



Déploiement dirigé par les modèles de jumeaux numériques dans les environnements intelligents

Gaëlic Bechu, Antoine Beugnard, Caroline Gl Cao, Quentin Perez, Christelle Urtado, Sylvain Vauttier

► To cite this version:

Gaëlic Bechu, Antoine Beugnard, Caroline Gl Cao, Quentin Perez, Christelle Urtado, et al.. Déploiement dirigé par les modèles de jumeaux numériques dans les environnements intelligents. HUT LaConf 2022 - L'interdisciplinarité au service des environnements intelligents, Nov 2022, Montpellier, France. hal-03862279

HAL Id: hal-03862279

<https://imt-mines-ales.hal.science/hal-03862279>

Submitted on 24 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Déploiement dirigé par les modèles de jumeaux numériques dans les environnements intelligents

Gaëlic Béchu¹, Antoine Beugnard¹, Caroline G. L. Cao¹,
Quentin Perez², Christelle Urtado², Sylvain Vauttier²

¹LabSTICC, IMT Atlantique, Brest, France

¹{firstname.lastname}@imt-atlantique.fr

²EuroMov Digital Health in Motion, Univ Montpellier, IMT Mines Ales, Ales, France

²{firstname.lastname}@mines-ales.fr

Résumé—Le concept de jumeau numérique a été introduit par les praticiens de l'ingénierie système dans le contexte de l'industrie 4.0. Un jumeau numérique fournit une représentation abstraite d'un système pour faciliter son observation et son contrôle en s'appuyant sur les techniques de l'internet des objets et de l'intelligence artificielle. Malgré sa pertinence, le concept de jumeau numérique manque encore d'une définition concrète et opérationnelle. Dans cet article, nous présentons un métamodèle permettant de décrire le déploiement de jumeaux numériques dans des environnements intelligents. Ce métamodèle couvre les différents aspects d'un déploiement, tels que la définition des différents composants matériels et logiciels qui composent l'architecture du système cyber-physique sur lequel le jumeau numérique est déployé, ou la définition des différentes tâches d'installation, de configuration et d'instantiation qui composent directement le processus de déploiement. Le métamodèle permet également la définition de configurations multiples pour adapter le déploiement à la variabilité des contextes d'exécution ou à l'évolution des systèmes. Nous avons réalisé une validation de ce métamodèle dans le cadre du projet HUT (Human at home projecT). Le projet HUT a mis en place un appartement observatoire collectant des données sur l'activité quotidienne de deux occupants à l'aide d'un ensemble de capteurs connectés. L'appartement étant démantelé à la fin du projet, un jumeau numérique de l'appartement et de ses capteurs permettrait de poursuivre les expérimentations en simulant la collecte des données à partir des historiques existants. Notre métamodèle permet de décrire intégralement le déploiement du jumeau numérique d'un capteur dans un environnement domotique. Ce modèle de déploiement a pu être directement implémenté à l'aide de scripts, validant ainsi la couverture et la complétude du métamodèle.

Index Terms—Jumeau Numérique, Ingénierie logicielle dirigée par les modèles, Déploiement d'architectures logicielles, Environnements intelligents.

I. INTRODUCTION

Le HUMAN at home Project¹ est un projet pluridisciplinaire étudiant les impacts des environnements intelligents et fortement connectés sur le bien-être. Il a mis en place un appartement observatoire équipé d'une grande variété de capteurs (température, hygrométrie, CO₂, lumière, présence, ouverture des portes, consommations d'eau et d'électricité...) recueillant les données générées par l'activité quotidienne d'occupants volontaires pendant une année. Il a ainsi été possible de recueillir des données pendant trois ans. Le projet HUT touchant

à sa fin, l'équipement de l'appartement doit être entièrement démantelé, posant la problématique de la poursuite des travaux de recherche associés. Il s'agit dans notre cas d'étudier les possibilités d'apprendre les habitudes des occupants de l'appartement, à l'aide de techniques d'intelligence artificielle, afin d'anticiper leurs besoins et de déclencher automatiquement des actions facilitant leur quotidien ou améliorant leur confort.

Étudié par l'ingénierie système dans le cadre de l'industrie 4.0, le concept de jumeau numérique est une solution à cette problématique de valorisation post-projet des données collectées. Un jumeau numérique se définit comme un ensemble logiciel couplé à des objets connectés (IoT) ayant pour fonction de fournir en temps réel une représentation virtuelle d'un système physique [1]. Cette définition conceptuelle, basée sur le rôle et les caractéristiques d'un jumeau numérique, manque encore de déclinaisons concrètes et opérationnelles [2].

