



Jumeau numérique, génie logiciel et architectures
(Quasi) tout système logiciel est un jumeau
numérique ; mais aucun n'est un jumeau
numérique

A. Beugnard

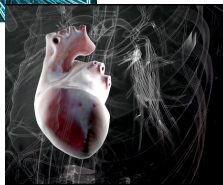
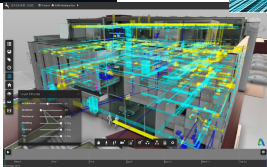
5 décembre 2022

 Cette présentation ne présente pas de résultats.

 Elle pose plus de questions qu'elle n'apporte de réponses.

C'est l'occasion de voir de nombreux points de vue sur un objet numérique complexe :

- Usage - services
- Architecture - frontière, interface, structure
- Variantes - feature model
- Adaptation - MAPE-K
- Configuration et administration d'un *machin* numérique



Une vidéo (c'est dynamique) : Exemple microsoft

- On décrit : descriptive twin (passif)
- On met à jour : informative twin (connecté au réel)
- On simule la configuration réelle : predictive twin (pour prédiction)
- On simule d'autres configurations : comprehensive twin (optimisations, améliorations)
- On reboucle : autonomous twin (apprentissage, décision)

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

Digital Twin Consortium

A digital twin is a virtual representation of real-world entities and processes, synchronized at a specified frequency and fidelity.

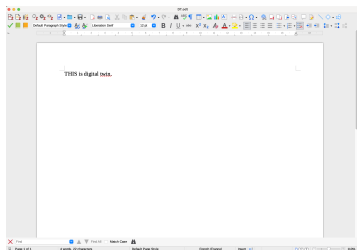
Digital twin systems transform business by accelerating holistic understanding, optimal decision-making, and effective action.

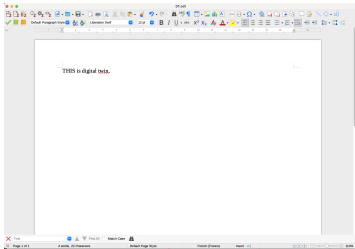
Digital twins use real-time and historical data to represent the past and present and simulate predicted futures.

Digital twins are motivated by outcomes, tailored to use cases, powered by integration, built on data, guided by domain knowledge, and implemented in IT/OT systems.

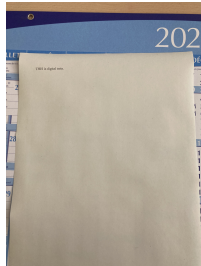
- (De quoi) un système, un asset, un truc réel (incluant du numérique)
- (Quoi) des modèles, des logiciels, des données et *des capteurs*.
- (Comment) couplage bidir avec un truc réel
- (Quand) sur tout le cycle de vie (y.c avant que le truc réel existe ? et après qu'il a été déconstruit ?)
- (Pour quoi) surveiller, analyser, apprendre, maintenir, optimiser, ...

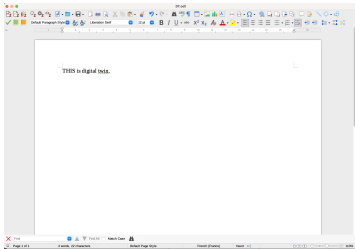
Plus largement, une méthode de construction de système(s) informatique(s).



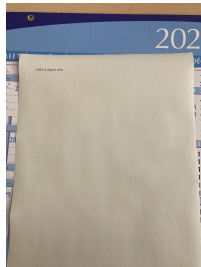


print ↘

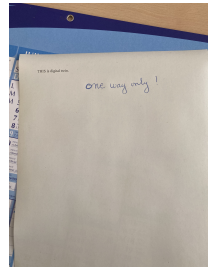


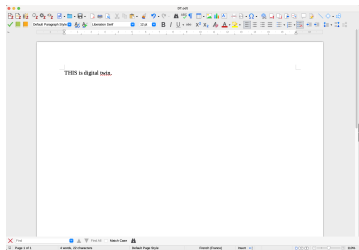


print ↘

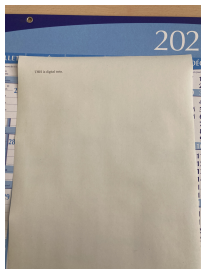


annotate →

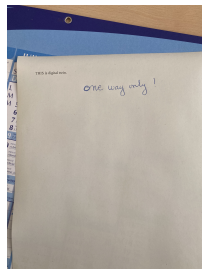




print ↘



update



annotate →

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

Une vision du DigitalTwin consortium, la table périodique :

