

Architecture générale et logicielle d'un système embarqué

Animé par Sylvain Labasse

INTRO



EN FIN DE MODULE, VOUS SAUREZ...

Aborder méthodiquement un projet d'ingénierie des systèmes

Modéliser ses interactions avec son environnement

Formaliser les exigences du système

Enumérer les composants répondant aux exigences

Spécifier les composants matériels et logiciels en SysML

Passer d'un diagramme de blocs au code

PRE-REQUIS

Public

I5

Nécessaire

EMBE710 - Systèmes Embarqués

SUPPORT

Copie des slides

MyLearningBox

Notes de cours

Réalisation des ateliers

L'ENVIRONNEMENT

Matériel

PC sous Windows ou Linux

Logiciel

draw.io

Eclipse EMF / Papyrus

EVALUATION

Ateliers

A remettre sur MyLearningBox en fin de séance

QCM (individuel)

CONTENU

Vue d'ensemble

Ingénierie des systèmes

SysML

Diagrammes

Diagrammes d'étude

Cas d'utilisation

Activité

Exigences

Blocs (ébauche)

CONTENU (SUITE)

Définition de blocs

- Bloc, Structure / Propriétés
- Comportement
- Accessoires

Vision interne des blocs

- Diagramme de bloc interne
- Du BDD à l'IBD
- Parties/Références
- Connecteurs

CONTENU (SUITE ET FIN)

Dynamique et spécifications

- Activité de blocs

- Séquence et état

- Paramétrique

- Allocations

COURS CONNEXE

RDYE912 - Conception d'une solution embarquée en temps réel

INGENIERIE DES SYSTEMES

OBJECTIFS

Caractéristiques d'un système complexe

Ingénierie des systèmes vs Ingénierie logicielle

Intérêt de SysML

VUE D'ENSEMBLE

→ Ingénierie des systèmes

SysML

Diagrammes

Synthèse

SYSTEME COMPLEXE

Éléments en interaction

Frontière, Interaction avec l'environnement

Entités qui le composent en interactions

Interactions non linéaires (Théorie du chaos)

Approche cartésienne inadéquate

Pas la somme des parties mais le produit des interactions

Cybernétique Wiener (48), Maturana/Varela (80)

Systèmes complexes : Simulation

SYSTEMES EMBARQUES

Caractéristiques

Sous-système informatique

Partie intégrante d'un système électrique/mécanique

88% des processeurs produits

Souvent temps réel

Contraintes

Consommation/Autonomie

Coût

Fiabilité/Résistance

Taille

INGENIERIE DES SYSTEMES

Technologies variées

Mécanique, électrique, électronique, matériels informatiques, logiciels, réseaux de communication, ...

Cadre

Secteurs : Aérospatial, transport, énergie, médical, ...

INCOSE (1991) : International

AFIS (1968) : France

ISO 15288

VUE D'ENSEMBLE

✓ Systèmes embarqués

→ SysML

Diagrammes

Synthèse

SysML

Objectif

Formalise l'architecture et les exigences d'un système informatique souvent embarqué.

Versions

FORMAL Version	Release date	URL
1.5	May 2017	http://www.omg.org/spec/SysML/1.5/
1.4	September 2015	http://www.omg.org/spec/SysML/1.4/
1.3	June 2012	http://www.omg.org/spec/SysML/1.3/
1.2	June 2010	http://www.omg.org/spec/SysML/1.2/
1.1	November 2008	http://www.omg.org/spec/SysML/1.1/
1.0	September 2007	http://www.omg.org/spec/SysML/1.0/

SysML

Niveaux

Opérationnel

Système

Composant

Catégories

Structurel : 4 diagrammes

Comportemental : 4 diagrammes

Transversal : 1 diagramme

VUE D'ENSEMBLE

✓ Systèmes embarqués

✓ SysML

➔ Diagrammes

Synthèse

DIAGRAMMES

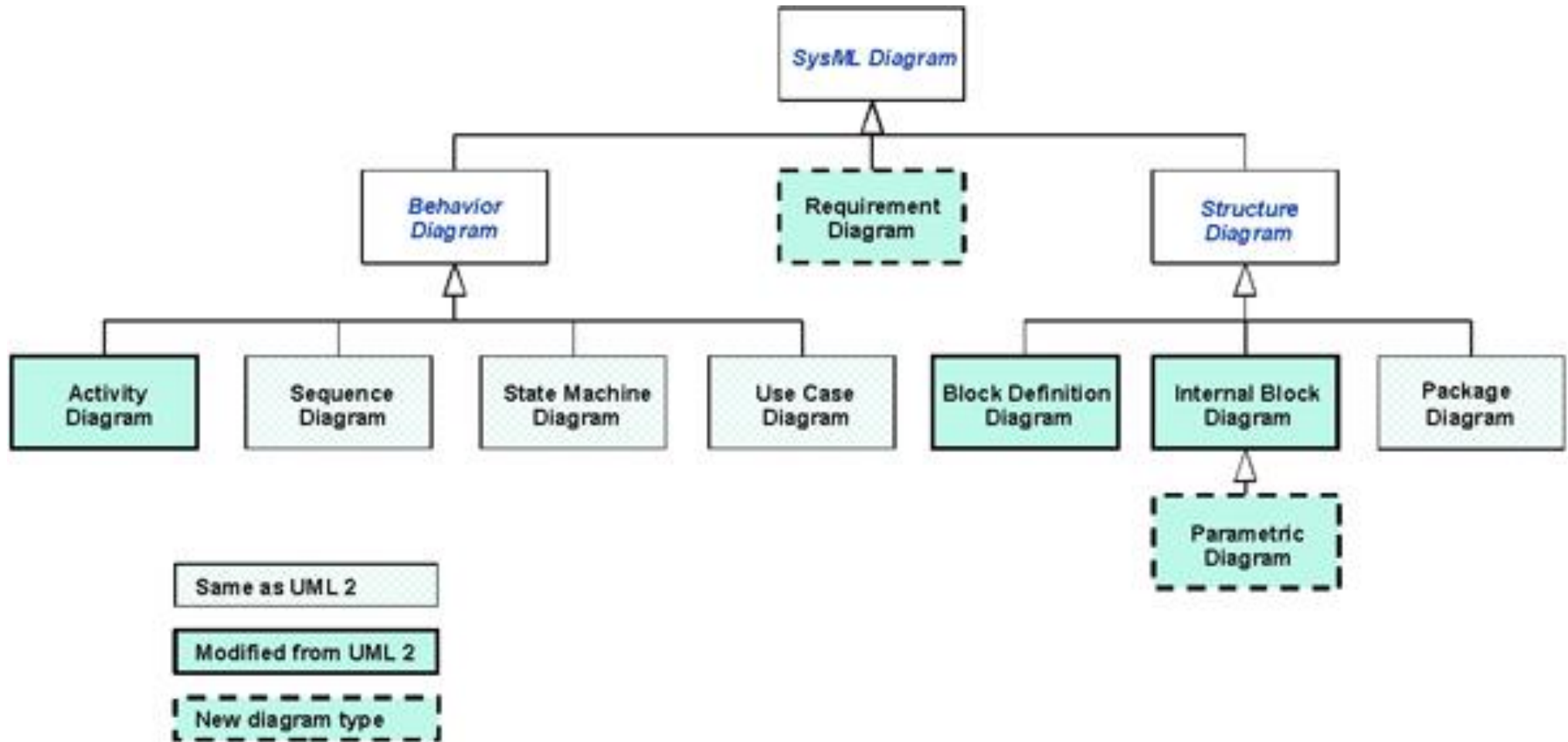


Figure 1 - Diagrammes de SysML (src: omgsysml.org)