En nous appuyant sur notre expertise dans le domaine de l'ingénierie dirigée par les modèles de logiciels et plus spécifiquement sur les langages de description d'architectures (ADLs) [3], [4], nous présentons dans cet article un métamodèle permettant de décrire le déploiement d'un jumeau numérique dans un environnement intelligent.

La suite de l'article est organisée de la manière suivante. La section II détaille le métamodèle de déploiement en présentant les différents aspects couverts. La section III présente le cas d'étude, dérivé du projet HUT, grâce auquel nous avons doublement validé la capacité du métamodèle à décrire le déploiement du jumeau numérique d'un capteur connecté dans un environnement domotique. La section IV fait une synthèse de travaux connexes permettant de positionner notre contribution dans l'état de l'art. La section V évalue ces premiers résultats et propose des perspectives pour ce travail.

II. UN MÉTAMODÈLE POUR LE DÉPLOIEMENT DIRIGÉ PAR LES MODÈLES DE JUMEUX NUMÉRIQUES

Le rôle et l'utilité des jumeaux numériques fait consensus en ingénierie système, qui l'identifie comme l'une des technologies clés de l'industrie 4.0, en complément de l'internet des objets et de l'intelligence artificielle. Le standard ISO23247 [5] et certains articles [2], [6] fournissent des architectures de

1. <https://www.hut-occitanie.eu/>

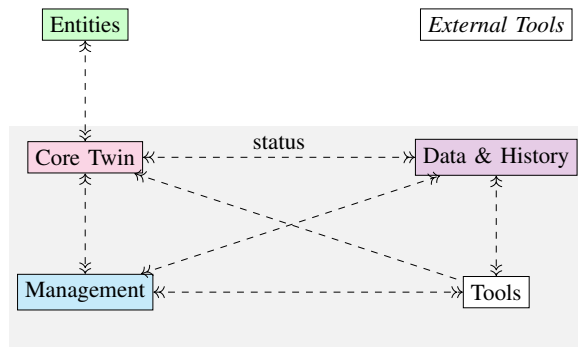


FIGURE 1 – Macro-architecture d'un Digital Twin

référence, définissant les différents types d'éléments composant un jumeau numérique, leurs interactions et leurs responsabilités. Mais ces architectures conceptuelles sont trop abstraites pour être utilisées de manière opérationnelle dans une démarche d'ingénierie dirigée par les modèles. La Figure 1 présente la macro-architecture que nous adoptons. Elle est composée de quatre sous-ensembles.

Core Twin correspond aux composants logiciels et matériels composant l'architecture cyber-physique du jumeau numérique. Elle inclut l'ensemble des capteurs, actionneurs, pilotes et applications assurant la collecte, l'analyse et le stockage des données gérées par le jumeau numérique. Elle contient également l'infrastructure (connecteurs physiques ou logiques, middlewares, ...) supportant les interactions entre les différents éléments de l'architecture.

Data & History désigne l'ensemble des données collectées et traitées pour alimenter les modèles gérés par le jumeau numérique puis généralement stockées pour permettre des analyses historiques ou statistiques. Il s'agit des informations échangées avec le système observé et piloté à l'aide du jumeau numérique (valeurs de capteurs, commandes transmises aux actionneurs, ...) mais également les modèles du système construits et maintenus par le jumeau numérique (modèles 3D, statistiques, modèles d'apprentissage, ...). Les données techniques liées à la configuration et au fonctionnement intrinsèque du jumeau numérique (logins, adresses IP, ...) font partie du *Core Twin*.

Tools identifie l'ensemble des outils d'analyse, de simulation ou d'intelligence artificielle qui sont déployés conjointement au jumeau numérique comme ressources pour la construction et la mise à jour de modèles ou le traitement des données.

Management inclut les logiciels, scripts et données utilisés pour gérer l'installation et l'exploitation du jumeau numérique (console d'administration, logs, ...).

External Tools fait référence aux outils et infrastructures présents dans l'environnement du jumeau numérique et qui fournissent des ressources nécessaires au fonctionnement du *Core Twin*, des *Tools* ou du *Management*.

