la nacelle du système étudié, il est possible de mettre en place le diagramme de définition de blocs associé à la nacelle orientable et présenté sur la figure 2.13.

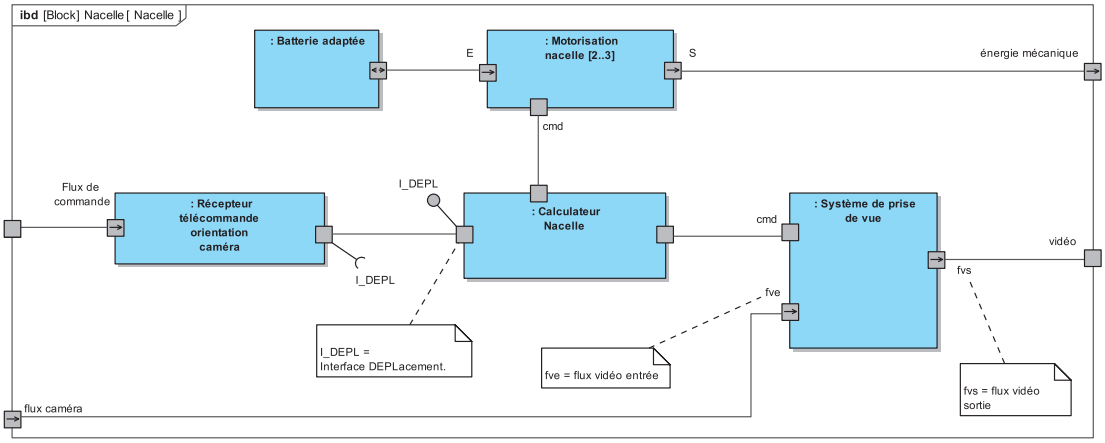


Figure 2.13. Un exemple de diagramme de blocs internes pour la nacelle du drone de cinéma.

Dans cet exemple de diagramme de définition de blocs associé à la nacelle, les seuls blocs présents sont ceux directement reliés au bloc « source » présentés sur le diagramme de définition de blocs de la figure 2.12, page 42.

Les interconnexions des différents blocs *via* les ports standard et de flux représentés sur un diagramme de blocs internes renseignent sur les relations entre les blocs : ainsi, par exemple, un port standard noté « cmd » a été utilisé pour commander le sous-système de prise de vue et deux ports de flux ont été ajoutés, le port entrant représentant le flux vidéo de la caméra (qui ne fait pas partie du système) et le port sortant représentant la communication avec la station au sol.

La présence d'un symbole à rotule entre les blocs « Récepteur » et « Calculateur » permet de montrer l'interface fournie par le calculateur pour piloter la nacelle (symbolisée par la sphère pleine) et l'interface requise par la télécommande (symbolisée par la sphère creuse).

3.8. Diagrammes d'états et d'activités

3.8.1. Introduction

Ces deux diagrammes sont traités dans la même partie car ils ont des liens forts. Pourtant, même s'ils possèdent une syntaxe proche, leurs domaines d'utilisation sont sensiblement différents et il est donc fondamental de ne pas les confondre.

Attention !

Le diagramme d'activités n'est pas explicitement au programme des CPGE.
Son utilisation est cependant pratique pour décrire la structure des programmes implantés dans les micro-contrôleurs, dont l'étude est au programme : c'est la raison pour laquelle ce diagramme est présenté dans le cadre de cet ouvrage.

3.8.2. Le diagramme d'états

Définition 2.7. Diagramme d'états - stm

Le diagramme d'états est un diagramme comportemental appelé *State Machine Diagram* (identifiant **stm**) dans le langage SysML.
Le diagramme d'états est rattaché à un bloc qui peut être le système, un sous-système ou un composant. Le comportement décrit par ce type de diagramme sert à montrer les différents états pris par le bloc en fonction des événements qui lui arrivent.
Un état représente une situation d'une durée finie durant laquelle un système exécute une activité, satisfait à une certaine condition ou bien est en attente d'un événement. Le passage d'un état à un autre se fait en franchissant une transition.

Les éléments graphiques utilisés dans ce diagramme sont principalement des rectangles aux coins arrondis. Les transitions entre les états sont représentées par des flèches orientées d'un état à un autre. La transition est franchie lors d'une occurrence de l'événement rattaché à la transition.