DIAGRAMMES

Structurels

- Définition de bloc

- Bloc interne

- Paramétrique

- Paquets

Comportemental

- Activité, Etat

- Séquence

- Cas d'utilisation

VUE D'ENSEMBLE

- ✓ Systèmes embarqués
- ✓ SysML
- ✓ Diagrammes
- ➔ Synthèse

REVISION

- ⌚ Sont des systèmes embarqués : ...
- ⌚ Pour chacun spécifier les contraintes : Coût, autonomie, fiabilité, dimension
- ⌚ Citer deux diagrammes spécifiques à SysML

RESUME

Caractéristiques d'un système complexe

Ingénierie des systèmes vs Ingénierie logicielle

Intérêt de SysML

DIAGRAMMES D'ETUDE

OBJECTIFS

Approche structurée d'un système embarqué

Enumération des interactions

Recensement des exigences

Premier découpage du système

DIAGRAMMES D'ETUDE

→ Cas d'utilisation

Activité

Exigences

Bloc (ébauche)

Résumé

CAS D'UTILISATION

Objectif

Enumérer les interactions du système et de son environnement.

Acteurs

Primaires/secondaires

Use case

Action de « bout en bout »

Associations

Pas de flèche, cardinalité optionnelle

CAS D'UTILISATION (EXEMPLE)

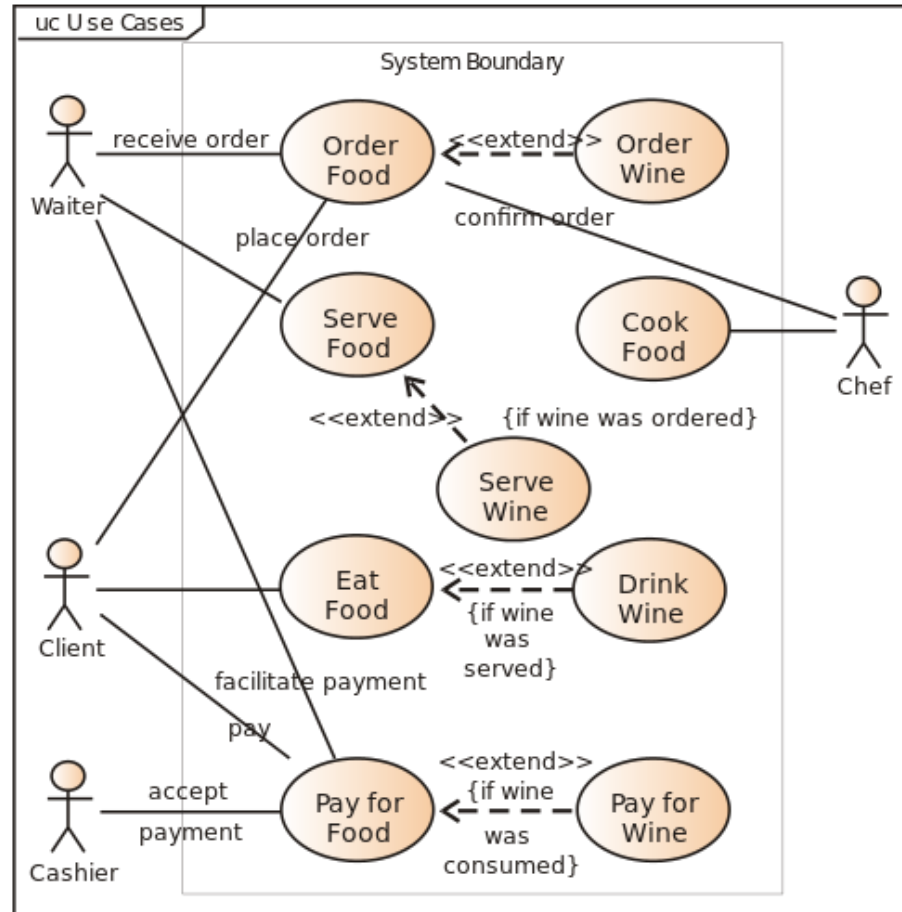


Figure 2 - Exemple de diagramme de cas d'utilisation (src: wikipedia)

DIAGRAMMES D'ETUDE

✓ Cas d'utilisation

→ Activité

Exigences

Bloc (ébauche)

Résumé

DIAGRAMME D'ACTIVITE

Objectifs

Décrire un processus, flux d'actions et leurs conditions

Éléments

Noeud initial/final

Activités reliées par des Flux

Décision

Barre de synchronisation

Optionnel

Couloirs, Objet/état

DIAGRAMME D'ACTIVITE (EXEMPLE)

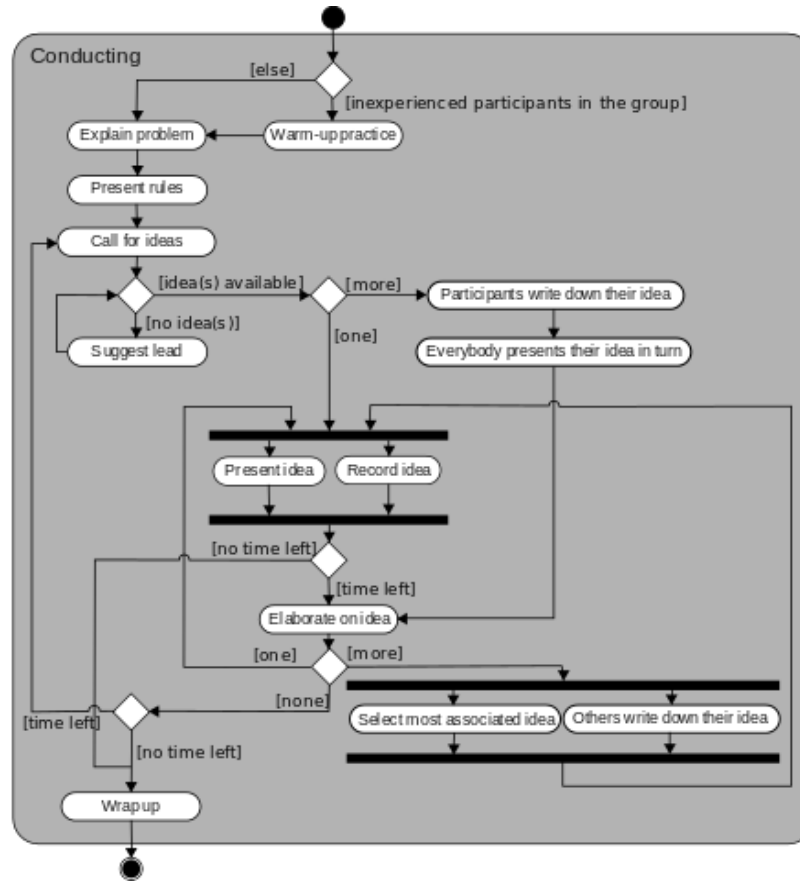


Figure 3 - Diagramme d'activité (src: wikipedia)