Entities représente le système socio-technique, incluant potentiellement des humains, qui est observé ou piloté à l'aide du jumeau numérique. Dans le cadre du projet HUT, le système

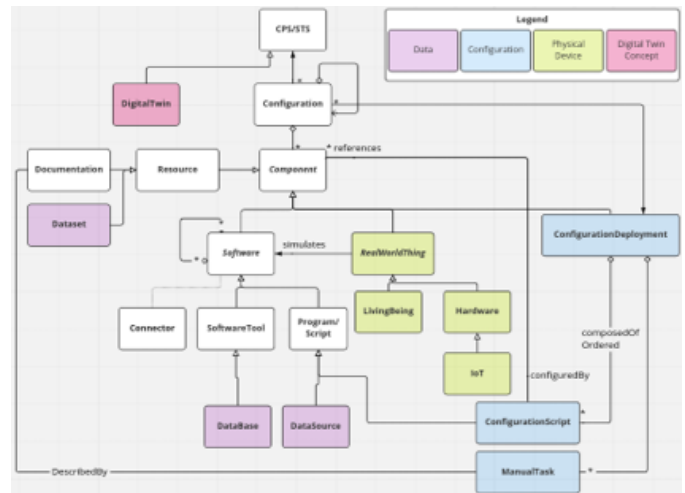


FIGURE 2 – Métamodèle de déploiement de jumeaux numériques (excerpt)

socio-technique à modéliser est l'appartement connecté et ses occupants, observés à l'aide des données collectées par les capteurs connectés répartis dans cet environnement.

Pour rendre opérationnelle cette macro-architecture, nous avons initié la conception d'un métamodèle permettant la définition de l'ensemble des éléments nécessaires au développement et à l'exploitation d'un jumeau numérique, dans une approche dirigée par les modèles. De manière agile, notre métamodèle, présenté en extrait sur la figure 2, fournit dans un premier temps un support au déploiement de jumeaux numériques. Il pourra être ultérieurement étendu à d'autres étapes du cycle de vie. Le métamodèle reprend le code couleur de la macro-architecture par souci de clarté. La partie verte correspond à la définition des éléments physiques (*RealWorldThing*), matériels *Hardware* ou humains *LivingBeing*. La partie rouge aux éléments composant le *Core Twin*. La partie bleue décrit la gestion du déploiement, définie comme des séquences de tâches manuelles ou automatiques d'installation et de configuration. La partie violette décrit la gestion des données, qu'il s'agisse de la création des bases de données (*DataBase*) assurant le stockage, de la configuration des accès aux bases de données (*DataSource*) ou de l'utilisation d'historiques de données en tant que ressources (*DataSet*).

La partie blanche du métamodèle permet de décrire l'architecture des jumeaux numériques (systèmes cyber-physiques) et des systèmes sociaux-techniques correspondants. Conceptuellement, il s'agit de décrire dans les deux cas l'architecture de systèmes de systèmes, composés au niveau de granularité le plus fin de composants élémentaires (sans architecture interne explicitement définie et prise en charge par l'outillage de déploiement). De plus, le métamodèle permet de représenter la variabilité de l'architecture d'un système de systèmes, afin de l'adapter à la variabilité ou à la variété des contextes d'exécution des systèmes socio-techniques et donc des contextes d'exécution des jumeaux numériques qui les

monitorent. Ainsi, dans notre métamodèle, un CPS/STS est associé à un ensemble de *Configuration* qui représente chacune une variante de la définition de son architecture. Chaque *Configuration* est composée soit d'un ensemble de *Configuration*, définissant pour chaque sous-système composant la configuration architecturale choisie, soit directement d'un ensemble de composants (niveau terminal de la décomposition architecturale). Parmi les sous-classes de composants, les *Connecteur* ont pour rôle la définition des liens qui sont établis entre les composants au sein des architectures.

III. CAS D'ÉTUDE : MODÉLISER LE DÉPLOIEMENT DE JUMEUX NUMÉRIQUES POUR LE PROJET HUT

Le projet HUT fournit un cadre particulièrement intéressant pour explorer les relations entre intelligence artificielle (IA) et génie logiciel (GL). Dans la perspective GL pour l'IA, il permet d'étudier la conception de mécanismes d'entraînement, de déploiement et d'évolution de modèles d'apprentissage automatiques, entraînés pour participer au contrôle d'environnements intelligents. L'intégration de ces différents mécanismes à des architectures logicielles est une problématique en pleine expansion [7]. Dans la perspective inverse d'IA pour le GL, la spécificité du projet HUT est de collecter des données provenant de l'activité quotidienne des occupants de l'appartement. Ces données permettent d'étudier l'entraînement de modèles pour identifier des activités ou apprendre des habitudes quotidiennes. L'ambition est d'anticiper les besoins des utilisateurs et de déclencher automatiquement des actions ou d'adapter le comportement d'un environnement intelligent pour satisfaire leurs attentes. En d'autres termes, il s'agit d'étendre le périmètre de la sensibilité au contexte de l'architecture de contrôle d'un environnement intelligent pour qu'elle intègre une forme de compréhension des intentions des utilisateurs ou de leurs états psychiques et physiques (empathie artificielle) [8].