1 Data Acquisition & Ingestion	9 Synthetic Data Generation	17 Enterprise System Integration	23 Edge AI & Intelligence	29 Prediction		39 Basic Visualization	45 Dashboards
2 Data Streaming	10 Ontology Management	18 Eng. System Integration	24 Command & Control	30 Machine Learning ML		40 Advanced Visualization	46 Continuous Intelligence
3 Data Transformation	11 Digital Twin (DT) Model Repository	19 OT/IoT System Integration	25 Orchestration	31 Artificial Intelligence AI	35 Prescriptive Recommendations	41 Real-time Monitoring	47 Business Intelligence
4 Data Contextualization	12 DT Instance Repository	20 Digital Twin Integration	26 Alerts & Notifications	32 Federated Learning	36 Business Rules	42 Entity Relationship Visualization	48 BPM & Workflow
5 Batch Processing	13 Temporal Data Store	21 Collab Platform Integration	27 Reporting	33 Simulation	37 Distributed Ledger & Smart Contracts	43 Augmented Reality AR	49 Gaming Engine Visualization
6 Real-time Processing	14 Data Storage & Archive Services	22 API Services	28 Data Analysis & Analytics	34 Mathematical Analytics	38 Composition	44 Virtual Reality VR	50 3D Rendering
7 Data PubSub Push	15 Simulation Model Repository	52 Device Management	54 Event Logging	56 Data Encryption	58 Security	60 Safety	51 Gamification
8 Data Aggregation	16 AI Model Repository	53 System Monitoring	54 Data Governance	57 Device Security	59 Privacy	61 Reliability	62 Resilience

● Data Services
 ● Integration
 ● Intelligence
 ● UX
 ● Management
 ● Trustworthiness

- CAD (3D)
- BIM (bâtiment) https://fr.wikipedia.org/wiki/Building_information_modeling
- Circuit, Modelica, Simulink, ...
- Math
- UML, SysML
- Autres espaces technologiques (XML, EMF, ...)

À relier, à faire coopérer, bref, à *fédérer*...

1. Manuelle (Digital Model - a priori/Digital Archive - a posteriori)
2. Réel \leftarrow Modèle (Digital Generator) [Rarement automatique]
3. Réel \rightarrow Modèle (Digital Shadow) [Automatisable, pas toujours]
4. Réel \leftrightarrow Modèle (Digital Twin) [Automatisable, pas toujours]

Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W : Digital Twin in manufacturing : A categorical literature review and classification. IFAC-PapersOnLine, 2018.

Q1 : Quid si le réel contient des données ? Sont-elles copiées ?

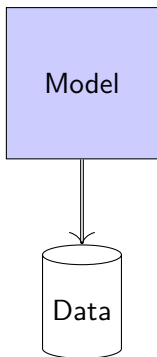
Q2 : Quid si le réel contient du code (comportements) ?

Sont-ils copiés ?

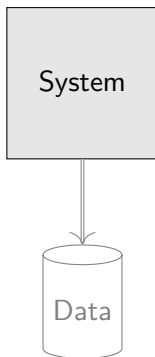
Systeme, modeles, services, donnees : on pense à quoi ?

- Conception&Développement (Digital Model)
 - Outil de communication
 - Simulation de comportements du system en développement
 - Evaluation de variantes de produits pour aider aux décisions de conception
- Production (Digital Twin)
 - Supervision de production
 - Trace du matériel, des composants, des pas de processus
- Utilisation&Maintenance (Digital Twin)
 - Fournit des informations sur l'état, l'histoire et l'utilisation du système
 - Facilite les améliorations
 - Permet de la maintenance prédictive
- Retrait (Digital Archive)
 - Transfert de connaissance aux générations suivantes du système

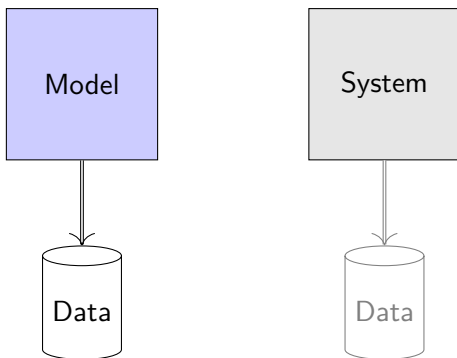
- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives



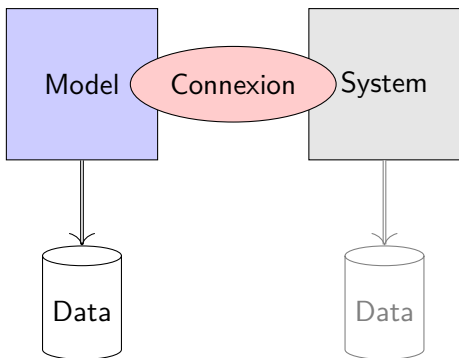
Design : Modèle seulement ; simulations



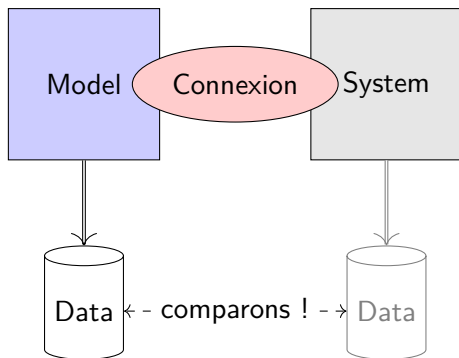
Building : Système seulement ; expériences



Assemblage : besoin de connexion

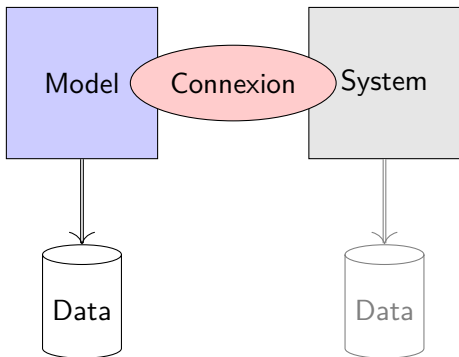


Connexion : intrusion dans le modèle et le système ? Du numérique dans le système, du matériel dans le modèle.



- attributs typé, valué (nécessaire, optionnelle, par défaut)
- valeur espérée ($\#x$), mesurée (x) ou estimée ($x+$) (a priori, calculée, comment?)


! on trace (mémorise) tout (ce qui est d'intérêt, ou le sera.)



Pour une même configuration, pour un même scénario.

Couplés ou non couplés :

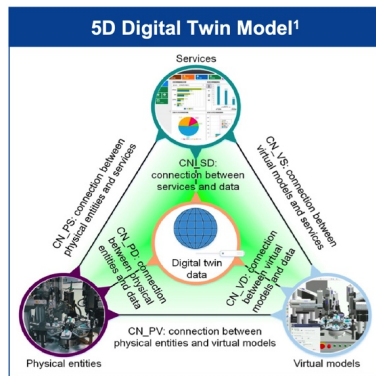
DT (1)	System (0)	Interpretation
x_1	x_0	Equal : OK (proxy twin)
x_1	x_0'	Close : OK but why (proxy twin, delayed ?)
$\#x_1$	x_0''	Close or equal : OK (mirror twin)
$\#x_1$	x_0'''	Different : KO why (mirror twin)
\emptyset	x_0	x_0 is not of interest. . .
x_1^+	\emptyset	x_1 emerged/is estimated

 The system has information that is not in its data. (ex : configuration, physical properties, etc.)

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - **Vue macroscopique**
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

1. Physical object : Beings, cyber-physical systems, ...
2. Digital object : Models, software infrastructures, VR, ...
3. Services : Monitoring, optimization, prediction, ...
4. Digital data : Sensor readings, manufacturing orders, ...
5. Connections : WiFi, ethernet, fieldbus, ...

