The Digital Twin Paradigm: the bridge to the real smart Cyber-Physical systems implementation

Matteo Gabellini { matteo.gabellini2@studio.unibo.it }

Final Presentation for Autonomous Systems Course

Dipartimento di Informatica, Scienza e Ingegneria—Università di Bologna

June 20, 2019

Talk Outline I

1. Brief introduction to Industry 4.0 and CPS system
2. From Simulation to Digital Twins
3. What is a Digital Twin
4. Digital Twins in Healthcare
5. How to model elements and systems with Digital Twins
6. Examples of implementations
7. About Autonomy and Digital twins

Talk Outline II

8 Conclusions

Next in Line...

1. Brief introduction to Industry 4.0 and CPS system
    - Industry 4.0
    - IoT, Big Data & CPS
    - Systems Life-cycle

Industry 4.0

Born in 2011 in Germany in the context of its High-tech strategy

Main idea

The Industry 4.0 is the new form of industry where information and communication technology and automation technology are fully integrated.[12]

Today in the industry, embedded systems, processing unit and software are already used (e.g. SCADA Systems, Robotics, ERP ecc...), but:

- its usage is limited to the automatic management of some limited process and business analysis.
- There is a lack of a real integration between different sectors of the same industry

Industry 4.0

The Industry 4.0 exploit the power of new technologies like:

- Cloud
- Augmented reality
- Industrial IoT
- Autonomous robots/systems
- Cybersecurity
- Simulation modelling
- Additive manufacturing
- Big Data and analytics

Industry 4.0

This technologies enable the creation of:

- New Autonomous Systems [13];
- New Production strategies [12][10][18];
- New Business model more and more oriented to the "as a Service" model [3][5]

IoT, Industrial IoT and CPS

The Industry involves processes that operates and interact with the real physical world.

- *Network of sensors* are used in order to measure physical quantities.
- This sensors in the recent years evolved and became part of the so called Internet of Things (IoT)
- Which, in the specific context of the industry usage, take the name of *Industrial IoT*

A big opportunity:

IoT and the latest IT technologies (such as Cloud Computing, Big Data ecc...) create the opportunity to develop new complex system that integrate the Cyber and the Physical world.

- This systems take the name of Cyber-Physical Systems (CPS)

Principal benefits of IoT and Big Data

IoT permits to significantly enhance [17]

- automation
- accuracy
- efficiency
- productivity

lead to create precision systems [5].

Precision Systems

- Precision Machines and Service: Improve quality of Service, Reduce cost of service, New Business models, Lower consumables cost, Higher quality product, Improved health, Improved safety

Principal benefits of IoT and Big Data

This is possible because IoT technologies

- permit to collect real-time data directly from physical entity to which is attached
- enable things to collaborate and communicate each other

So a physical product could be *monitored*, *controlled* and *upgraded remotely*.[17]

Principal benefits of IoT and Big Data

The enormous amount of data generated from IoT devices presents 4 important characteristics:

- high volume: The amount of data generated is very high (order of terabyte, petabyte, ecc..)
- high velocity: The data are produced a higher rate than traditional systems
- variety: data are extremely heterogeneous in the format and in the way they represent information (format can be structured, semistructured and unstructured)
- veracity: the quality level of the data is also heterogeneous

This types of data sets are called Big Data.

Principal benefits of IoT and Big Data

- The Big Data analytics elaborate this data in order to extract useful information [17] could permit to take data-driven decision.
- For example analyzing a lot of data relative to the evolution a certain physics quantity measured by a sensor, is possible to *understand patterns, predict faults, detect anomalies.* [6]

Systems Life-cycle

A lot of human creations especially in engineering (eg. systems itself, product manufacturing, building construction, process management, production line ecc...) have a life-cycle that is composed by multiple phases [4] [8] such as:

- design/creation
- production
- operation
- disposal

Systems Life-cycle

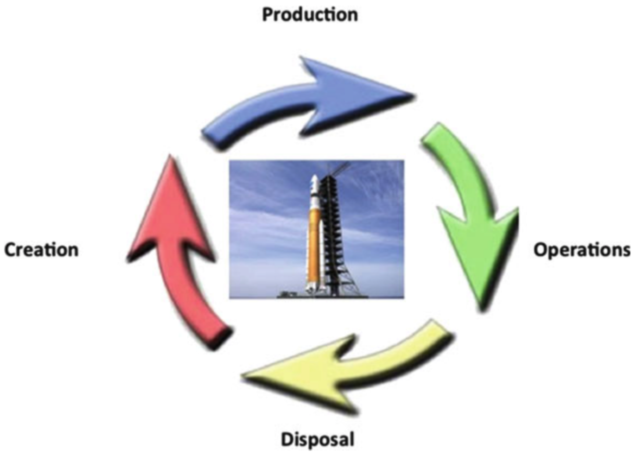


Figure: Product life cycle example from [8]

Systems Life-cycle

Often in each phase are used different technologies and techniques in order to implement it [14]. In each phase different information are generated.

- Some of this techniques/information are used only in some phases, also if it use in other phases could create new opportunity.[10]
- The use of different techniques and modelling strategies can bring to the loosing of the overall picture, especially in the case of complex systems.[14]
    - Because between the IT resources and techniques used in the various phases (and often in the same phase), true integration is often lacking, which if present could improve the entire life cycle.

Next in Line...

2. From Simulation to Digital Twins
    - Needs that bring to the development of digital twins

System Life-cycle - Quick look to simulation usage

The first example of a technology used only some phases is the simulation.

The case of Simulation [10][7][14]

To date simulation is only used during the design/production process

- in order to test and certificate product (e.g a vehicle, a manufacturing product, ecc...) characteristics
- according to some standard parameters, statistics, similitude or heuristics about the future condition that the product will be subjected (e.g physical forces, loadings, usage, ecc...).

System Life-cycle - Quick look to simulation usage

Lack of reuse and generalization

Most simulation application solve one specific application, changing only one aspect could require to set up an entire new simulation application. [14]

Multi-domain aspect

Often for a single product will be run independent simulation for different domain and several approaches is used.