Un jumeau numérique peut avoir de multiples fonctions, couvrant l'ensemble du cycle de vie d'un système [1], [6]. Dans le projet HUT, le jumeau numérique joue le rôle de *Digital Proxy*. Il est constitué de représentations virtuelles des capteurs de l'appartement qui fournissent des API de haut niveau facilitant l'utilisation des données produites. Le jumeau numérique permet l'extension logicielle des fonctionnalités des capteurs, comme la gestion d'historiques ou le calcul de données agrégées et dérivées. Dans une situation classique, le jumeau numérique est alimenté en temps réel par les données des capteurs qu'il représente. Nous utilisons le rôle de proxy du jumeau numérique pour simuler l'existence des capteurs et prolonger les possibilités d'expérimentations malgré le démantèlement de l'appartement (*Digital Archive*). Le jumeau numérique est alors alimenté par les historiques des données produites par les capteurs et sauvegardées dans des bases de données. Du point de vue des applications, l'absence des capteurs et la simulation de la collecte des données sont complètement transparentes puisqu'elles utilisent de manière

identique le jumeau numérique comme interface d'accès aux données.

Le jumeau numérique de l'appartement HUT est composé des jumeaux numériques des différents capteurs. Il est conforme à notre métamodèle qui définit l'architecture d'un jumeau numérique comme une structure composite hiérarchique (un système de systèmes). Cette modularité permet d'envisager la définition de différentes configurations de capteurs pour répondre à différents besoins d'expérimentation.

La Figure 3 présente l'expérimentation qui a servi de support à la validation de notre métamodèle. Elle consiste à entraîner un modèle de prédiction et à utiliser ce modèle au sein d'un environnement domotique. Concrètement, nous construisons un jumeau numérique prédisant la valeur d'un capteur de pression indiquant la présence d'une personne dans son lit. Nous utilisons cette prédiction pour anticiper le lever de cette personne et piloter l'allumage progressif d'un éclairage (réveil doux par effet d'aube). L'environnement domotique est constitué d'un ensemble d'éléments matériels (capteurs, actionneurs, ordinateurs, routeurs, dongles...) et logiciels (système d'exploitation, serveurs applicatifs, serveur domotique...).

Le prédicteur a été réalisé en utilisant les données d'un dataset produit dans des conditions similaires au projet HUT (appartement connecté occupé au quotidien) [9]. Ce dataset préfigure celui qui pourra être créé à terme à partir des données collectées dans le projet HUT. Mais elles nécessitent encore des étapes de nettoyage, de consolidation et de mise en forme pour former un dataset directement exploitable. C'est une perspective importante car les datasets portant sur des données domestiques quotidiennes sont rares ou non disponibles en ligne. Le modèle de prédiction a été construit grâce au framework *Sktime*² en entraînant un *naive forecaster* qui est capable d'apprendre et de reproduire des motifs récurrents dans des séries de données temporelles. Le prédicteur est ensuite intégré à l'environnement domotique à l'aide d'un jumeau numérique qui combine les fonctionnalités du prédicteur à celles du capteur de pression. Ainsi le jumeau numérique permet d'étendre les capacités du capteur de pression initial en y ajoutant des mécanismes d'IA. C'est un cas d'étude typique pour explorer les bonnes pratiques de conception d'architectures logicielles pour des jumeaux numériques intégrant de l'IA.

La proposition d'un métamodèle capable de définir formellement l'architecture d'un jumeau numérique est une première étape indispensable à une démarche rigoureuse et outillée d'ingénierie. La Figure 5 présente la modélisation du jumeau numérique mis en oeuvre dans le cas d'étude. Notre métamodèle étant orienté composants, nous utilisons les notations des *Composite Structure Diagram* d'UML 2.0³ comme syntaxe concrète graphique associée aux concepts de notre métamodèle. Les composants de l'architecture sont directement représentés comme des composants sur le dia-