Qi et al. : Enabling technologies and tools for digital twin. In : Journal of Manufacturing Systems, Elsevier, 2019



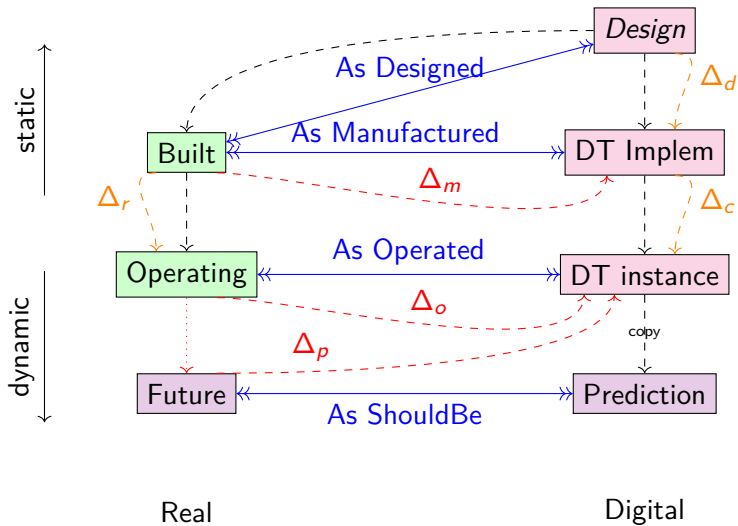
Ça marche aussi avec un SI...

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

- synchronicités (manuelle, uni-dir, bi-dir)
- niveau d'automatisation du couplage (Réel/Numérique) (%)
- fréquences
 - de mise à jour
 - d'analyse
 - fréquence des prédictions
 - fréquence de reconfiguration
- couverture
 - spatiale (systèmes, sous-systèmes, éléments, ...)
 - temporelle (présent, passé, futur, futur antérieur, ...)
- fidélités : Δ (réel - virtuel)

Pour qualifier, *valider* des JN...

Un jumeau numérique est *fidèle* lorsque ... ?



- Proxy - *sans* copie des données, *ni* des comportements (du SI?)
- Miroir - copie des données, *sans* les comportements
- Complet - copie des données *et* des comportements

Sur les trois, on peut faire des analyses, des simulations, de l'apprentissage, des rétroactions automatiques, etc.

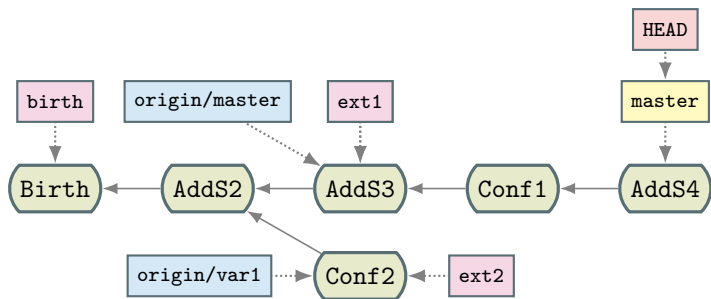
Q3 : Pour un jumeau complet, que se passe-t-il si les deux divergent dans leurs décisions ?

- On stoppe ? (invalidité)
- On alerte ?
- On donne raison au réel ?
- On donne raison au numérique ? (dispose de plus d'informations)
- ...

1. Jumeau numérique...implanté : en cours d'exécution
2. Instance de jumeau numérique : prêt à être exécuté, une configuration évaluée
3. Type de jumeau numérique : prêt à être configuré, sans valeurs

Tiens, de la modélisation multi-niveau !

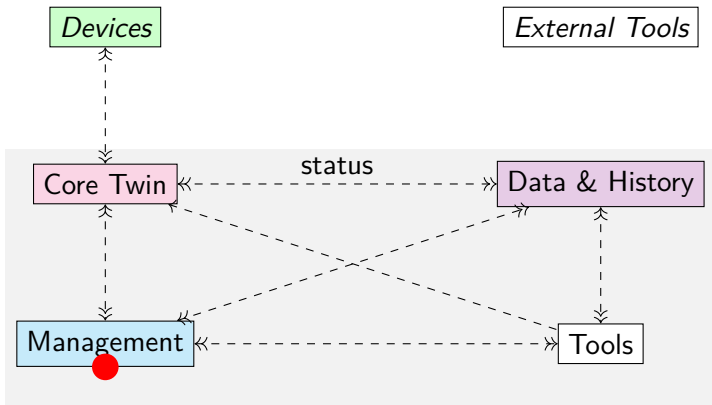
Et si on utilisait *git* pour gérer cela ?