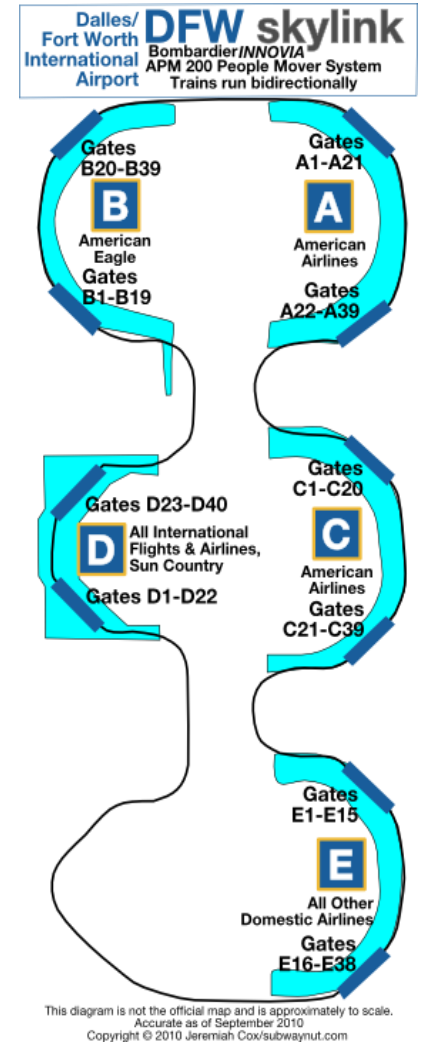
ATELIER 1 – ETUDE

Sujet

Etude d'un VAL/APM qui transporte des passagers en transit entre les terminaux d'un aéroport. Les 2 rames interagissent avec les usagers, un système de porte palière (ou PSD), le canal d'info. des passagers (horaires de vol), le service de supervision et de maintenance.

Compétence

Aborder méthodiquement un projet d'ingénierie de sys.
Modéliser ses interactions avec son environnement



DIAGRAMMES D'ETUDE

✓ Cas d'utilisation

✓ Activité

→ Exigences

Blocs (ébauche)

Résumé

DIAGRAMME D'EXIGENCES

Rôle

Exigences qualitatives (textuelles)

Traçabilité des exigences

Peut être sous forme de matrice

Relations

« requirement » contains \oplus « requirement »

« requirement » deriveReq « requirement »

« block » satisfy « requirement »

« useCase » refine « requirement »

« testCase » verify « requirement »

DIAGRAMME D'EXIGENCES — EXEMPLE

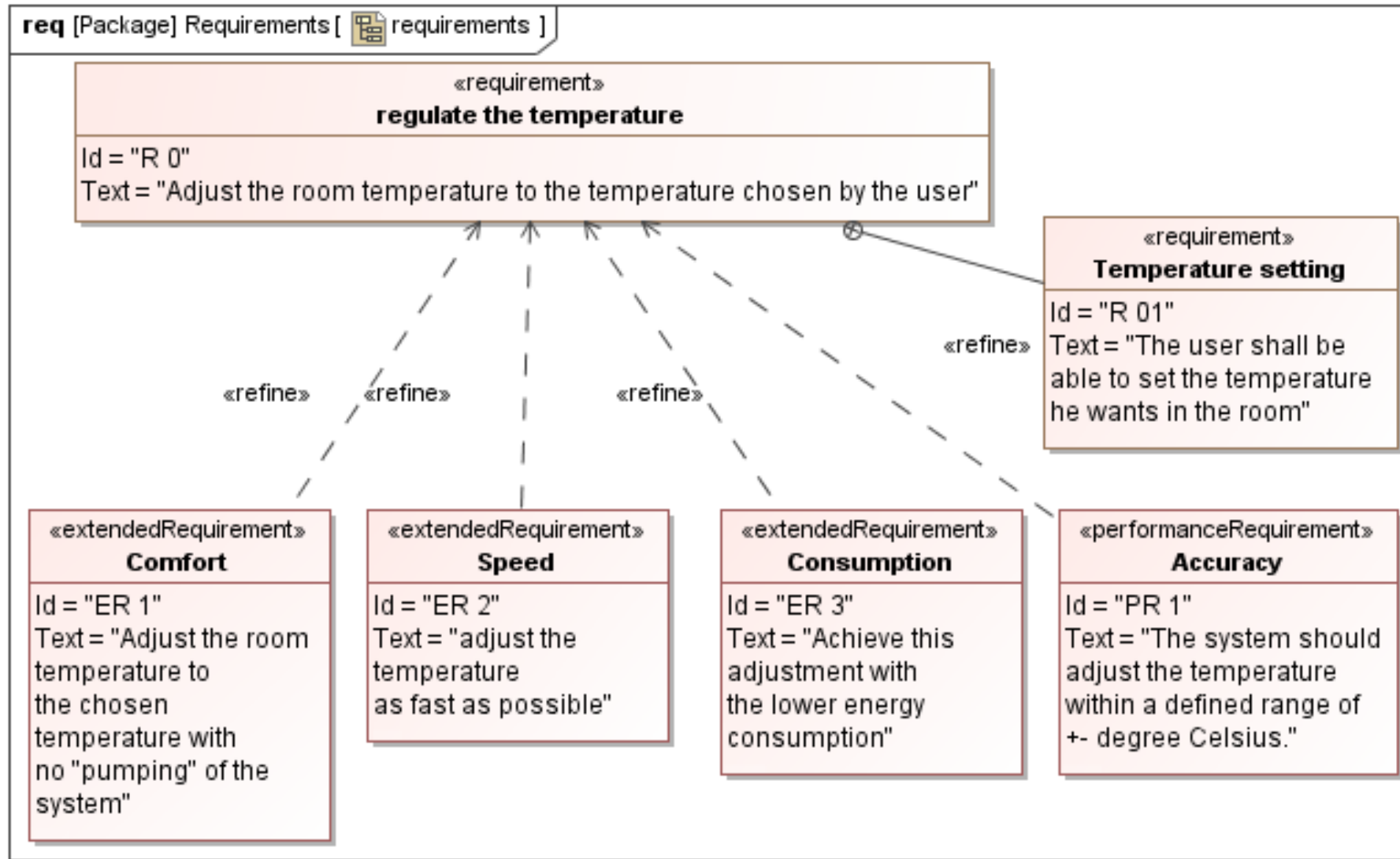


Figure 4 - Diagramme d'exigence d'un thermostat (src: <https://docs.nomagic.com>)

DIAGRAMMES D'ETUDE

✓ Cas d'utilisation

✓ Activité

✓ Exigences

➔ Blocs (ébauche)

Résumé

BLOCS (EBAUCHE)

Objectif

Avoir une vue synthétique des grandes parties du système.