Dans ce diagramme, il est possible de rajouter des événements internes qui permettent de montrer la réponse à un événement mais sans changer d'état. Les événements *entry*, *do* et *exit* sont classiques et indiquent ce qu'il se passe à l'entrée dans l'état (mot-clé *entry*), pendant l'état (mot-clé *do*) et à la sortie de l'état (mot-clé *exit*).

Conseils méthodologiques

Chaque bloc d'un diagramme de définition de blocs ne conduit pas nécessairement à un diagramme d'états car cela n'est pas toujours possible.

3.8.3. Le diagramme d'activités

Définition 2.8. Diagramme d'activités - act

Le diagramme d'activités est un diagramme comportemental appelé *Activity Diagram* (identifiant **act**) dans le langage SysML.
Ce diagramme permet de représenter le déroulement d'un processus sous la forme d'une activité correspondant à une décomposition séquentielle d'actions, aussi appelées tâches.
Dans sa forme la plus restreinte, ce diagramme représente un algorigramme, c'est-à-dire un flux de contrôle (ce flux n'a rien à voir avec ceux présents dans le diagramme de blocs internes : il ne faut donc pas les confondre).

Les éléments graphiques utilisés dans ce diagramme ressemblent fortement à ceux du diagramme d'états : chaque tâche est représentée par un rectangle aux coins arrondis et est ensuite reliée à une autre tâche par des transitions représentées par de simples flèches (voir exemple). Lorsqu'une tâche est terminée, la suivante commence.

3.8.4. Illustration

Le diagramme d'états

Pour le système étudié, il est possible de mettre en place le diagramme d'états (identifiant **stm**) présenté sur la figure 2.14 qui permet de représenter les différents modes de fonctionnement du système, ainsi que les événements qui permettent de passer de l'un à l'autre.

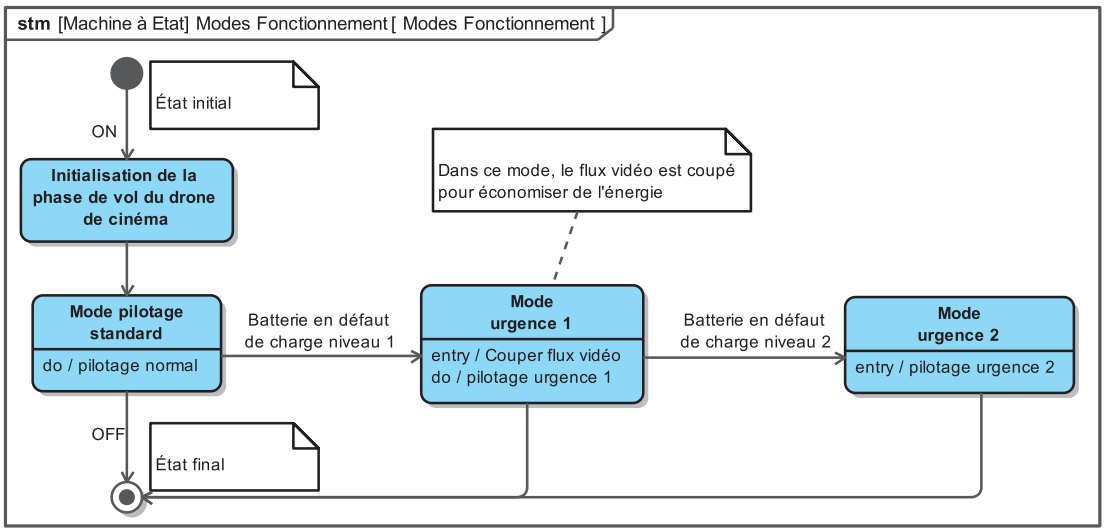


Figure 2.14. Un exemple de diagramme d'états pour le drone de cinéma.

L'initialisation est un mode temporaire dans lequel le système effectue tous ses tests avant de donner la main aux pilotes. La transition est ensuite franchie automatiquement car il n'y a pas d'événement particulier à marquer (c'est la fin de l'initialisation). Viennent ensuite le mode de fonctionnement normal puis les modes d'urgence en cas de niveau de charge insuffisant. Le comportement lors de l'entrée et de la sortie de chaque état est décrit : par exemple, l'entrée dans le mode urgence 1 (événement interne *entry*) provoque automatiquement la coupure du flux vidéo pour économiser l'énergie et un mode de pilotage dégradé (pilotage à vue) où le drone ne peut que redescendre.