System Life-cycle - Quick look to simulation usage

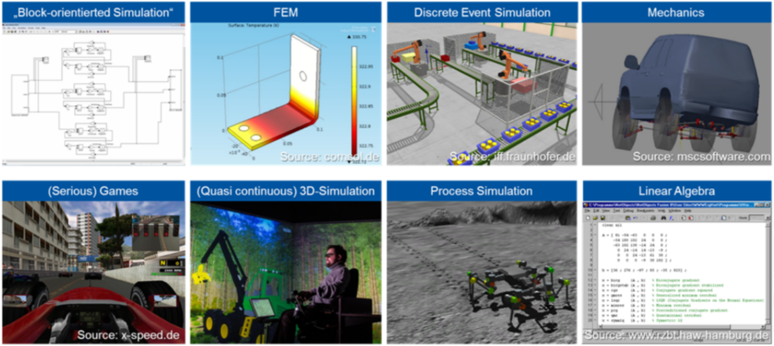


Figure: Different simulation approaches examples from [14]

The use of simulation in the Operation Phase - The NASA use case

Such as explained also in the NASA's paper [7]

- The simulation could be very useful also in the operation phase in order to:
  - predict *system evolution*.
  - predict *system failure*

  $\Longrightarrow$ so to improve the overall *health analyses and maintenance activity*.
- This could be possible also thanks to use of IoT, wich allows to **integrate simulation directly with real-time data** collected during the normal product *operation phase*.
  - This could enable also new certification modality, not based only on heuristics and statistics, but directly on real data collected with sensors.

The use of simulation in the Operation Phase

The use of simulation in operation phase could also:

- enable to create situation aware systems
  - where system controller is based on the concept of Simulation-based Control [14]
- lead to an optimization of the product itself
  - not only based on history from previous versions of the same product, but integrated with the current sensed data, which could materialize in a real-time ad-hoc optimization and support decision making [10].

Lack of a general model

So could be identified the necessity of a *new approach and paradigm* that:

1. support the creation of a common general conceptual model that be multi-disciplinary (computer science, mechanics, physics, ecc...)
2. can be used to encapsulate multi-dimension simulations
3. can be used to manage a product/system evolution during its life-cycle
4. integrate IoT management and oriented to the Big Data analytics
5. so usable in order to monitor a system, support health analyses and maintenance
6. support the decision making, previsions and management of events, so enable the creation of intelligent systems

A possible response

In order to satisfy the previous necessity in the scientific world in the last years appears a new concept

- **The Digital Twin Paradigm**

Next in Line...

3. What is a Digital Twin
   - Definitions
   - its role in the Industry 4.0
   - Enabling technologies

Not only one definition

- Such reported in the survey [10], scientific literature **not provide a unique definition** of this concept.
- But we can find different interpretations that put the focus on different aspects, often depending on the field where it was formulated
  - e.g. Aeronautics and Space, Robotics, Manufacturing, Computer Science, ecc...

Not only one definition

- The digital twin concept was born in a specific field (aerospace) for a specific intent [1].
- But when engineers from other fields met this concept, they identify new opportunity and potential uses in their respective fields

  $\Longrightarrow$ so they evolved the concept generalizing it, expanding or adapting it to their fields.

So today is difficult to define a *clear and precise boundary* to the concept of Digital Twin.

- *Subsequently are exposed some definitions from which the key elements will be extracted.*

[1] explained in next slides

The born of Digital Twin concept

- The concept of digital twin was born in the Aerospace field, precisely at NASA
- The first draft of the concept of Digital Twin appeared in 2010 in the NASA's Draft Modeling, simulation, information technology & processing roadmap [15]
- and subsequently finalized in 2012 in [16] and [7]
- it is one of the most commonly used as affirmed in [17]

For NASA a Digital Twin is: (1/2)

As reported in [15, 16, 7].

- "A Digital Twin is an integrated multi-physics, multi-scale, probabilistic simulation of an as-built vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., **to mirror the life** of its corresponding flying twin."
- "The Digital Twin is ultra-realistic and *may consider one or more important and interdependent vehicle systems*[...]"
- "In addition to the backbone of high-fidelity physical models of the as-built structure, the Digital Twin integrates sensor data from the vehicle's on-board integrated vehicle health management (IVHM) system, maintenance history and all available historical and fleet data obtained using data mining and text mining."

For NASA a Digital Twin is: [15, 16, 7] (2/2)

- "The Digital Twin can also predict system response to safety critical events and uncover previously unknown issues before they become critical by **comparing predicted and actual responses**."
- "Finally, the systems on board the Digital Twin are capable of mitigating damage or degradation by activating **self-healing mechanisms or by recommending changes** in mission profile to decrease loadings thereby increasing both the life span and the probability of mission success."

Other Definitions (1/3)

Other definitions that we can find in the scientific literature are:

- "a 'digital twin' is a virtual representation of a real world subject (person, software system,...) or real world object (machine, component, part of the environment,...). A digital twin **contains models** of its "data" (geometry, structure, ...), its functionality (data processing, behavior,..) and its communication interfaces." [14]
- "The concept of Digital Twin posits that the flow of data, process and decision is captured in a software avatar that mimics the operation or offers, at the least, a digital proxy." [6]

Other Definitions (2/3)

Some of others interesting definitions reported in [10] are:

- "A cradle-to-grave model of an aircraft structure's ability to meet mission requirements, including submodels of the electronics, the flight controls, the propulsion system, and other subsystems"
- "Coupled model of the real machine that operates in the cloud platform and simulates the health condition with an integrated knowledge from both data driven analytical algorithms as well as other available physical knowledge"
- "Very realistic models of the process current state and its behavior in interaction with the environment in the real world"
- "Product digital counterpart of a physical product"

Other Definitions(3/3)

And in [10] we can find also:

- "Virtual substitutes of real world objects consisting of virtual representations and communication capabilities making up smart objects acting as intelligent nodes inside the internet of things and services"
- "Digital representation of a real world object with focus on the object itself" [4]
- "'A unified system model that can coordinate' architecture, mechanical, electrical, software, verification, and other discipline-specific models across the system lifecycle, federating models in multiple vendor tools and configuration-controlled repositories"

Extract Key Characteristics

From the previous definitions is possible to summarise some key characteristics of a Digital Twin:

1. Is an Avatar of a real world entity, that contains model of the structure, functionality and behaviour
2. Is focused on the entity itself
3. Usable to model the system evolution from cradle-to-grave
4. Updated in real-time with the current status of the real counterpart
5. Integrated with historical data from other same physical entities
6. Can constitute a unified system model for different disciplines and at different level of abstraction (multi-physics, multi-scale)
7. Usable to simulate and predict system evolution and responses, also in order to mitigate damage or degradation and enable optimizations
8. Usable to making up smart objects acting as intelligent nodes