Éléments

Bloc

Valeurs

Associations

BLOCS (EBAUCHE)

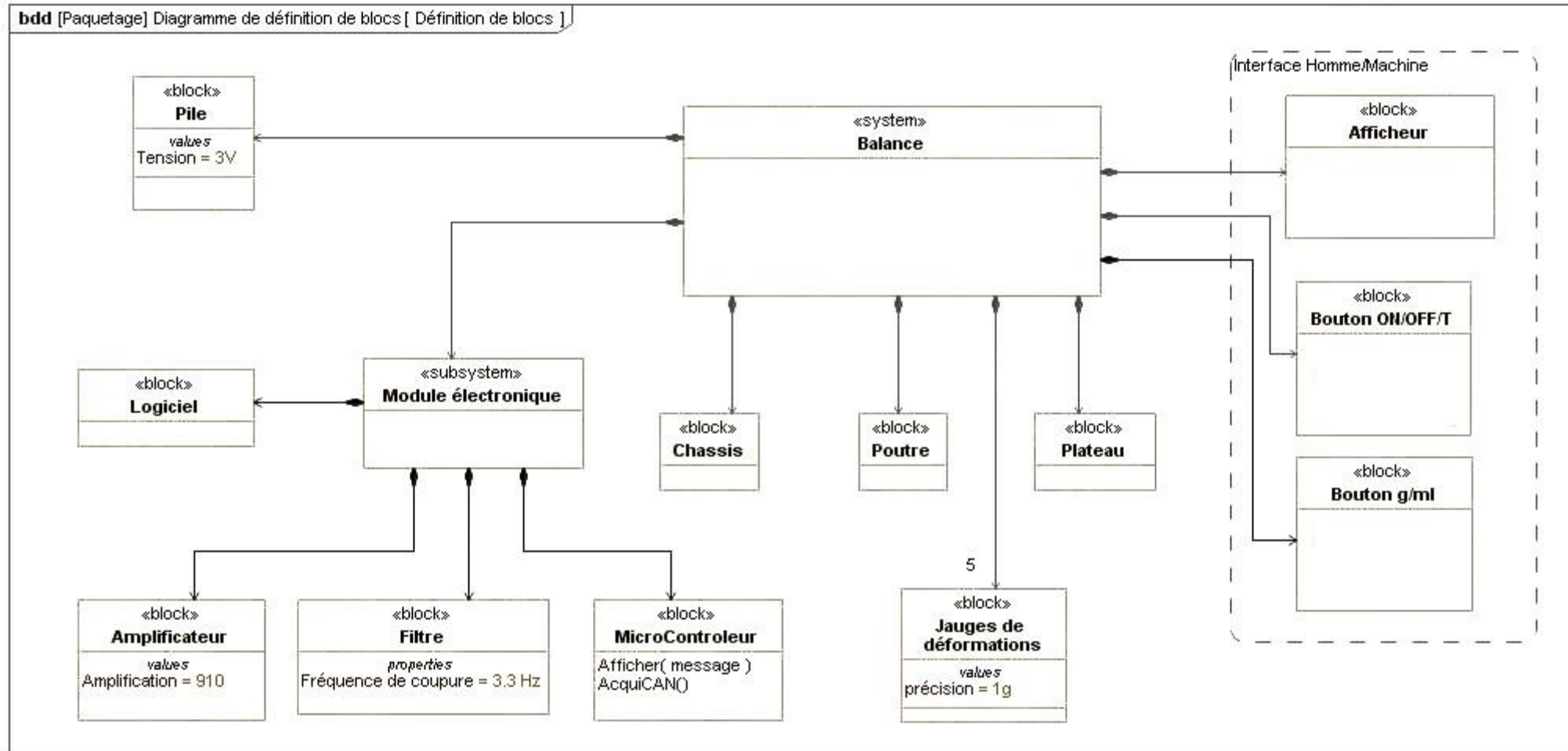


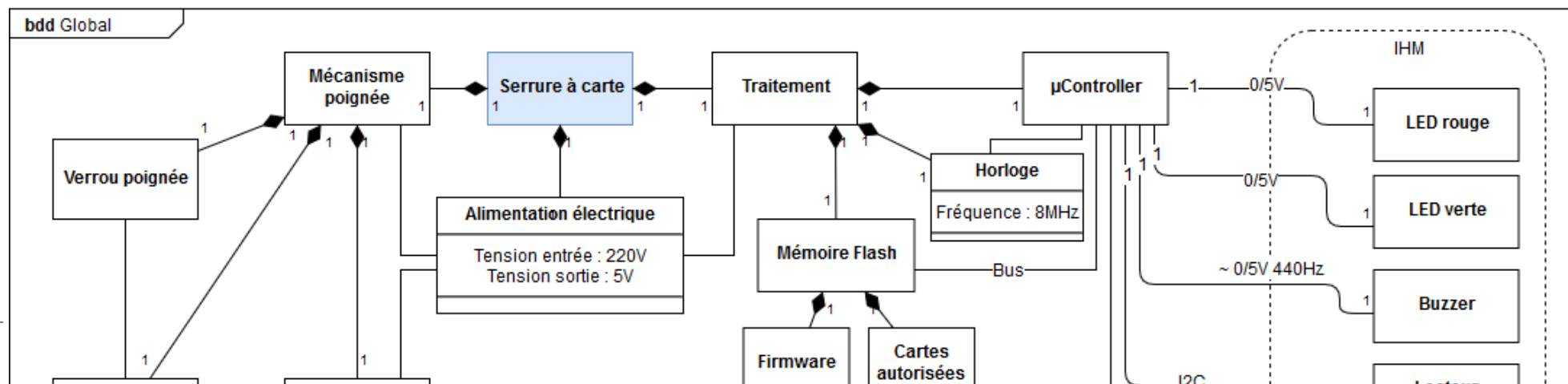
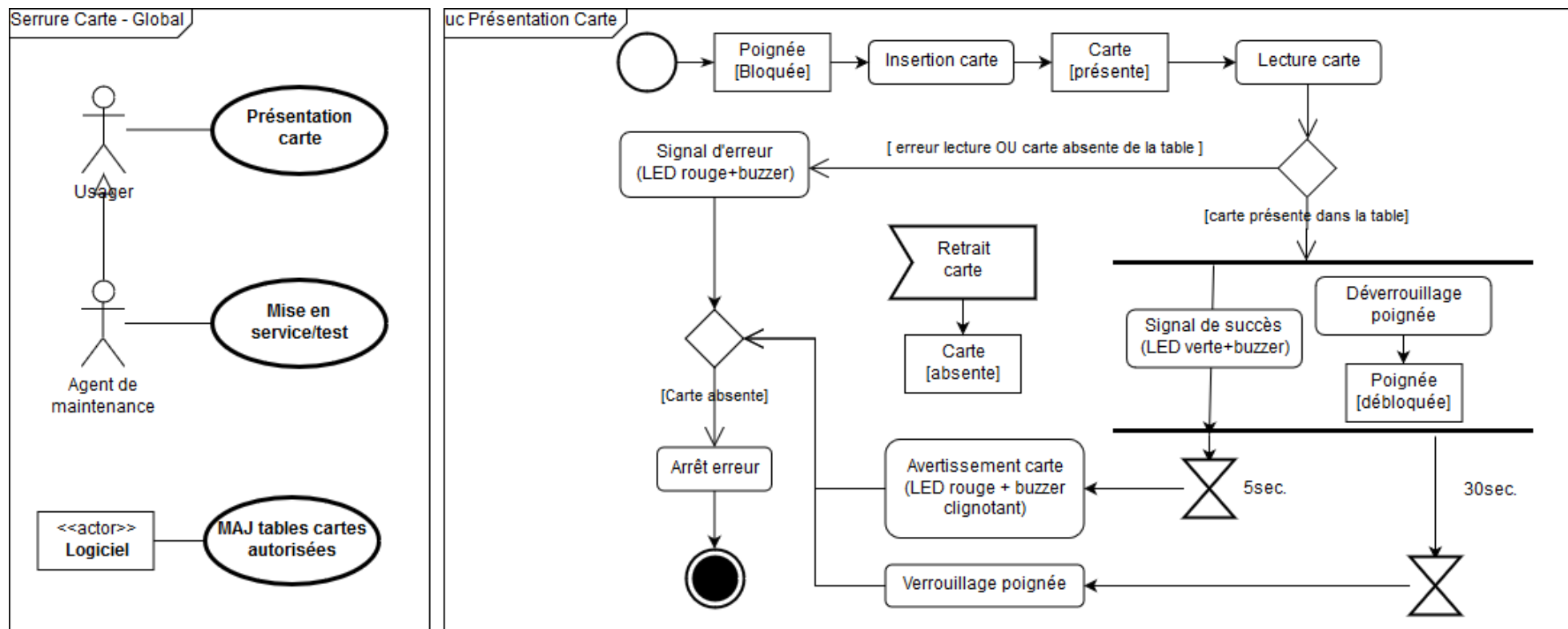
Figure 5 - Diagramme de bloc (src: wikimeca)

DIAGRAMMES D'ETUDE

- ✓ Cas d'utilisation
- ✓ Activité
- ✓ Exigences
- ✓ Blocs (ébauche)
- ➔ Résumé

REVISION

- ⌚ Cas d'utilisation d'une serrure à carte ?
- ⌚ Diagramme d'activité de chaque cas d'utilisation de la serrure à carte ?
- ⌚ Exigences de la serrure à carte ?
- ⌚ Diagramme de bloc de la serrure à carte ?



ATELIER 2 – EXIGENCES

Sujet

À partir de la correction de l'Atelier 1 fournie, et en prenant en compte de nouvelles contraintes :

- Proposer une arborescence d'exigences pour le déplacement entre 2 arrêts
- Ebaucher un diagramme de bloc
- Transformer l'arborescence en matrice

Compétences

Formaliser les exigences du système

Enumérer les composants répondant aux exigences

RESUME

Approche structurée d'un système embarqué

Enumération des interactions

Recensement des exigences

Premier découpage du système

DEFINITION DE BLOCS

OBJECTIFS

Liste exhaustive des blocs d'un système

Description complète de chaque bloc

Connexions et interfaces entre blocs

Vision des moyens d'interaction

Figure 6 - SysML Distilled - Lenny Delligatti - Addison Wesley

DEFINITION DES BLOCS

→ Bloc

Structure

Comportement

Accessoires

Résumé

BLOC

Représentation

Modèle \neq instance

Rectangle : « bloc » + nom du bloc

Compartiments

Structurels : Parties, Valeurs, Contraintes, Ports