Le diagramme d'activités

Dans le diagramme d'états de la figure 2.14, ce qui se passe dans chaque mode n'est pas décrit : pour ce faire, il est possible de choisir des nouvelles machines d'états ou des activités.

Par choix, la description de chaque mode a été réalisée uniquement par des diagrammes d'activités (identifiant **act**), comme sur la figure 2.15 où il apparaît que :

- Le mode normal correspond à un pilotage complet du drone avec en parallèle l'envoi du flux vidéo (les deux barres noires marquent le début et la fin de l'envoi des deux flux).
- Dans le mode d'urgence 2, la procédure automatique pour atterrir est décrite, la note précisant les limitations de ce mode.

3.8.5. Synthèse sur le positionnement relatif de ces deux diagrammes

Les deux types de diagrammes sont différents car :

- Le diagramme d'états montre les événements déclenchant le passage d'un mode à un autre et il y aura quasiment toujours un événement associé à une transition [12].
- Le diagramme d'activités ne possède aucun événement associé aux transitions entre actions : la fin d'une action implique automatiquement le passage à la suivante, donc dans un ordre déterminé

[12] Il existe des contre-exemples : à la fin d'une phase d'initialisation, il est par exemple inutile de mettre une indication du type 'Fin d'initialisation' car c'est implicite.

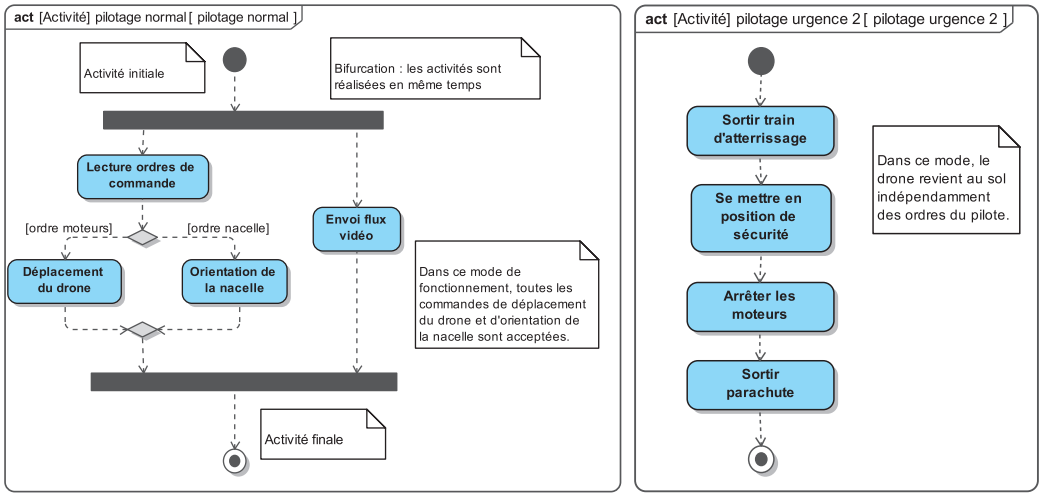


Figure 2.15. Exemples de diagrammes d'activités du drone de cinéma.

d'actions menant à un résultat. Lorsque le processus est enclenché, il va à son terme selon un ordre précis.

En conclusion :

- Le diagramme d'états ne se rattache qu'à un bloc, alors que le diagramme d'activités peut être supporté par plusieurs blocs.
- Un pilotage par des événements se traduit par un diagramme d'états : il ne doit donc pas devenir un diagramme d'activités.
- Dans un processus décrit par un diagramme d'activités, il est possible de mettre en évidence l'élément associé à la tâche. Avec le diagramme d'états, la question ne se pose pas car il est associé à un seul bloc.

3.9. Diagramme paramétrique

3.9.1. Présentation

Définition 2.9. Diagramme paramétrique - par

Le diagramme paramétrique est un diagramme structurel appelé *Parametric Diagram* (identifiant **par**) dans le langage SysML. Ce diagramme est une extension du diagramme de définition de blocs (identifiant **ibd** présenté page 41) et il partage donc les mêmes éléments graphiques.
Il présente la particularité de pouvoir connecter entre elles des contraintes ajoutées au diagramme de blocs par le biais d'un bloc particulier, dit « de contraintes » (*constraint block*) qui contient des paramètres et une relation, en général mathématique, les reliant.