Extract Key Characteristics

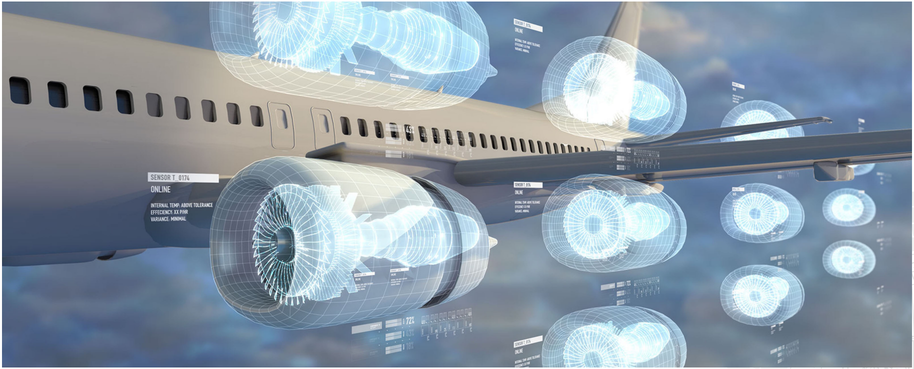


Figure: An illustration of a DT of an airplane turbine from [2]

Extract Key Characteristics

Not all of the previous characteristics must necessarily be present and/or present in the same way in order to talk about digital twin.
For example such as reported once again in [10]

- The lifecycle perspective is not shared by all papers. (the cradle-to-grave modelling)
- And Simulation seems to be a key point of Digital twin, but there is authors who do not mention it
- From some authors, DT is viewed as a model that represents the system that different types of simulations can be based upon. From others such as a simulation of the system itself.

Digital Twin as a bridge

Is interesting the vision reported in [17] where digital twin is viewed as a:

- *"bridge between the physical world and the digital world. Different from CAD (that exclusively focuses on the digital world) and IoT (that heavily concentrate on the physical world), DT is characterised by the two-way interactions between the digital and physical worlds, which can possibly lead to many benefits. On one hand, the physical product can be made more 'intelligent' to actively adjust its real-time behaviour according to the 'recommendations' made by the virtual product. On the other hand, the virtual product can be made more 'factual' to accurately reflect the real-world state of the physical product."*

Digital Twin as a bridge

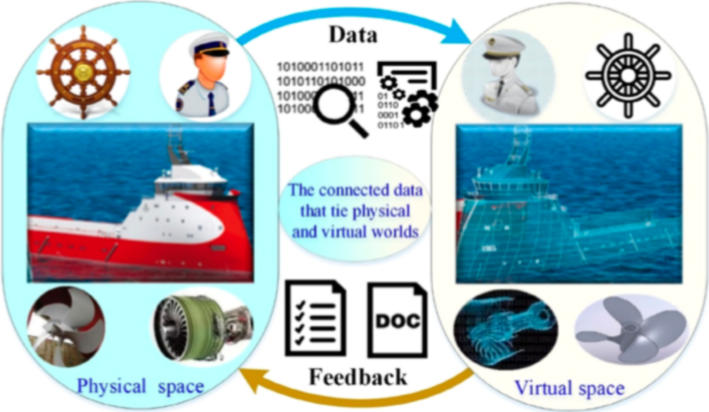


Figure: An illustration of the two-way interactions between the digital and physical worlds from [17]

Cyber Twin & Digital Twin

In the context of CPS system we can find the concept of *Cyber Twin*. About that the authors of [17] report:

- *"The notion of digital twin should be differentiated from cyber twin. Cyber twin is based on the conception of CPS. Generally speaking, the notion of cyber twin is derived based on the development of cyber physical system (CPS). Cyber space is defined as a networked space that consists of multiple interconnected cyber twins of critical objects. In other words, cyber twins purely exist and are merely meaningful within the cyber space. In sharp contrast, a complete DT must include a physical model, a virtual model, and connections between the physical and virtual models. In other words, it emphasises interaction, communication and collaboration between physical space and cyber space."*

Personal Consideration

Tao et al. [17] consider also that *"DT is composed by three parts: physical product, virtual product and the linkage between physical and virtual product"*, this concept is reported in a similar way also in [9]

- In my opinion say that also the physical entity is part of the digital twin *is a stretch*
- However from the previous two slides can be understood that the *interaction between cyber and real world and their linkage* has the same importance to have a correct model of the physical system.
  - so its design should not be underestimated
- Is also important find a way to associate uniquely a digital twin to its physical counterpart

The digital twin in the Industry 4.0

As can be inferred from the definitions of digital twin. This paradigm can be used also as a core element in Industry 4.0 development. [18]
Is possible to identify two macro-uses of the digital twin paradigm:

- In the develop of new smart product
- In the design and implementation of production systems

The digital twin in the Industry 4.0

- As a tool in order to develop new smart product [14, 10, 17], DT are used to:
  - Support health analyses
  - Digitally mirroring the life of a physical entity, so predict performance
  - improve the management during the lifecycle
  - support decision making and optimization
- As a tool in order to develop, manage and implement production system and production lines [18, 17], according to papers DT gives the opportunity to:
  - have a near real-time production control
  - support strategic planning and operation planning
  - have near real-time production optimization
  - have a comprehensive image of the production system
  - enable self-reconfiguration and self-adaptability between elements of the production line [13]

Enabling Technologies

After have understood what are digital twins, is possible to summarize the enabling technologies. As could be easily guess some of this technologies are the same of the Industry 4.0

- Cloud Computing
- IoT, embedded systems, sensors and actuators
- Simulation
- Physics models
- Cyber-Security
- Big Data
- Machine Learning

Next in Line. . .

4 Digital Twins in Healthcare

Opportunities and evaluations about the use of digital twins in the healthcare field

Moving to Healthcare field, the use of DT could improve a lot the patients treatment and medical staff work. As reported also in [9], the use of DT could:

- For *Medical staff*:
  - Assist researchers do experiments and testing of medicines in virtual patients to reduce risks and costs
  - Medical experts or doctors do not need to see patient in person, because they can determine the symptoms and prescribe treatment plans based on various real-time dynamic data from DT models
  - Optimize treatment options through model iteration on a virtual platform
  - Doctors, when treat a clinical case, could not base only on personal experience, but could extend its knowledge with the data provided by the digital twins relative to other similar cases.