2. <https://www.sktime.org/>

3. <http://uml.org>

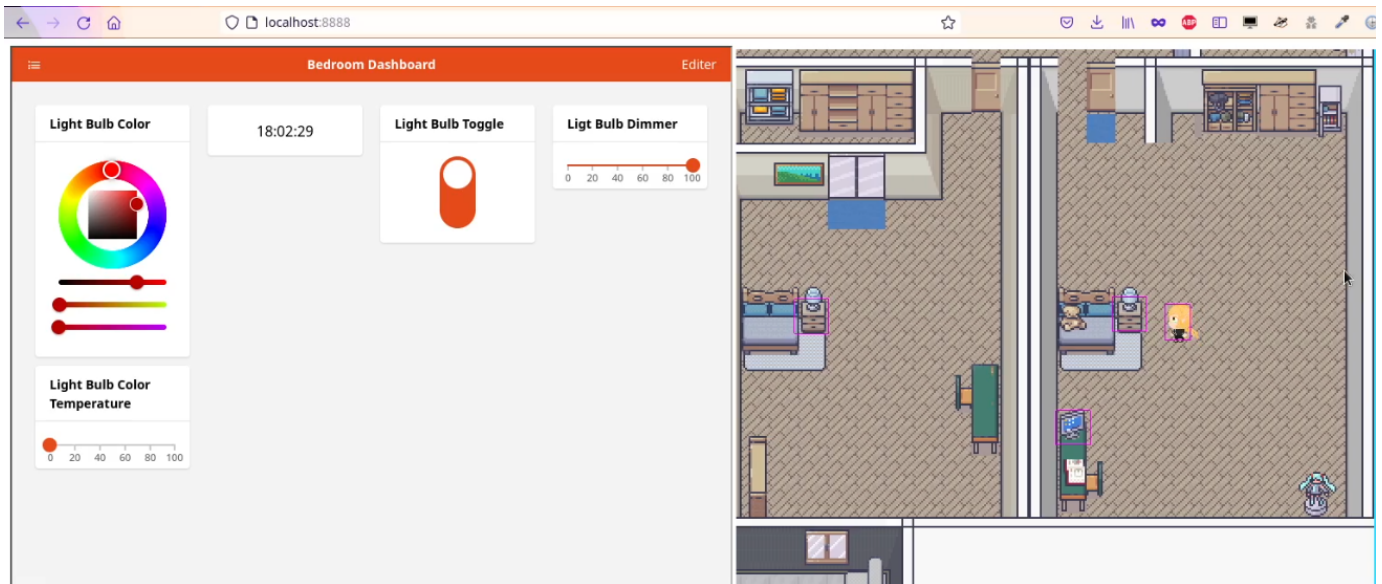


FIGURE 3 – Interfaces graphiques de l’environnement conçu pour expérimenter le développement de jumeaux numériques de l’appartement connecté du projet HUT



FIGURE 4 – Exemple de matériels utilisés dans la réalisation de l’environnement d’expérimentation

gramme. Les types des composants (i.e. leurs classes dans le métamodèle) sont précisés à l’aide des étiquettes textuelles des composants. Notre métamodèle est capable de représenter l’ensemble logiciel et matériel composant le système domotique et les jumeaux numériques en tant que systèmes cyber-physiques. Le système domotique est constitué d’une Raspberry Pi 4 sur laquelle est installée un système d’exploitation Raspberry Pi OS et le logiciel de domotique OpenHAB⁴. OpenHAB joue le rôle de hub et peut intégrer différents protocoles de communication par la configuration de *bindings*. La carte Raspberry Pi est ainsi équipée d’un *dongle* Z-Wave pour communiquer avec des objets connectés utilisant ce protocole standard de la domotique. Dans notre cas d’étude, il s’agit d’une ampoule connectée dont il est possible de commander l’allumage, l’intensité et la couleur (voir la Figure 4). Par

4. <https://www.openhab.org/>

ailleurs, la carte Raspberry Pi dispose d’une carte wifi utilisée pour communiquer, via un réseau local dédié, avec un serveur applicatif sur lequel est installé le prédicteur. Il s’agit d’un serveur Python Flask⁵, installé sur un PC sous linux, permettant d’utiliser le prédicteur au travers d’une API REST. OpenHAB utilise ainsi le prédicteur au travers d’un *binding* HTTP.

La Figure 3 présente les différentes interfaces graphiques implémentées pour le développement du jumeau numérique de l’appartement connecté du projet HUT. Au travers de ces interfaces, il est possible de déplacer un avatar dans une représentation 2D de l’appartement et d’interagir avec les objets physiques mis en œuvre dans cette étude. Une vidéo de démonstration est disponible en ligne : https://youtu.be/_jDCzb_1YhM

OpenHAB propose des représentations abstraites des objets connectés, les *Thing*, fournissant des API permettant d’interagir avec les objets connectés (i.e. lire des informations ou envoyer des commandes). Nous configurons trois *Things* pour utiliser la lampe connectée ainsi que le prédicteur et le capteur de pression (simulé) gérés par le jumeau numérique.