1. Remote = type
2. Local/tag instance = Instance
3. Local/tag implem = implantation (running)

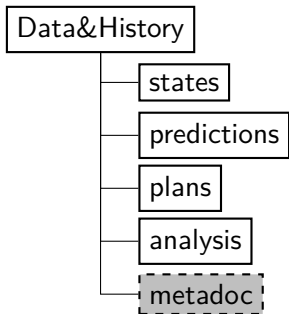
C'est bien *git*, on peut faire des *clones*.

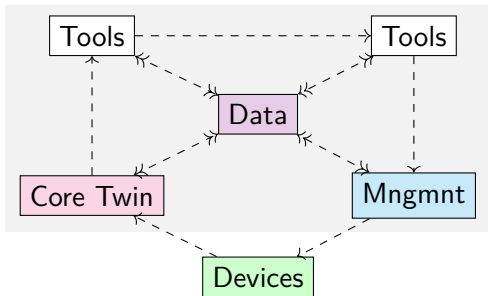
- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives



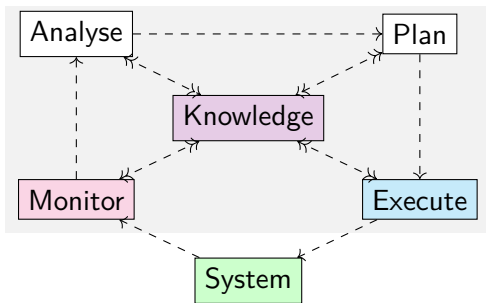
Trop de diversité d'outils. Difficile d'identifier des frontières et des interfaces. Architecture "molle".

One life = 1 git tag.



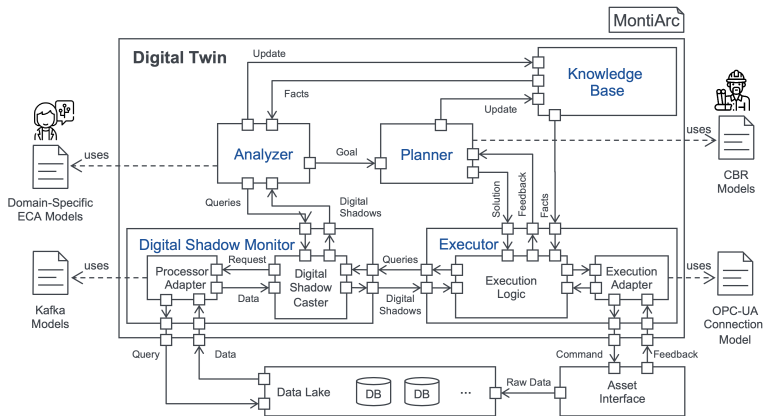


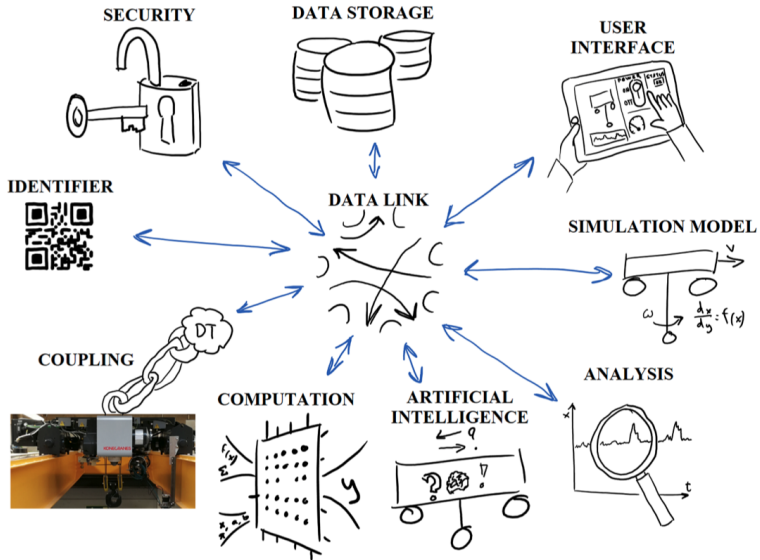
Vous fait penser à quoi ?