Opportunities and evaluations about the use of digital twins in the healthcare field

- For *medical device organizations*, DT could be used:
  - to do real-time monitoring, structural life prediction, and devices management of the medical devices.
- For *Patients*, DT can:
  - do real-time monitoring of their physiological states through DT models
  - feedback their information data to patients or their informal caregivers in real-time

Opportunities and evaluations about the use of digital twins in the healthcare field

Considering the complexity (dynamic nature and uncertain properties) of current healthcare systems is very difficult to research behavior, patterns and laws with real scenarios.

- Modelling and simulation provide a good way to deal with healthcare systems [9]
- Also in this sector, the use of DT could bring to obtain a reliable overall picture of the healthcare system.

At the following link is reported a Philips's video example of the opportunity created by the use of digital twins in healthcare.

- `https://www.youtube.com/watch?v=H6JzPCbyVSM`

Next in Line. . .

5. How to model elements and systems with Digital Twins
   - Software system engineering point of view
   - Digital by design
   - Modularity and multi-level modelling in the Digital twin paradigm
   - A possible formal representation of a digital twin models

Notes

- *A lot of papers on digital twins are written by authors that came from different disciplines from Software Engineering.*

  $\Longrightarrow$ *They concentrate more on potential uses in their field rather than analyze and propose real implementations at the software level.*
- Today there isn't a standard implementation or way in order to create digital twins.
- some big companies, such us General Electrics and Philips, propose commercial solutions, but them aren't already became the "*standard de facto*".
- *the analysis is continued using a perspective more linked to software engineering and computer science*
  - In this view some *principles* will be reported and...
  - *some practice implementation examples are exposed*
  - **But today there isn't a prevalent practice approach**

From software system engineering point of view (1/4)

DT can be used to create fully functioning virtual prototypes during development, that allow efficient and goal-oriented development, test and verification of components and the entirely system itself. [14]

- So DT can be used as a *framework to develop the controller of the physical entity* (e.g. a robotic system) [14]
  - potentially using artificial intelligence techniques applied to event history from other similar DT.
- This enable the controller development concurrently to the physical systems production.

From software system engineering point of view (2/4)

DT could be useful also in order to project, develop and manage Smart City, IoT and complex system environment.

1. DT can be used as block components in order to model system elements at different level of abstraction [6, 3, 2]
    - a Smart City can be seen as a inter-dependent cascade of systems and networks
    - in the Industry context: *Process twin* informed by a set of *System or Unit twins*, where each is composed of several *Asset twins*, that are collection of *Component twins*

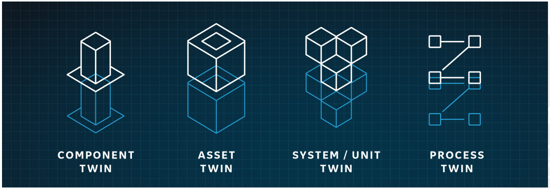


Figure: Digital Twins at different level of abstraction [2]

From software system engineering point of view (3/4)

2. DT enable the creation of services where physical things can exploit the computation and network resources of the Digital counterpart [3] (*DT increase the capacity of physical things*)
    - possibility to communicate with other things also if there isn't a direct physical link
    - possibility to enable sensors and services fusion in the digital twin when the physical counter part haven't enough computation power
    - possibility to select dynamically the computation location according to the context (e.g. execute some task on DT when the physical thing has low battery)

From software system engineering point of view (4/4)

3. DT enable also the possibility to create intelligent services that exploit spatio-temporal relations between physical entities, directly captured by their digital counterpart [4, 3] (e.g. the relation between a DT of a person and a DT of a car can capture when user drive the car)

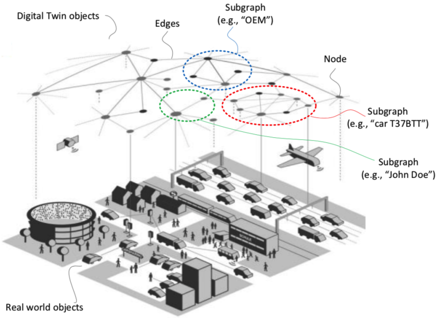


Figure: Spatio-temporal Relation between DT [4]

Which came first the chicken or the egg? [6]

The digital twin concept assumes that we are creating a digital duplicate of a physical entity.

- But often try to integrate digital technologies in a physical product after its development and production, could bring to a not optimized information technologies usage.
- In the industrial world, entity are created starting from an engineering plan, technical specs represented with CAD/CAM models. So today physical entity are conceived already as digital model before production.

Machine and Parts *Born Digital* [6]

"In the world of machines and parts, being conceived as digital is the design, to use a pre-natal metaphor. The physical entity is the post-natal stage, metaphorically."

Digital by design [6]

In order to facilitate the convergence to system that exploit the correct potentiality of IT sector:

Digital Conception - *Born Digital*

The development of product that want to use the *DT* paradigm must follow the **Digital-by-design** approach.

- Is necessary to think from the beginning at the digital counterpart of a product and not create it after the product manufacturing.

$\Longrightarrow$ **The development of the product must be driven by the DT and not vice-versa**

- This facilitate a more natural integration, because the aspect and needs of interaction through network with other entities is considered from the beginning.
- This approach is also one of the core part of the IoT metaphor

Digital twin driven product design pros.

According to [17], the use of DT in order to drive the development could:

- In task clarification phase, DT serves as an 'interpreter' to facilitate the translation process from customer need into functional requirements
- In conceptual design phase, DT serves to facilitate designers
  - *"to identify system contradictions within a product"*
  - *"to capture various uncertainties associated with a product"*
  - *"to identify and diagnose various complexities"*
  - *"to incorporate a variety of contextual information into concept generation"*

Exception that prove the rule [6]

Not all entities of the world are created by people.

- For example Humans, animals, plants, ecc....
- But is possible to attach to them devices in order to observe and monitor their state. (e.g. medical devices)

Digital Twins that not Born Digital

- Is possible to create a Digital Twin of this entities as a combined state of the devices attached to them.
- In this cases the Digital Twin is not referred as *born digital* [6]

The Blocks abstraction

- Almost every entity, natural or not, is composed by different sub-unit, that encapsulate different domain of semantics and ontologies.
- In [6] and also in [4], the authors propose to use the "block" abstraction in order to model digital twins.

Block abstraction

- The block constitute the baseline 'unit'
- Through this *blocks* is possible to represents objects, data, process and decisions (outcomes).
- Through the connections between this blocks is possible to synthesize entities or networks.
- The semantic and ontologies intra-unit is modelled by the unit, while the semantic and ontologies between different units is modelled directly through connection between units.