Comportementaux : Opérations, Signaux/Réceptions

Associations

Composition \blacklozenge/\lozenge , association — , dépendance $- - -$

Généralisation/Spécialisation $\text{—}\triangleright$

DEFINITION DES BLOCS

✓ Bloc

→ Structure / Propriétés

Comportement

Accessoires

Résumé

PARTIES (PARTS)

Rôle

Partie/composant (non partagé-e) du bloc

Syntaxe

<nom> : <type> [<cardinalité>]

Remarque

Alternative : Composition

REFERENCES (REFERENCE)

Rôle

Référence à un composant dont dépend le bloc

Syntaxe

<nom> : <type> [<cardinalité>]

Remarque

Alternative : Association

VALEURS (VALUES)

Rôle

Caractéristiques scalaires d'un bloc

Syntaxe

<nom> : <type> [<cardinalité>] = <valeur par défaut>

Remarque

Valeur calculée (aka dérivée) est précédée de /

CONTRAINTES (CONSTRAINTS)

Rôle

Domaine de définition d'une ou plusieurs valeurs

Syntaxes

{ assertion représentant la contrainte }



<nom> : <type>

Remarque

<type> : type de contrainte, ailleurs dans le diagramme

PORTS

Représentation

 ou  sur la bordure du bloc

Standard

Service offert/requis par le bloc

Cercle/demi-cercle par interface resp. proposée/requise

Flux (flow)

<nom> : <type> (entrée) ou <nom> : ~<type> (sortie)

<> dans le carré = flux « non-atomiques »

Flux non-atomiques nécessite 1 bloc « flowSpecification »

DEFINITION DES BLOCS

- ✓ Bloc
- ✓ Structure
- Comportement
- Accessoires
- Résumé

COMPORTEMENT

Opérations

Liste des services « appelables » du bloc (méthodes)

<operation> (<paramètres>) : <retour> [card.]