- The Core Twin monitors Devices
- Some Tools analyse from Data and produces Data
- Other Tools decide (and plan) changes (new configuration)
- Management executes the plan and reconfigure Devices

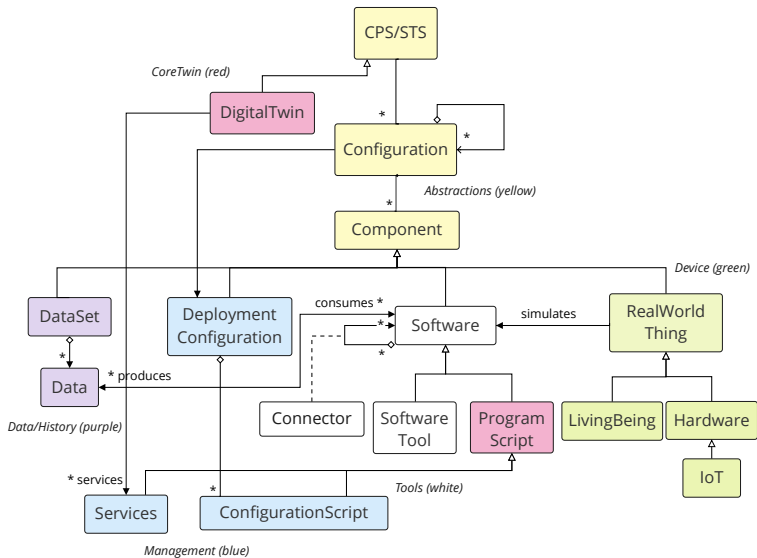
Modèle d'adaptation MAPE-K



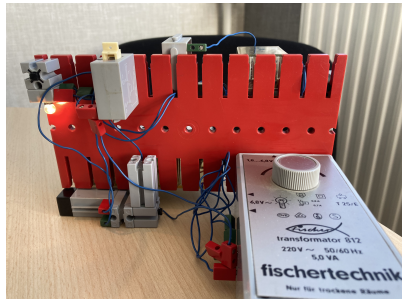
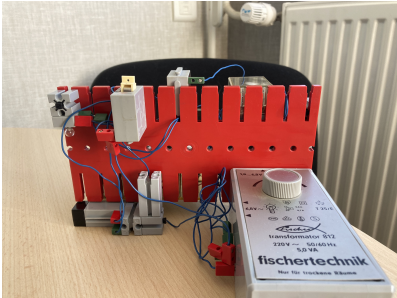


- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

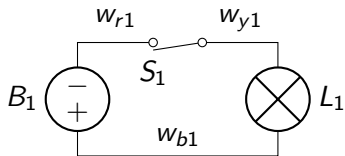
- TwinCoBot
 - Projet interCarnot avec les Mines d'Alès
 - Construction de jumeaux numériques
 - Dans un logement domotique
 - D'un bras articulé en interaction avec un humain
- CPER Industrie du Futur
 - Avec UBS, UBO, ENSTA Bretagne
 - Plateforme de développement et déploiement de JN
- Chaire Industrie du Futur
 - Coopération avec University of South Australia, Adélaïde



- S_0 , un système simple
- S_1 , un modèle numérique du système simple
- $S_0 + S_1$, l'association des 2 systèmes
- C , un ensemble de variantes de connexions



Les éléments : B_0 , S_0 , L_0 , w_{r0} , w_{b0} , w_{y0}



Les éléments : $B_1, S_1, L_1, w_{r1}, w_{b1}, w_{y1}$

A-t-on besoin des positions, des couleurs, des poids, des formes, etc. ?
On peut avoir un modèle totalement abstrait, ou un modèle avec des contraintes (couleur choisie, forme contrainte, etc.)



Modélisation multi-niveaux

Pas de connexion !

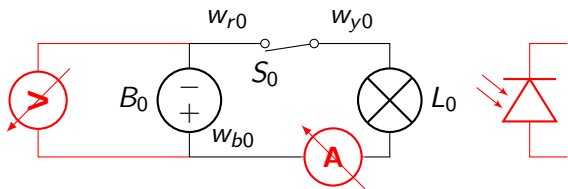
Pas d'interaction !

Des intentions :

- Connaitre l'état de B_0 (état = use/unuse, #¹use/unuse, durée, e, cumul (mWh), etc.)
- Connaitre l'état de S_0 (état = open/close, #open/close, broken)
- Contrôler S_0
- Connaitre l'état de L_0 (état = on/off, #on/off, durée on/off, broken, color, intensity)
- Collecter des statistiques, ...