Association by function

In Enterprises using this approach is possible to model as block

- Things/objects, Software Agents
- work units, process
- authentication, authorization
- decisions, outliers, feedback
- security
- metrics
- dependencies

thinking as the enterprise ecosystem as a *operational sequence*, disassociated by modular structure and **associated by function**.[6]

Who will build these blocks?

As could be imagined, like in the IoT and connectivity world, to make the block approach feasible

- There may will not be any industry or company that will be the front-runner.
- there is the necessity of global alliances such as an academic leadership of industry-government consortia in partnership with global organizations or standardization bodies, that establish the tools for interoperability [6]
- even because this block approach "spanning and overlapping IoT, DT and blockchains (not only for financial transaction)" [6]

In addition to the principle of **Digital-By-Design**, is possible to add **Interoperability-by-Design** and **Connectivity-by-Design**.

Different Level of complexity [6]

- A Digital Twins of a granular units must be connected by their relationships to the relevant data feeds from sensors/gateways
- The relationships could be also **Bio-inspired**
- Similar to the block approach in IoT, "where a thing may be an integrated platform, synthesized from subunits or blocks of things."

How to manage complexity?

In order to manage and separate different level of semantics and ontologies could be useful to use the **"layer cakes" approach**, similar to the tcp/ip, where each layer is constructed on specific principles, concepts or ontological frameworks.

Interoperability between standards is required.

Example: The temperature sensor example [3]

- Imagine to have four smart temperature sensors deployed to four corners of a room.
- Their data are collected through its respective *digital twins*.
- Fusion data from this four *digital twins* is possible to create a *digital twin* of the room that provide an aggregate temperature value relative to the entire room.

Step of Digital Twin creation

From the different papers analyzed could be summarized the creation of a DT in different points:

- Creation of a common repository that capture the Physics entity information (e.g. geometric structure, ecc...)
- Creation of the model of sensors and actuators
- Creation of the model of the behaviour (functional factors, functional unit, control logic, ecc...)
- Establish a two-way real-time connection between the physical entity and the virtual counterpart
- Depending on the semantic level considered or the system engineer interpretation of DT definition, implements directly as part of the DT or as separated middleware or service:
  - the management of simulation
  - the management of data analytics and knowledge extraction
  - the management of system behaviour and models evolution, according to the analytics and A.I. algorithms outcome

Step of Digital Twin creation

- The previous points should not be considered as to do one after another, but could potentially done concurrently.
- The implementation of previous steps could be achieved through composition and association of one or more DT of a low level of abstraction [3, 2]
- Depending on the context, the implementation of simulation could require also a modelling of the environment where the physical counterpart of the digital twin live and operate [14].

A common repository

- The first step in order to create a Digital Twin is to implement a common repository that constitute virtual representation of the physical entities. [17]
- So is necessary to create a model that capture all interesting information relatively to the problem to be addressed.
- This repository should be easily expandable and upgradeable in order to follow the physical system evolution.

A Common repository II

1. If the entities is a manufacturing product or more generally a generic object:

   $\Longrightarrow$ This repository will contain CAD/CAM models, with physics of the material, and the engineering characteristics of its operational function.[6]

   - This model could be represented for example with SysML, Modelica, SolidWorks, 3DMAX, AutoCAD [9], MATLAB [14] or other modelling languages and techniques.
   - if one of the target of the digital twin is monitor also how the object behave when subjected to physical forces, is necessary to model the latter through a dedicated layer ("layer cake") from which the physical model of the component can inherit the physical laws [6]

A Common repository III

2. If the entities is a human, animal or living creature
    - According to [9] this repository could contain the information about: *age, sex, genetic and health record information*
    - Physical properties (physical information) are captured by sensors. The sensor model should be created and will include parameters *type index, alert and prompt*
    - if deemed appropriate, Digital twins of devices used to capture physiological data could be created and linked to the relative main living creature DT.

Where store models?

In the literature analyzed are exposed two way to store the model of digital twins:

- The first and more used is to store models in the cloud [3, 9, 1]
- The second way is in memory directly on the embedded things [13]

Virtual Testbeds

In [14], the authors cite *Virtual Testbeds*.

- It is a central method of *eRobotics* wich provide the necessary basis for developing new "Simulation based X"
- It enable engineers to examine the entire technical system (that authors consider as digital twin) in its environment.

Virtual Testbeds

- The DPS contains also the necessary algorithms for data processing and control.
- Finally, turning switches T_sense and T_act (see figure) is possible to take back to the real world to control the real system.

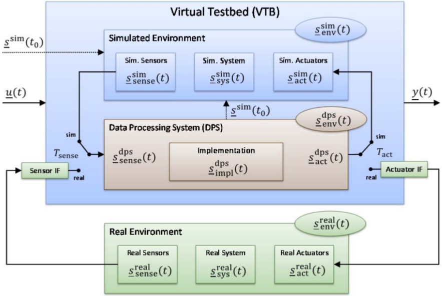


Figure: Systematic view on Virtual Testbeds [14]

Versatile Simulation Database

Starting from Virtual Testbeds method, Schluse et al. [14] present a practice implementation of a micro kernel software database called **Versatile Simulation Database (VSD)**

- It is usable as a elementary block to implements digital twins.
- "It provides the central building blocks for data management, meta information, communication, persistence, and user interaction"

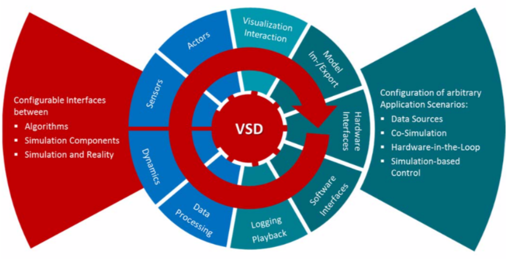


Figure: The micro kernel architecture from [14]

Versatile Simulation Database

- "Basically is an object-oriented real-time database holding a description of the underlying simulation model. Fully implemented in C++"
- is not only a data container, but contains also the algorithms and interfaces to manipulate the data.
- Provide a pub-sub messaging system and essential functionalities for parallel and distributed simulation
- "All simulation functionality is achieved by creating specialized plugins which build upon and interact with the VSD core"

Digital Twin as a FSM

In [3] the authors propose the use of block diagram and Moore's finite state machine, in order to model physical things, digital twins and also the modelling of hybrid things where the computation is executed in part on the physical layer and other in the cyber layer.