Nous obtenons ainsi un premier niveau de validation de notre métamodèle. Ses concepts permettent de définir l’architecture logicielle et matérielle d’un jumeau numérique en tant que système cyber-physique en modélisant les éléments qui la composent, leurs relations d’imbrications (quels éléments servent d’hôtes pour le déploiement et l’exécution d’autres éléments) et les connections nécessaires à leurs interactions. Nous avons doublement validé cette capacité de modélisation à l’aide d’un second cas d’étude (configuration d’un bras robotique pour l’apprentissage de tâches de “*pick and place*”) [10].

5. <https://palletsprojects.com/p/flask/>

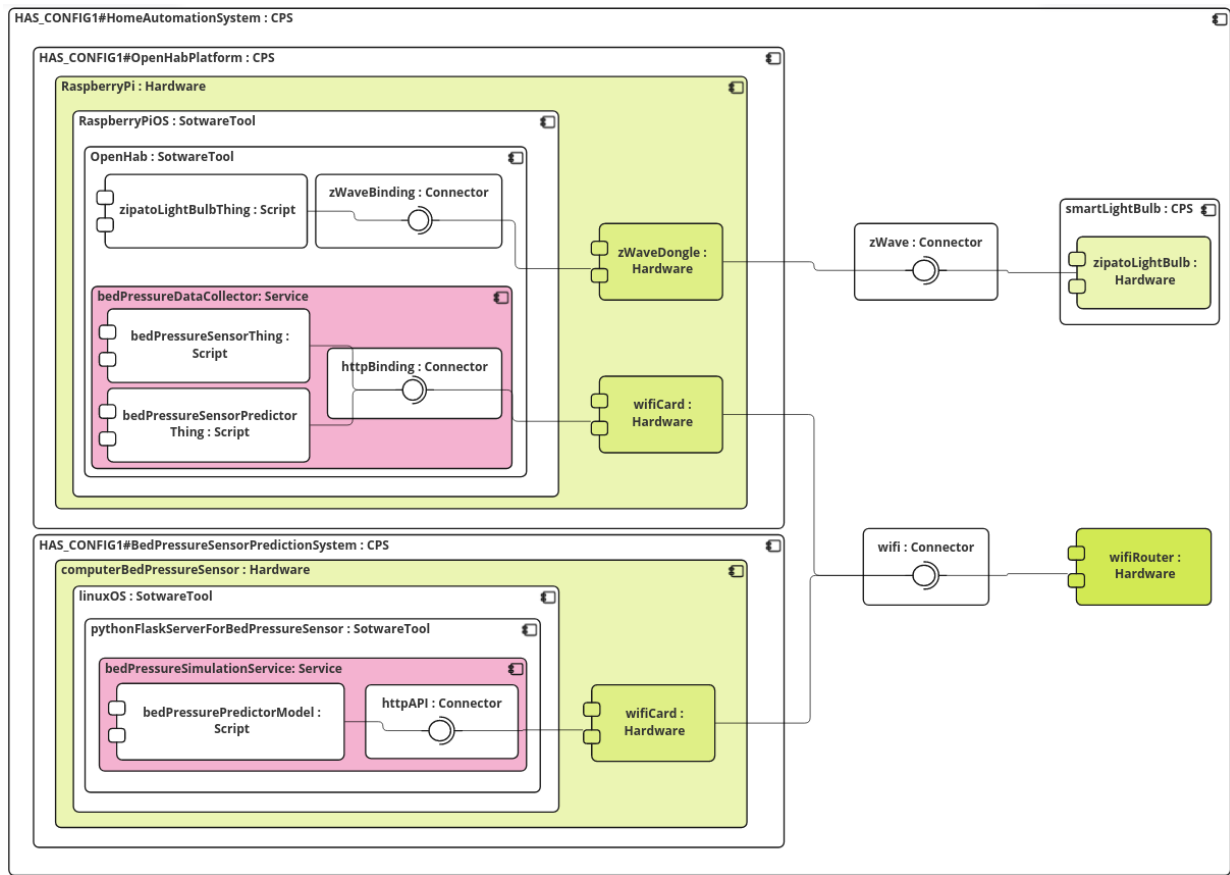


FIGURE 5 – Modèle de l'architecture du jeu numérique et de son environnement

```
- name: "Install OpenHab backup"
  hosts: 159.31.103.193
  tasks:
    - name: "Copy OpenHab backup on Raspberry Pi"
      copy:
        src: ./oh3backup_not_full.back
        dest: /tmp/oh3backup_not_full.back
        owner: pi
        mode: 0644
    - name: "Stop openhab service on Raspberry Pi"
      service:
        name: openhab
        state: stopped
        become: true
    - name: "Install OpenHab backup on Raspberry Pi using CLI"
      ansible.builtin.shell: yes | openhab-cli restore /tmp/oh3backup_not_full.back
      register: openhab_cli_output
      become: true
    - name: "Openhab-CLI output"
      debug:
        msg: "{{ openhab_cli_output.stdout_lines }}"
    - name: "Start service openhab"
      service:
        name: openhab
        state: restarted
        become: true
```

FIGURE 6 – Extrait d'un script Ansible implémentant des tâches de déploiement du jeu numérique

Un second niveau de vérification est obtenu grâce à l'implémentation complète du déploiement et de la configura-

tion des éléments logiciels définis par le modèle à l'aide d'un ensemble de scripts Ansible⁶ (voir la Figure 6). Cela ouvre la perspective d'une génération automatique de l'ensemble de ces scripts par transformation de modèles. Il sera nécessaire de détailler la partie déploiement de configuration, simplement amorcée dans le métamodèle actuel.

IV. TRAVAUX CONNEXES

Paradigme récent, le jeu numérique fait toujours l'objet d'une abondante recherche pour préciser sa définition conceptuelle et ses applications emblématiques [1], [2], [11]. D'autres travaux recensent les standards technologiques (modélisation, web sémantique, middleware, IoT, IA...) pouvant être mis en oeuvre dans sa réalisation [12], [13], mais sans définir ou détailler les architectures fonctionnelles nécessaires ou recommandées.

Le standard ISO23247 [5] propose une architecture de référence qui organise suivant un modèle en couche les fonctionnalités générales d'observation, de collecte, de modélisation et de décision d'un jeu numérique. Minerva *et al.* [14] cataloguent les propriétés/fonctionnalités (identification, reflection, replication, persistency, memorization, servitization...) et les qualités (composability, promptness,