<in|out|intout> <paramètre> : <type> [card.] = <def>

Signaux/Réceptions (signal)

Liste des stimuli du bloc (événements)

« signal » <operation> (<paramètres>)

<paramètre> : <type> [card.] = <def>

DEFINITION DES BLOCS

- ✓ Bloc
- ✓ Structure
- ✓ Comportement
- ➔ Accessoires

Résumé

ACCESSOIRES

Dépendances

- Flèche pointillé

- Complémentaire des associations

Acteurs

- Inspiré UseCase

- Rarement utile

Types de valeur

- Bloc définissant un type complexe de valeur

- Héritage possible, ...

BLOCS DE CONTRAINTES

Représentation

Bloc « constraint »

Parties : « constraints » et « parameters »

Contraintes

Assertions entre accolades

Paramètres

Liste et type des littéraux des contraintes

DEFINITION DES BLOCS

- ✓ Bloc
- ✓ Structure
- ✓ Comportement
- ✓ Accessoires
- ➔ Résumé

REVISION

 BDD d'une serrure à carte ?

 1 ou 2 contraintes ?

 Opérations et signaux ?

ATELIER 3 – DIAGRAMME DE BLOCS

Sujet

L'atelier consiste à compléter le diagramme de définition des blocs fourni du robot de purification de l'air Samsung Bot Air présenté au CES 2019.

Il est autorisé d'ajouter des blocs si nécessaire.

Compétences

Spécifier les composants matériels et logiciels en SysML.

RESUME

Liste exhaustive des blocs d'un système

Description complète de chaque bloc

Connexions et interfaces entre blocs

Vision des moyens d'interaction

VISION INTERNE

OBJECTIFS

Description interne de blocs

Conception à partir de la définition du bloc

Visualisation des composants et interfaces

Interconnexions et flux entre parties

VISION INTERNE

➔ Diagramme de bloc interne

Du BDD à l'IBD

Parties/Références

Connecteurs

Résumé

DIAGRAMME DE BLOC INTERNE

Objectif

- Détail d'un bloc
- Structure interne

Composants

- ~~Bloc~~
- Parties
- Références
- Flux entre parties

VISION INTERNE

✓ Diagramme de bloc interne

➔ Du BDD à l'IBD

Parties/Références

Connecteurs

Résumé

COMPLEMENTARITE IBD / BDD

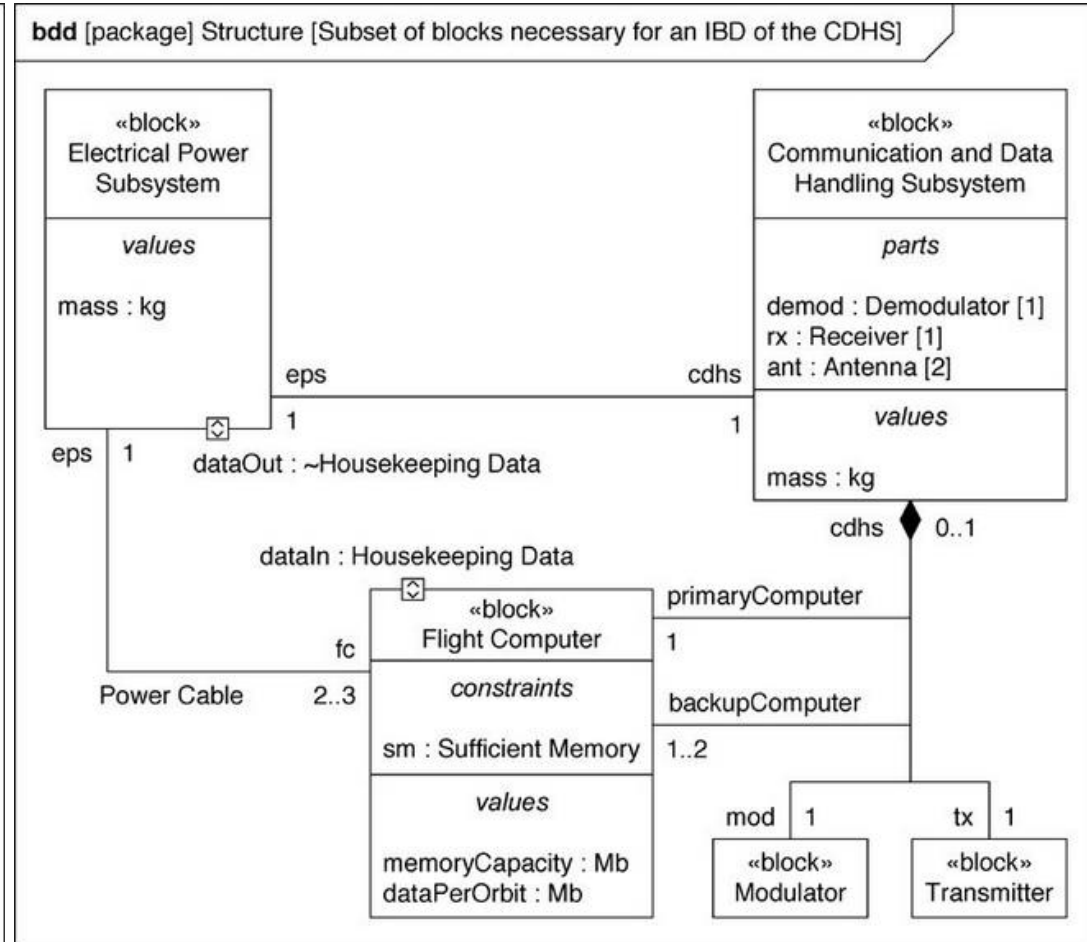
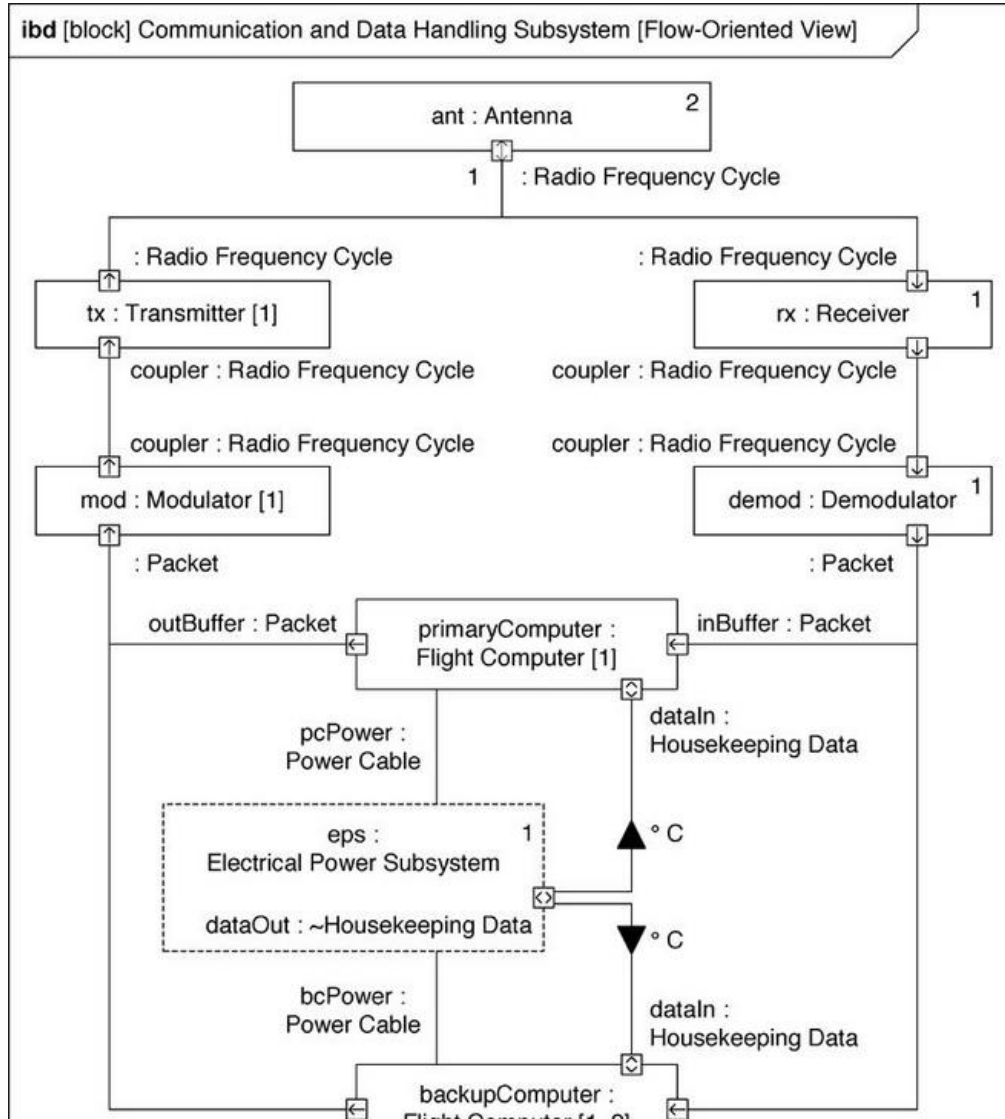