A-t-on besoin des positions, des couleurs, des poids, des formes, etc. ?
(Un JN est toujours approximatif; on l'enrichie, le fait évoluer)

1. number of...




2 catégories

- intrusif
- non-intrusif

Dans notre exemple :

- ampèremètre : intrusif
- cellule photoélectrique : non-intrusif
- interrupteur intelligent (IoT) : intrusif [sauf si prévu dans le système initial]
- caméra : non-intrusif
- voltmètre : intrusif?

 Hypothèse les capteurs fournissent une API.

La question de la frontière ne se pose pas. $S = S_0 \cup S_1 \cup C$; S_0 est altéré.

- (A) ampèremètre
 - getIntensity() (offered service)
 - ↑use ↑unuse (output events)
- (C) cellule photoélectrique
 - getLumens()
 - ↑on ↑off
- (S) interrupteur intelligent (IoT)
 - doOpen() doClose()
 - ↑open ↑close
- (V) voltmètre
 - getVolt()
- (K) caméra (avec de la magie)
 - ↑open ↑close
 - ↑on ↑off
 - getLumens()
 - ↑disconnected (yes, can detect if wires are)
- caméra (sans magie)
 - ↑ videostream

État des composants

- B_0 : e instantanée (mA), cumulée (mWh), durée *use*
- S_0 : open, close, #open, #close, broken
- L_0 : on, off, #on, #off, broken
- w_* connectés ou non

Invariants

- inv σ : $|\#open - \#close| \leq 1$
- inv λ : $|\#on - \#off| \leq 1$
- inv α_{on} : $\#open = \#on$
- inv α_{off} : $\#closed = \#off$
- inv μ_{on} : on si $\#open > \#close$
- inv μ_{off} : off si $\#close > \#open$
- inv ϵ : on si $e > 0$ et off si $e = 0$
- inv ν : $v > v_{min}$

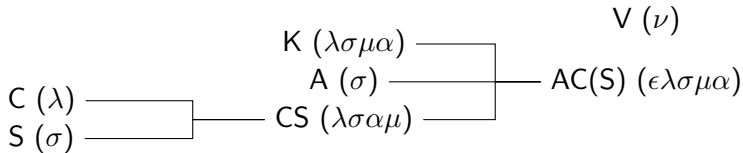
Un invariant violé : quelque chose est en panne ! A-t-on assez d'information pour décider quoi ?

- Alertes sur les invariants
- on, #on, #on before failure
- close, #close, #close before failure
- #wire disconnected (r,y,b)
- total e consommée
- tension batterie ν
- lumens produits (?)

DéTECTABILITÉ DES VIOLATIONS D'INVARIANT :

Inv.	V	A	C	S	K	AC	CS	ACS
ν	d	N	N	N	N	N	N	N
σ	N	i	i	d	d	i	d	d
λ	N	i	d	i	d	d	d	d
α	N	i	i	i	d	d	d	d
μ	N	i	i	i	d	d	d	d
ϵ	N	N	N	N	N	d	i	d

DéTECTABILITÉ/PRÉCISION



! Couleurs, positions, autres... pas nécessairement détectés

- Ajouter des éléments côté (1)
- Observer les comportements côté (1)
- Faire évoluer côté (0)

Exemple : ajout d'une K_1 ; On développe un logiciel de détection de lumière. . . pour détecter que L_1 s'allume quand on appuie sur S_1 (ou S_0^2). Le jumeau numérique est quoi ? une L_2 ?

2. dans ce cas il y a un autre détecteur pour que L_1 s'active.

- deploy configuration X (S_1) in context Y (S_0)
- run/stop ; lance une vie pour collecter des données dans S_1 ³
- replay life L [with acceleration] –
- apply tool on life X result in Y with metadata Z – analyse
- clone configuration X (S_1) in environment Y and run [with options]
- browse lives, tools, configurations, environments, models

Quelles données et métadonnées pour gérer un JN ?

3. La collecte de données coté S_0 peut exister indépendamment

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

Monsieur Jourdain

Par ma foi ! il y a plus de quarante ans que je dis de la prose sans que j'en susse rien, et je vous suis le plus obligé du monde de m'avoir appris cela.