6. <https://www.ansible.com/>

accountability,...) attendues d'un jumeau numérique. Une architecture générale en couches, similaire à celle du standard, est présentée au travers d'un ensemble de cas d'usages (patient virtuel, ville intelligente, ...). Il y est noté que "[...] from a software platform perspective, the broad implementation and span of the Digital Twin concept will require an extreme flexibility in deploying, executing and managing the software [...]". Notre proposition répond directement à cette problématique.

Tekinerdogan et Verdouw [6] établissent également une liste des rôles joués par les jumeaux numériques et associent à chacun un patron de conception architectural. Mais à la différence des travaux précédents, aucun style architectural n'est proposé. De même, les patrons sont définis de manière conceptuelle. Ils ne fournissent pas de support concret à la description et à la gestion des architectures des jumeaux numériques, en tant que systèmes de systèmes ou de systèmes cyber-physiques.

Hugues *et al.* [15] présentent une approche similaire à notre proposition. Elle met également en oeuvre des solutions d'ingénierie dirigée par les modèles pour développer et déployer des jumeaux numériques. Mais les objectifs poursuivis sont différents. Dans ces travaux, l'objectif est d'utiliser un pipeline DevOps pour mettre en oeuvre les modèles numériques au sein d'une architecture existante (Azure IoT). Dans notre cas, les modèles numériques sont des composants au sein de l'architecture globale du jumeau numérique. Notre objectif est d'outiller le déploiement de jumeaux numériques en tant que systèmes cyber-physiques et systèmes de systèmes.

V. DISCUSSION ET CONCLUSION

Cet article présente les premières validations du métamodèle que nous définissons pour outiller le déploiement de jumeaux numériques avec une approche dirigée par les modèles. Une première caractéristique de notre proposition est sa genericité. En modélisant des systèmes de systèmes et des systèmes cyber-physiques, notre métamodèle offre la possibilité de représenter toutes les variantes du concept de jumeau numérique, dont les frontières sont encore floues. Par exemple, la composante IoT est parfois considérée comme partie intégrante du jumeau numérique ou comme faisant partie de son environnement. De même, le jumeau numérique est parfois défini uniquement comme un ensemble de modèles indépendamment de l'architecture logicielle et matérielle permettant leur exploitation. Une des perspectives de notre travail est de préciser la définition d'un jumeau numérique et de la formaliser afin d'évaluer la correction des modèles d'architectures (respect de bonnes pratiques, styles et patrons architecturaux).