Figure 7 - BDD/IBD d'un même bloc (SysML Distilled - L.Delligatti)

COHERENCE IBD / BDD

Essentiel

Parties (part properties)

Ex : demod, rx, ant, primaryComputer, backupComputer, mod, tx

Annexe

Références (reference properties) : eps

Cardinalités

Ajouts

Connecteurs, flux

VISION INTERNE

- ✓ Diagramme de bloc interne

- ✓ Du BDD à l'IBD

- ➔ Parties/Références

Connecteurs

Résumé

PARTIES ET REFERENCES

Parties

Cardinalité : coin haut droit ou [...]

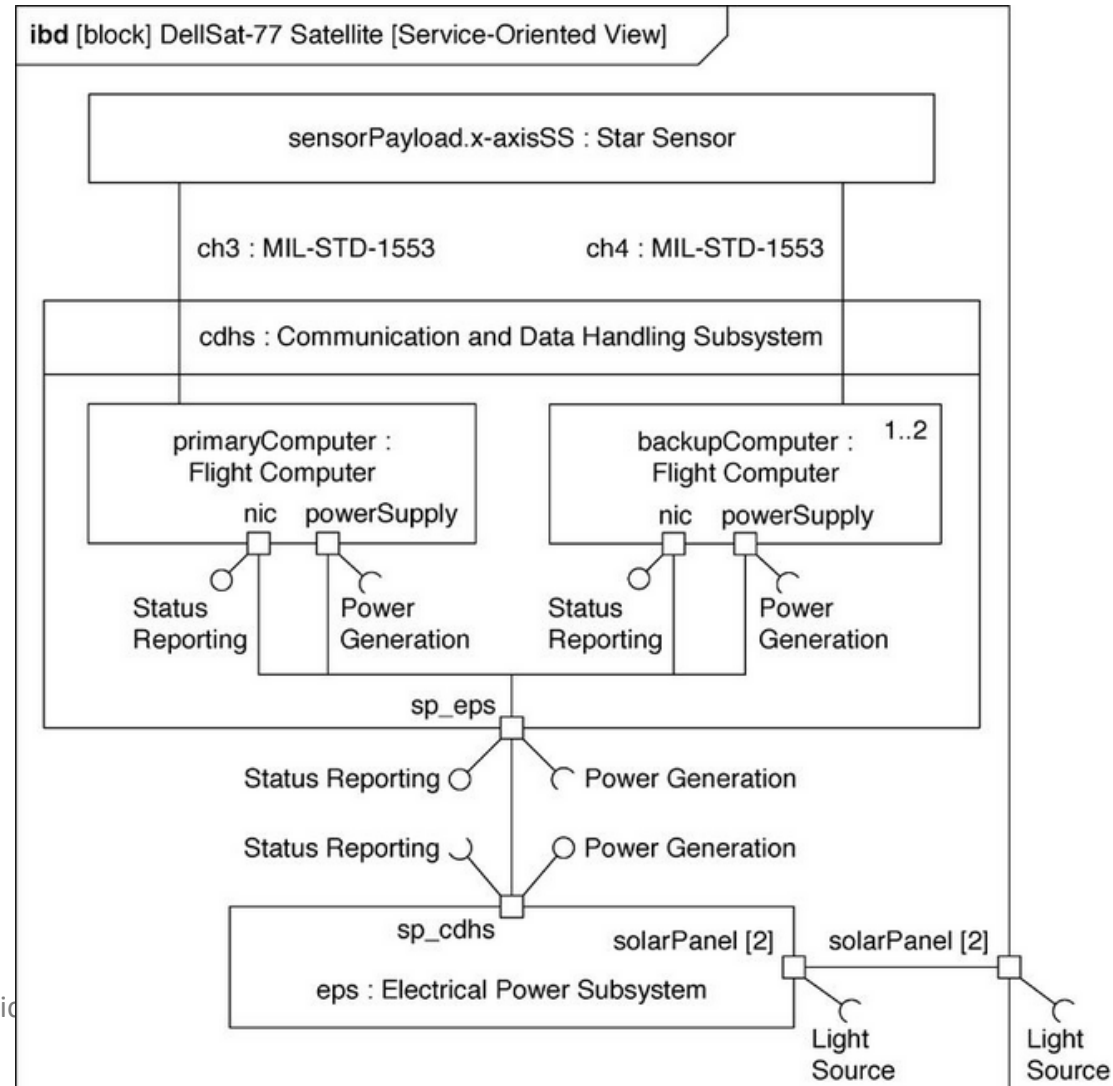
nom : type

Références

nom : type

Cadre en pointillés

PARTIES IMBRIQUEES



VISION INTERNE

- ✓ Diagramme de bloc interne
- ✓ Du BDD à l'IBD
- ✓ Parties/Références
- ➔ Connecteurs