Pour information, les contraintes sont le plus souvent définies par un langage OCL (*Objet Contrainst Language*) issu d'UML : ce langage formel est simple d'accès et représente un juste milieu entre un langage naturel et un langage mathématique.

Conseils méthodologiques

Les paramètres de la relation peuvent faire référence à des éléments du système, par exemple des propriétés de blocs : des processus physiques peuvent ainsi être modélisés dans un diagramme paramétrique, qui peut alors être directement utilisé pour une simulation de tout ou partie de l'évolution du système.

3.9.2. Illustration

L'exemple de la figure 2.16 est tiré du tutoriel SysML de l'OMG.

Comme dit précédemment, il faut d'abord poser des blocs de contraintes dans un diagramme de blocs avant de pouvoir construire un diagramme paramétrique. Dans la figure 2.16, plusieurs équations liant des paramètres entre eux apparaissent, telles que l'équation **e2** liant l'accélération à la force et la masse du véhicule {f = m*a }. L'outil de calcul permettra par la suite d'exploiter l'ensemble afin d'en tirer des résultats.

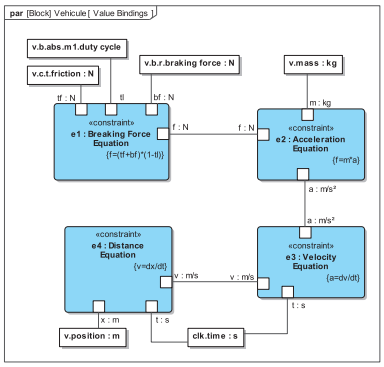


Figure 2.16. Un exemple de diagramme paramétrique (source : tutoriel SysML de l'OMG).

3.10. Remarque

La mise en place d'un diagramme paramétrique par les logiciels permettant le développement de modèles SysML est, dans l'état actuel de leur développement, un processus long et relativement complexe. Dans le cadre du programme de CPGE où le langage SysML est proposé uniquement à la lecture, il est donc, pour le moment, pertinent d'utiliser d'autres logiciels pour simuler des modélisations multi-physiques tels que, par exemple :

- La suite Matlab + Simulink + Simscape + SimMechanism (propriétaire).
- La suite Maple + MapleSim (propriétaire).
- La suite Scilab + Xcos + Coselica + Simm (libre).
- Le logiciel dédié OpenModelica (libre).

3.11. Diagramme de paquetages

Définition 2.10. Diagramme de paquetages - pkg

Le diagramme de paquetages est un diagramme structurel appelé *Package Diagram* (identifiant **pkg**) dans le langage SysML.
L'objectif de ce diagramme est de créer des « macros-associations » de concepts ou d'éléments afin de relier différentes parties d'un même modèle.
Ce diagramme n'apparaît pas explicitement dans le programme des CPGE : il est donc **donné uniquement à titre d'information**.

Un paquetage [13] est un élément « conteneur » permettant le regroupement de tout élément graphique tel que, par exemple, des blocs, des interfaces, des acteurs ou des cas d'utilisation qui ne seraient par ailleurs pas forcément associés dans les autres diagrammes. Ce diagramme permet donc de réunir des éléments de modèles de types très différents dans une seule zone et cette réunion peut se faire selon plusieurs points de vue : par hiérarchie d'échange sur le système (entreprise, sous-traitant, système, élément du système, etc.), par domaine ou type de diagramme (comportement, structure ou exigences) ou par toute autre structure permettant d'améliorer une vision particulière du modèle étudié.

Les paquetages ainsi créés peuvent, comme pour les blocs, avoir des relations de contenance ou de dépendance. De plus, comme pour les autres concepts du langage SysML, les paquetages peuvent être imbriqués les uns dans les autres. Ce diagramme permet donc, d'une manière connexe aux diagrammes de définition de blocs (identifiant **bdd** : voir page 40) ou de blocs internes (identifiant **ibd** : voir page 41), une modélisation de l'architecture du système.