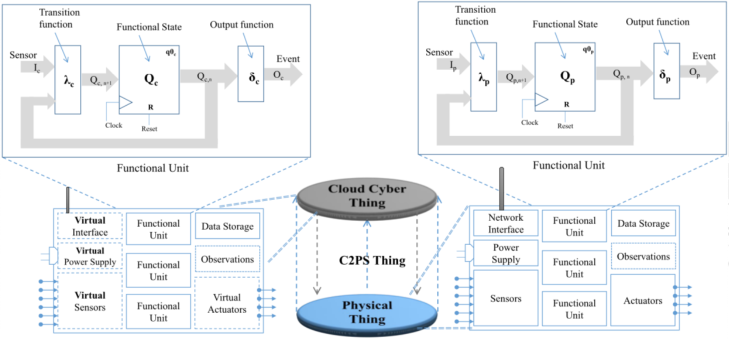


Figure: Architecture of the physical and cyber layer [3]

Digital Twin as a FSM - Physical Thing [3]

The Physical Things $p \in \mathbb{P}$ is comprised of $(S_p, A_p, F_p, E_p, N_p, P_p, D_p)$ where:

- S_p constitute the sensors that act as input to the system
- A_p constitute the actuators that perform action on the physical environment
- F_p represents at least a functional unit that process sensory values
- E_p represents events identified by the functional units
- N_p is at least one network interface
- P_p is the power supply
- D_p is data storage sets were results are stored

Digital Twin as a FSM - Cyber Thing [3]

The Cyber counterpart (Digital Twin) $c \in \mathbb{C}$ is comprised of $(S_c, A_c, F_c, E_c, N_c, P_c, D_c)$ where:

- S_c constitute the **virtual** sensors that act as input to the system
- A_c constitute the **virtual** actuators that perform action on the physical environment
- F_c represents at least a functional unit that process sensory values
- E_c represents observed events
- N_c represents virtual network interfaces
- P_c represents virtual the power supply
- D_c is data storage sets were results are stored

Digital Twin as a FSM - Functional Unit [3]

Every functional unit could be represented as a sequential finite state machine of 6-tuple

$$Physical \longrightarrow f_p = (Q_p, I_p, O_p, q0_p, \lambda_p, \delta_p)$$

$$Cyber \longrightarrow f_c = (Q_c, I_c, O_c, q0_c, \lambda_c, \delta_c)$$

where:

- Q represents various states
- $q0$ is the initial state
- I is the sensor values that act as inputs $I \subseteq S$
- O output associated to a state $O \subseteq E$
- λ is the transfer function $\lambda : Q$ x $I \longrightarrow Q$
- δ is the output function $\delta : Q \longrightarrow O$

Hybrid Things [3]

Using the same formalism exposed in previous slides is possible to represents *hybrid things* were part of computation is executed in the *physical layer* and other in the *cyber layer*.
The functional unit $f_h = (Q_h, I_h, O_h, q0_h, \lambda_h, \delta_h)$ could be described as a side-by-side synchronous composition of state machines, where:

- $Q_h = Q_p \times Q_c$
- $I_h = I_p \times I_c$
- $O_h = O_p \times O_c$
- $\lambda_h : (Q_p \times Q_c) \times (I_p \times I_c) \longrightarrow (Q_p \times Q_c)$
    - $\lambda_h((q_p, q_c), (i_p, i_c)) = (\lambda_p(q_p, i_p), \lambda_c(q_c, i_c))$
- $\delta_h : (Q_p \times Q_c)\ X\ (I_p \times I_c) \longrightarrow (Q_p \times Q_c)$
    - $\delta_h((q_p, q_c), (i_p, i_c)) = (\delta_p(q_p, i_p), \delta_c(q_c, i_c))$

Next in Line. . .

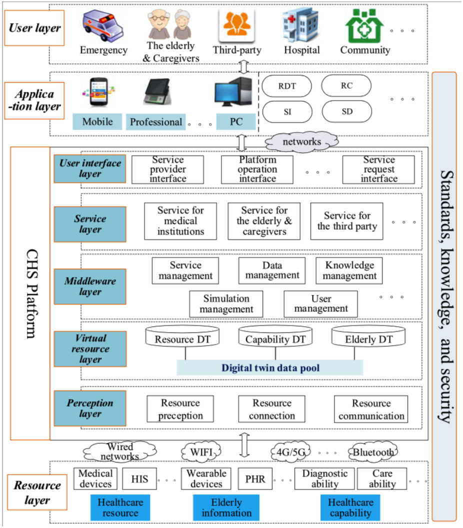
6 Examples of implementations

Note:

In the following slides are presented some schemas of architecture exposed in the different analyzed papers and used to create and manage digital twins.

An example of an architechture for Elderly Healhcare

In [9] authors presents a framework that uses Digital Twins for the Elderly Healthcare management



An example of an architechture for Elderly Healthcare

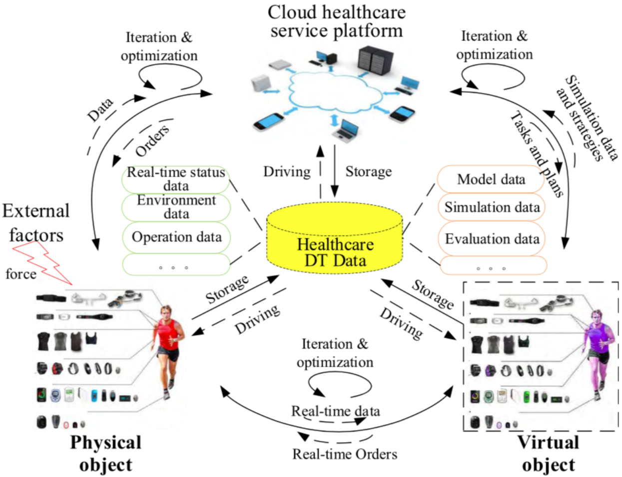


Figure: Reference model of a system that uses digital twin in healthcare from [9]

An example of an architechture for Elderly Healhcare

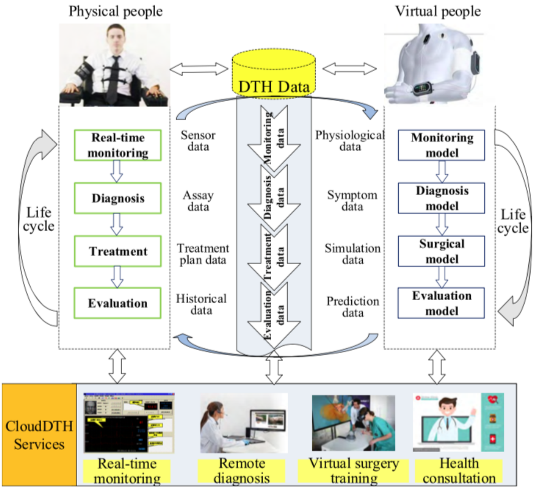


Figure: Personal Health management Lifecycle flow [9]

C2PS: Cloud based cyber-physical system architecture

Alam et al. [3] propose the following architecture in order to use digital twin to manage CPS systems.