Résumé

CONNECTEURS

Association

nom: type

Cohérence BDD : nom et/ou type + lien propriétés

Ports

Voir BDD : Carré sur bordure

nom: type ou \sim nom: type

VISION INTERNE

- ✓ Diagramme de bloc interne
- ✓ Du BDD à l'IBD
- ✓ Parties/Références
- ✓ Connecteurs
- ➔ Résumé

RESUME

Description interne de blocs

Conception à partir de la définition du bloc

Visualisation des composants et interfaces

Interconnexions et flux entre parties

DYNAMIQUE ET SPECIFICATIONS

OBJECTIFS

Dynamique d'un système embarqué

Inventaire des paramètres des sous-systèmes

Formalisation du cahier des charges

DYNAMIQUE ET SPECIFICATIONS

→ Activité de blocs

Séquence et état

Paramétrique

Allocations

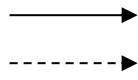
Résumé

ACTIVITE DE BLOCS

Principe

Formalisme du diagramme d'activité
Fonctionnement attendu d'un bloc
Utile en phase d'analyse pas de spéc.

Éléments

Flux/Contrôle de matière, énergie ou données 

Activité de traitement 

Alternatives / regroupement 

Objet : Etat 

Symbole de temporisation 

DYNAMIQUE ET SPECIFICATIONS

✓ Activité de blocs

→ Séquence et état

Paramétrique

Allocations

Résumé

DIAGRAMME DE SEQUENCE

Utilité

Interaction entre blocs

Signaux synchrones et asynchrones

Forme

Ligne de vie, messages et barres d'exécutions

Contraintes de temps / paramétriques

Alt / Loop (0, *) de UML 2

DIAGRAMME D'ETAT

Utilité

- Etats d'un bloc et transitions possibles
- Encadrement de l'implémentation
- Visualisation cas et transitions non gérées

Éléments

- Etat initial / final
- Simple ou composite
- Transitions éventuellement temporelles

DYNAMIQUE ET SPECIFICATIONS

- ✓ Activité de blocs
- ✓ Séquence et état

➔ Paramétrique

Allocations

Résumé

DIAGRAMME PARAMETRIQUE

Principe

Equation interne

Paramètres pris en compte

Exemple

Cf Figure 9

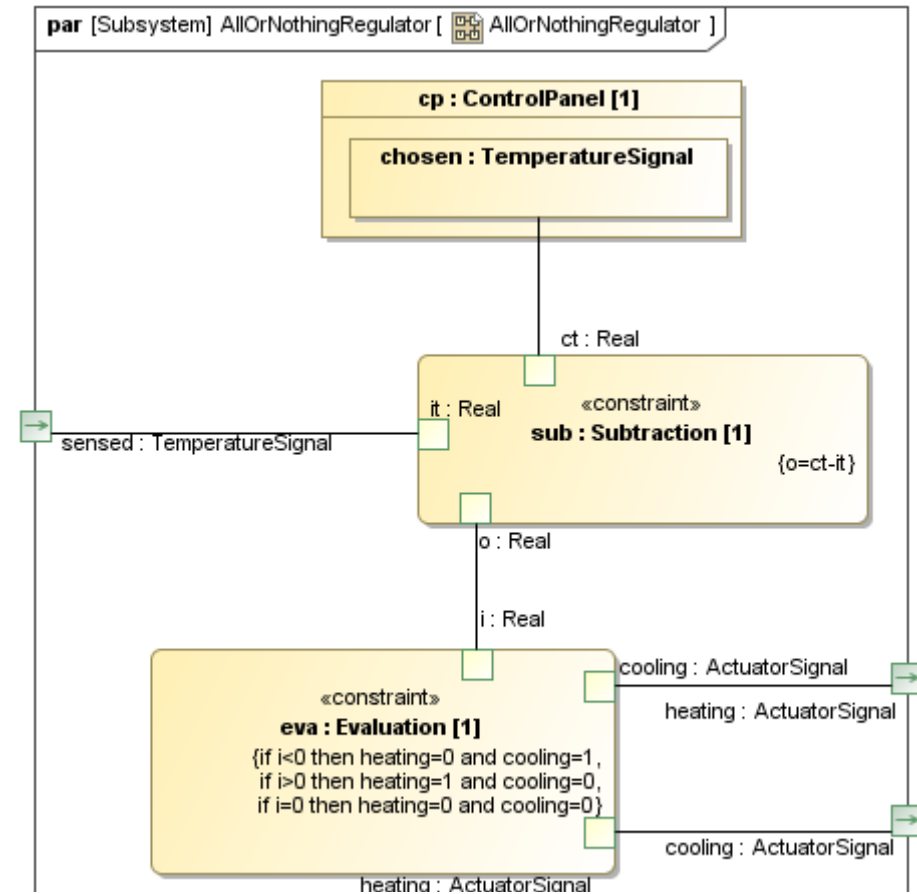


Figure 9 - Diag. paramétrique thermostat (src: <https://docs.nomagic.com>)

DYNAMIQUE ET SPECIFICATIONS

- ✓ Activité de blocs
- ✓ Séquence et état
- ✓ Paramétrique
- ➔ Allocations

Résumé

RELATIONS D'ALLOCATIONS

Types

Structurel : Entre blocs

Comportemental : activité « allocate » ...

Formes

Dans le diagramme

Synthèse matricielle

DYNAMIQUE ET SPECIFICATIONS

- ✓ Activité de blocs
- ✓ Séquence et état
- ✓ Paramétrique
- ✓ Allocations
- ➔ Résumé

RESUME

Dynamique d'un système embarqué

Inventaire des paramètres des sous-systèmes

Formalisation du cahier des charges

BILAN

ATELIER 4 – ASPECT LOGICIEL DU BDD

Sujet

A partir du diagramme de définition de blocs, dans un des langages suivants : Java, C#, Python, C/C++, js, PHP :

- Créer une interface IMotion inspirée du bloc Motion
- Proposer une implémentation de la méthode scan() qui utilise IMotion pour scanner une pièce et remplir une grille de 66x66 entiers¹.
- Optionnel : Transformer cette grille en quadtree

Compétence

Passer d'un diagramme de blocs au code.

¹ On suppose que : l'unité est 40cm, le bot part d'un coin de la pièce case (1,1), mur sur la gauche. Une case inaccessible vaut -1, inexplorée 0, accessible 1.

Au début, les colonnes et lignes 0 et 65 sont à -1, la case (1,1) à 1, les autres à 0.

MAINTENANT, VOUS POUVEZ...

Aborder méthodiquement un projet d'ingénierie des systèmes

Modéliser ses interactions avec son environnement

Formaliser les exigences du système

Enumérer les composants répondant aux exigences

Spécifier les composants matériels et logiciels en SysML

Passer d'un diagramme de blocs au code