Ferions-nous des jumeaux numériques depuis plus de 60 ans sans que nous en sussions rien ?

Oui, si le JN est un outil abstrait

Non, si le jumeau numérique est une entité logicielle avec interface, contrat d'usage, architecture. . .

- Ex : *clone*, *setEnv*, *setSimulData*, *simulate*, . . .

Un jumeau numérique

Est un système qui offre un service de *clonage*, qui retourne une instance numérique simulable

Un jumeau numérique

Est un système qui offre un service de *clonage*, qui retourne une instance numérique simulable dans un contexte (environnement et configuration de simulation) qui sera défini. Les informations produites pourront être exploitées (analyse, apprentissage, optimisation) [par des outils ou des humains] et permettront des prises de décision [par des outils ou des humains] de reconfiguration [par des outils ou des humains] du système d'origine.



Le jumeau numérique est un truc polymorphe, compliqué.

From Digital Twin Consortium definition to...

A digital twin is *a system connected to* real-world entities and processes, *synchronized at a specified frequency and fidelity*.

Digital twin systems *transform business by accelerating holistic understanding, optimal decision-making, and effective action*.

Digital twins are *motivated* by outcomes, *tailored* to use cases, powered by integration, built on data, guided by domain knowledge, and implemented in IT/OT systems.

Digital twins use *real-time and historical data* to represent *the past and present and simulate predicted futures*. *To do so, Digital Twin offers services for cloning, configuring the systems, and simulating and replaying different lives of the real-world entities.*

Plus largement, une méthode de construction de système(s) informatique(s).

- Qui a *besoin* de DigitalTwins pour quoi faire ?
 - Quelle(s) variante(s) ? couverture, contenu, reconfigurabilité, ... ?
- Quelles sont les *propriétés* conceptuelle d'un Digital Twin ?
- Comment sont *conçus* les Digital Twins ?
 - Comment se composent-ils
 - Comment intégrer des choses non-technologiques (humains, bâtiments, environnement, eau, etc.)
- Comment sont *déployés* les Digital Twins ?
- Comment sont *opérés* les Digital Twins ?
- Comment sont *validés* les Digital Twins ?
- Comment sont *évalués* les Digital Twins ?
- Est-il raisonnable de développer *massivement* les Digital Twins ? (Projet Gemini)

- 1 Qu'est-ce qu'est un jumeau numérique
 - Définitions
 - Usages
- 2 Les structures d'un jumeau numérique
 - Big picture
 - Vue macroscopique
- 3 Les propriétés d'un jumeau numérique
 - Propriétés
 - Architectures
- 4 Des travaux en cours
- 5 Conclusion
- 6 Perspectives

La fédération semble une approche naturelle pour gérer ces ensembles de modèles.

Il faudrait développer des TA (Technological Adapter) pour divers modèles :

- BIM
- CAD
- Unity
- Modèles IoT ? Foisonne !

Quelques questions ouvertes :

- Quelle liste de services ? ([SAP](#), [Azure](#))
- Quelle architecture logicielle ?
 - Le JN est vide, miroir ou complet ?
 - Embarque-t-il son environnement ? (simulable en autonomie ?)
 - Quelle reconfigurabilité ? M-K ; MA-K ; MAP(e)-K ; MAPE-K
- Quel processus de développement (et d'évolution) ?
 - Les implantations peuvent être très variées. . . (technologies, protocoles, outils, etc.)
- Comment composer des JN (verticalement, horizontalement) ?

- Architecture
 - T//S ; T>S ; T<S ; T<>S ; T
 - proxy (D/B) ; mirror (D/B) ; mixte (par composition)
- Purpose
 - Monitor, Contrôle, Simule, Analyse
- Management
 - Data/metadata ; configuration ; contexte
 - Tools (ML, AI, git, temporal DB, deployment)
- Special Aspect
 - Security, privacy, safety, reliability, resilience, autonomy, dots

Grille de clasification.

- Définir un métamodèle de JN
- Spécifier les services (clone, connecte, deploy etc.) [des DSL ?]
- Définir des processus de « JumeauNumérisation » de l'existant (à la IMRI - Improvement of Match between Resource and Information)
- Développer un dépôt (portail) de JN avec catalogue et classification
- Fédérer (en intégrant de nombreux adaptateurs à Openflexo)