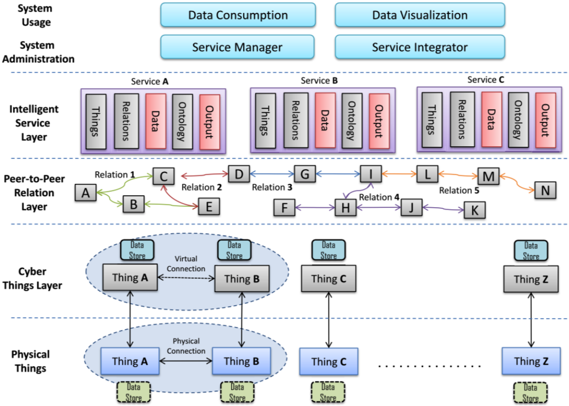


Figure: System architecture from [3]

GE's Digital Twins

GE provide a platform called *Predix* that among the many features provide the possibility to create and manage digital twins

IoT Platform.png

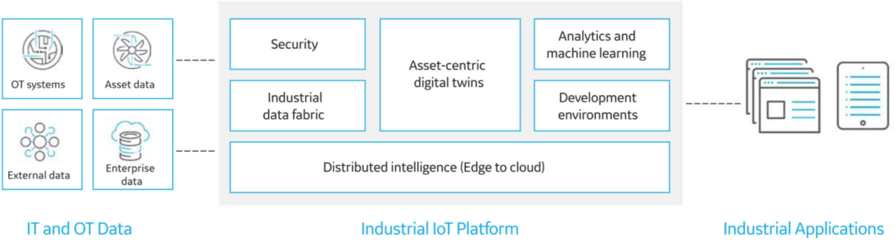


Figure: GE'S Predix Platform [1]

Digital Twin of a Turbine

At the following link is exposed a nice example of a digital twin of a turbine of an electric generator implemented trough *GE's Predix Platform*.

https://www.youtube.com/watch?v=2dCz3oL2rTw

Results very concise and interesting the definition of Digital Twin reported by the speaker in the video:

- The digital twin *"is a living model that drives business outcome"*

Next in Line...

7. About Autonomy and Digital twins
   - The digital twin as a tool
   - Emergence and Swarm intelligence

About Autonomy and Digital twins

One of the motivation that brings to the development of digital twins by NASA [7] is the necessities to find a way to implement "self-aware" vehicles capable to manage also unpredictable situation during the design and development.

- This require a certain degree of autonomy.

Autonomous Agents [11]

Autonomous Agents can behave in the real world without human intervention.

Autonomous Systems [13]

"Autonomous systems are able to execute high-level task specifications without explicitly being programmed"

The view of the world

As reported in [13], autonomous systems (/agents):

- "Use sensors to perceive the environment and the current situation"
- "Employ automated reasoning and planning to determine their course of action"
- "Use actuators to execute the determined action sequence in order to achieve the overall goals set"

More relevant information concerning the overall state has the system, more precise could be the reasoning of the system controller.

- Some of this information could be generated during design phase and not directly perceptible by the sensors or inferred from them. (e.g the geometric structure)

The digital twin as a key enabler of autonomy

The digital twins of the system could be used directly from the controller itself to have a more extended vision of the world, composed by:

- the current state of the world perceived through sensors
- Integrated with structural models and other information generated during design and production process
- Integrated also with historical information from other digital twins of similar products (or, more in general, entities)
- Enriched with information generated by the interaction with other cyber entity (digital twin, agents, services ecc...)

Proactiveness

As exposed before, using simulation with digital twins could be used to *"anticipate the consequences of actions [...] in a given situation"* [13]

- This open the door to the implementations of not only Reactive, but also **Proactive Behaviors**

Cognitive aspects

- Digital Twin could be used to model entities to different level of semantic levels and enable the possibility to model ontologies.
- Considering also its nature of strong coupling with the twin physical entity

$\Longrightarrow$ Digital Twins could be used as base block in order to develop the next generation of Cognitive computing application

Integrating specific sector models (e.g. CAD), facilitate the interaction between users (not only humans) of different sectors

$\Longrightarrow$ because the coupling between the different models is maintained directly by the Digital Twin itself

Emergence

Emergent Behavior [11]

"One of the fascinating features of autonomous agents is that they exhibit so-called *emergent* behaviors, that is, behaviors not programmed into agents by designer"

In this way through digital twins is possible to think at systems that:

- using simulation and artificial intelligence algorithms

$\Longrightarrow$ can learn directly from different simulated scenarios some ad-hoc behaviour

- and tuning and evolving it according the changes at the world model stored in the digital twin.

$\Longrightarrow$ so we could be capable to manage also unpredictable situation during the design and development. [7]

The shift from centralized to local view

Using digital twins is also possible to think to reach a specific (and maybe complex) macro behaviour of a system, developing only the basic rule of interaction and relative evolution between the sub-entities.

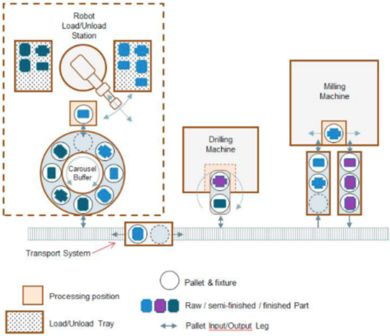
- In this case instead of having a centralized controller that manage all the system part, some behavior is **emergent from the interaction between the parts**.

$\Longrightarrow$ like in swarm intelligence.[6]

Example of Autonomy in a production line [13]

In [13] is exposed an example relative to a Cyber-Physical Production System (CPPS). Are considered 4 production unit:

- A robot load and unload station
- A CNC drilling machine
- A CNC milling machine
- A Transport system



Example of Autonomy in a CPPS [13]

- Each *production unit* have a Digital Twin of itself
- Each *product* in production have a Digital Twin
- The part flow is orchestrated autonomously through a process of negotiation
- The production unit **detect which is the next manufacturing operation to do on the part, watching directly at the digital twin state of the product**.
- The production unit request to other machines (potentially also to itself) which could resolve the job.
- The production unit that wins the order, take the part to do the next operation.