Une autre perspective est la valorisation des jeux de données collectés lors du projet HUT par la création de jumeaux numériques des différents capteurs. Ils permettront, par composition, la création de jumeaux numériques de l'appartement HUT et la gestion de configurations adaptées aux différents besoins.

À l'instar du cas d'étude décrit dans cet article, nous souhaitons utiliser le jumeau numérique de l'appartement HUT

pour expérimenter l'intégration d'IA aux architectures cyber-physiques contrôlant des environnements intelligents. Définir le rôle architectural des mécanismes d'IA (extensions fonctionnelles des jumeaux numériques ou composants indépendants) est une perspective. Gérer l'entraînement et la mise-à-jour continus des modèles d'apprentissage (MLOps) en est une autre.

REMERCIEMENTS

Ce travail est financé pour partie par les instituts CARNOT MINES et TSN par le biais du projet TwinCoBot ainsi que par la Chaire Industrie du Futur de la Région Bretagne et l'University of South Australia, et pour partie par le GT VOCA du projet HUT financé par le FEDER et la Région Occitanie. Les auteurs remercient Panagiotis Papadakis, Christophe Lohr et Alireza Asvadi pour leur expertise.

RÉFÉRENCES

- [1] N. Julien and E. Martin, "How to characterize a digital twin : A usage-driven classification," *17th IFAC Symposium on Information Control Problems in Manufacturing*, pp. 894–899, 2021.
- [2] F. Tao, M. Zhang, and A. Y. C. Nee, "Digital twin driven smart manufacturing," *Academic Press*, 2019.
- [3] A. Mokni, C. Urtado, S. Vauttier, M. Huchard, and Z. H. Yulin, "A formal approach for managing component-based architecture evolution," *Science of Computer Programming*, pp. 24–49, 2016.
- [4] P. Perrotin, N. Belloir, S. Sadou, D. Hairion, and A. Beugnard, "Hosml : Socio-technical system adl dedicated to human vulnerability identification," *26th International Conference on Engineering of Complex Computer Systems*, 2022.
- [5] *ISO 23247-2 :2021 Automation systems and integration - Digital twin framework for manufacturing - Part 2 : Reference architecture*, International Standard Organization, 2021.
- [6] B. Tekinerdogan and C. Verdouw, "Systems architecture design pattern catalog for developing digital twins," *Sensors*, vol. 20, no. 18, 2020.
- [7] S. Garg, P. Pundir, G. Rathee, P. Gupta, S. Garg, and S. Ahlawat, "On continuous integration / continuous delivery for automated deployment of machine learning models using mlops," in *2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, 2021, pp. 25–28.
- [8] A. Paiva, I. Leite, H. Boukricha, and I. Wachsmuth, "Empathy in virtual agents and robots : A survey," *ACM Trans. Interact. Intell. Syst.*, vol. 7, no. 3, 2017.
- [9] G. Chimamiwa, M. Alirezaie, F. Pecora, and A. Loutfi, "Multi-sensor dataset of human activities in a smart home environment," *Data in Brief*, vol. 34, p. 106632, 2021.
- [10] G. Bechu, A. Beugnard, C. G. L. Cao, Q. Perez, C. Urtado, and S. Vauttier, "A software engineering point of view on digital twin architecture," in *IEEE 27th Int. Conf. on Emerging Technologies and Factory Automation (EFTA2022)*, 2022.
- [11] C. Semeraro, M. Lezoche, H. Panetto, and M. Dassisti, "Digital twin paradigm : A survey on technical literature review," *Computers in Industry*, vol. 130, 05 2021.
- [12] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Nee, "Enabling technologies and tools for digital twin," *Journal of Manufacturing Systems*, vol. 58, pp. 3–21, 03 2021.
- [13] M. Jacoby and T. Usländer, "Digital twin and internet of things—current standards landscape," *Applied Sciences*, vol. 10, no. 18, 2020.
- [14] R. Minerva, G. M. Lee, and N. Crespi, "Digital twin in the iot context : A survey on technical features, scenarios, and architectural models," *PROCEEDINGS OF THE IEEE*, vol. 108, no. 10, pp. 1785–1824, October 2020.
- [15] J. Hugues, A. Hristosov, J. J. Hudak, and J. Yankel, "Twinops - devops meets model-based engineering and digital twins for the engineering of cps," ser. MODELS '20. New York, NY, USA : Association for Computing Machinery, 2020. [Online]. Available : <https://doi.org/10.1145/3417990.3421446>