La notion de paquetage est particulièrement intéressante dans le cas de modélisations très complexes et transversales, par exemple pour un avion : dans ce type de systèmes, la diversité des solutions technologies devant agir de manière simultanée rend la modélisation très complexe et cette notion de paquetage permet de réunir des informations très diverses pouvant être utiles pour un cas donné ou pour un acteur particulier (par exemple un sous-traitant) auquel il peut être problématique de transmettre tout le modèle.

Attention !

Dans le cadre de la lecture de modèles simples, cette notion n'apporte pas d'éléments nouveaux et elle ne devrait donc pas être proposée dans le cadre des CPGE.

3.12. Le diagramme de contexte

Définition 2.11. Diagramme de contexte (pas d'identifiant)

Le diagramme de contexte est une extension *non normalisée* du langage SysML qui permet de définir les frontières de l'étude et la phase du cycle de vie dans laquelle on situe l'étude (il s'agit généralement de la phase d'utilisation normale du système).
Ce diagramme permet de préciser, si possible de manière exhaustive, les acteurs et éléments environnants au système étudié. Il permet également de faire apparaître les différents acteurs ou éléments intervenant dans une exigence.

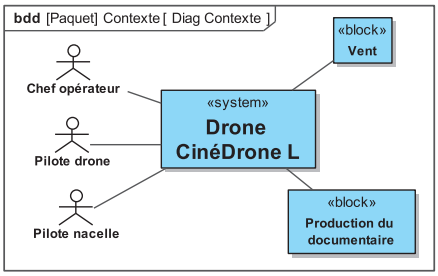


Figure 2.17. Un exemple de diagramme de contexte pour le drone de cinéma.

De par sa position d'extension, il n'y a absolument *aucune recommandation spécifique sur la manière dont ce diagramme sera établi* : classiquement, on utilise un diagramme de définition de blocs (voir page 40) mais il pourrait également être mis en œuvre par une carte mentale (ou carte heuristique permettant de représenter graphiquement les liens qui existent entre un concept et les informations qui lui sont associées), un diagramme de blocs internes (voir page 41), etc.

Pour le système étudié, il est possible de mettre en place le diagramme présenté sur la figure 3.12. où apparaissent différents acteurs qui interagissent avec le drone de cinéma.

[13] Le terme anglais *package* est également employé par certains auteurs.

La collection incontournable pour réussir et faire la différence en CPGE scientifiques

Pour chaque chapitre :

Tout le cours

Pour maîtriser l'intégralité des notions du programme, accompagnées d'exemples et de démonstrations

Méthode et conseils

Pour acquérir les techniques et astuces de résolution des exercices avec des rubriques claires

Fiches de synthèse

Pour retenir l'essentiel du cours avant les écrits, les oraux et dans la perspective des concours

Entraînement intensif

Pour vous préparer efficacement et vous tester avec + de 280 exercices de difficulté progressive : vrai/faux, application, approfondissement et problèmes types concours

Tous les corrigés détaillés

Pour comprendre chaque étape de résolution et acquérir les bons réflexes

+ EN LIGNE

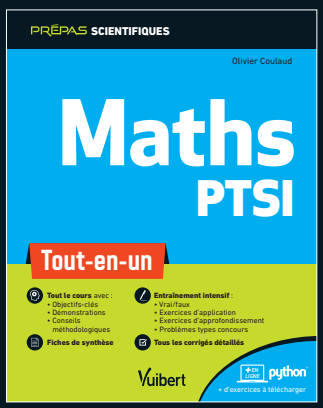
+ de TP complémentaires à télécharger

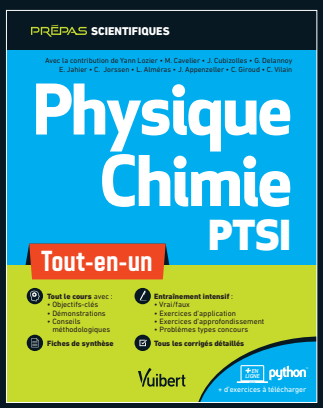
Tous les auteurs sont professeurs en classes préparatoires scientifiques et membres de jurys de concours.

ISBN : 978-2-311-40631-3

PRÉPAS SCIENTIFIQUES

UNE COLLECTION COMPLÈTE





www.Vuibert.fr