Example of Autonomy in a CPPS [13]

Production Flow:

1. *"A pallet is taken out of a tray and fixed onto a pallet at the load/unload station."*
2. *"The pallet carrying the raw part is first brought to buffer"*
3. *"When the milling machine is available, the pallet is transported to its input leg."*
4. *"After the part has been milled, its Digital Twin is updated."* The part may be brought to the *carousel buffer* or wait in the *output leg* *depending* on the space necessities.
5. *"Once the drilling machine becomes available the pallet is brought there for processing"*
6. *"After the part has been drilled, its Digital Twin is updated and the pallet brought to the buffer"*
7. From the buffer, the part is brought to the load/unload station where it is removed from the pallet and placed in a tray.

Example Fault Handling [13]

Considering the precedence CPPS example

- assuming that the milling machine has the skill to *mill*, *drill* and *buffer pallets*.
- consider a breakdown of the *carousel buffer*
- This normally could cause a deadlock because the finished part in drilling machine has nowhere to go.
- *"Since the milling machine has buffering skill, the transport system assign temporarily to be the system pallet buffer in place of the carousel buffer"*
- *"In this way, The CPPS can continue production until machining of all the parts is finished, before which time the carousel buffer should have been repaired."*

Next in Line. . .

8 Conclusions

Summary

In the previous slides the concept of digital twin has been analyzed extracting the general principles, starting from its definition up to take a look of some practical implementation

- Unfortunately other research is necessary before reach a solid practice foundation.
- Even because a lot of aspects overlap other different computer science aspects and technologies, in primis the IoT
- But from the principles that have been extracted it is possible to understand the potential of this powerful paradigm, which goes beyond the simple satisfaction of the needs that led to its development in the context of NASA

Conclusions

Finally is possible to conclude that Digital twin constitute the bridge

- to the creation of the new generation precision systems, and to the full exploit of the IoT Technologies potentialities.
- to the implementation of dynamic systems where autonomous software agent could operate and human can easily analyze and verify the evolution using directly **sector specific model** (potentially also formal models).
- to cognitive application implementation, thanks to the capacity of digital twin to model **ontological, semantic and dynamic aspects**.

$\Longrightarrow$ so *to the real smart Cyber-Physical systems implementation.*

References I

[1] The digital twin: Compressing time-to-value for digital industrial companies.
https://www.ge.com/digital/sites/default/files/download_assets/The-Digital-Twin_Compressing-Time-to-Value-for-Digital-Industrial-Companies.pdf.

[2] Industrial digital twins: Real products driving $1b in loss avoidance.
https://www.ge.com/digital/blog/industrial-digital-twins-real-products-driving-1b-loss-avoidance.
Accessed: 2019-06-06.

[3] Kazi Masudul Alam and Abdulmotaleb El Saddik.
C2ps: A digital twin architecture reference model for the cloud-based cyber-physical systems.
IEEE Access, 5:2050–2062, 2017.

[4] Arquimedes Canedo.
Industrial iot lifecycle via digital twins.
In *2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*, pages 1–1. IEEE, 2016.

References II

[5] Timothy Chou.
Precision-Principles, Practices and Solutions for the Internet of Things.
CrowdStory, 2016.

[6] Shoumen Palit Austin Datta.
Emergence of digital twins-is this the march of reason?
Journal of Innovation Management, 5(3):14–34, 2017.

[7] Edward Glaessgen and David Stargel.
The digital twin paradigm for future nasa and us air force vehicles.
In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA*, page 1818, 2012.

[8] Michael Grieves and John Vickers.
Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems.
In *Transdisciplinary perspectives on complex systems*, pages 85–113. Springer, 2017.

[9] Ying Liu, Lin Zhang, Yuan Yang, Longfei Zhou, Lei Ren, Fei Wang, Rong Liu, Zhibo Pang, and M Jamal Deen.
A novel cloud-based framework for the elderly healthcare services using digital twin.
IEEE Access, 7:49088–49101, 2019.

References III

[10] Elisa Negri, Luca Fumagalli, and Marco Macchi.
A review of the roles of digital twin in cps-based production systems.
Procedia Manufacturing, 11:939–948, 2017.

[11] Rolf Pfeifer and Christian Scheier.
Understanding intelligence.
MIT press, 2001.

[12] Blaž Rodič.
Industry 4.0 and the new simulation modelling paradigm.
Organizacija, 50(3):193–207, 2017.

[13] Roland Rosen, Georg Von Wichert, George Lo, and Kurt D Bettenhausen.
About the importance of autonomy and digital twins for the future of manufacturing.
IFAC-PapersOnLine, 48(3):567–572, 2015.

[14] Michael Schluse and Juergen Rossmann.
From simulation to experimentable digital twins: Simulation-based development and operation of complex technical systems.
In *2016 IEEE International Symposium on Systems Engineering (ISSE)*, pages 1–6. IEEE, 2016.

References IV

[15] Mike Shafto, Mike Conroy, Rich Doyle, Ed Glaessgen, Chris Kemp, Jacqueline LeMoigne, and Lui Wang.
Draft modeling, simulation, information technology & processing roadmap.
Technology Area, 11, 2010.

[16] Mike Shafto, Mike Conroy, Rich Doyle, Ed Glaessgen, Chris Kemp, Jacqueline LeMoigne, and Lui Wang.
Modeling, simulation, information technology & processing roadmap.
National Aeronautics and Space Administration, 2012.

[17] Fei Tao, Fangyuan Sui, Ang Liu, Qinglin Qi, Meng Zhang, Boyang Song, Zirong Guo, Stephen C-Y Lu, and AYC Nee.
Digital twin-driven product design framework.
International Journal of Production Research, pages 1–19, 2018.

[18] Thomas H-J Uhlemann, Christian Lehmann, and Rolf Steinhilper.
The digital twin: Realizing the cyber-physical production system for industry 4.0.
Procedia Cirp, 61:335–340, 2017.

The Digital Twin Paradigm: the bridge to the real smart Cyber-Physical systems implementation

Matteo Gabellini { matteo.gabellini2@studio.unibo.it }

Final Presentation for Autonomous Systems Course
Dipartimento di Informatica, Scienza e Ingegneria—Università di Bologna

June 20, 2019