An introduction to Model-Driven Engineering (MDE)

Dipartimento di Matematica e Informatica,
Universita’ di Firenze

mohamad.gharib@unifi.it

Outline

- What is Model-Driven Engineering?
- Why Model-Driven Engineering?
- Disadvantages of Model-Driven Engineering
- Modeling and models
    - What do we mean by modeling?
    - What is a model?

What is Model-Driven Engineering?

Figure: Difference between a model and a system

A model could be: a small scale version of a system, software simulation, 3D design, UML/SysML design, etc.

Usually, **Model** is not equivalent to **System**

What is Model-Driven Engineering?

- *MDE* is an approach to software development, where **models** rather than **programs** (code) are the principal outputs of the **development process**.
- *MDE* is a way of documenting and sharing **knowledge** depending on **models.**
- *MDE* is a fairly new area of research (less than 30 years!).

Why Model-Driven Engineering?

- Software systems are getting increasingly **complex**
  - Millions of lines of code, written by different teams that need to cooperate together.
  - Code is **not easy** to be understood by programmers who did not wrote it.
  - Code-based approaches are **complex**, **slow**, and most importantly **error-prone**.

Why Model-Driven Engineering?

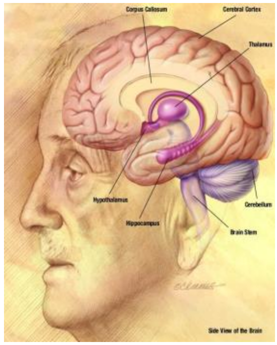
- Software systems are getting increasingly **complex**
  - Millions of lines of code, written by different teams that need to cooperate together.
  - Code is **not easy** to be understood by programmers who did not wrote it.
  - Code-based approaches are **complex**, **slow**, and most importantly **error-prone**.

- **MDE** is a way to deal with such problem
  - **Simplifying** the problem to be solved by **abstraction, i.e.,** the system can be considered at higher levels of **abstraction**:
    - Better **understandability** of the system, since the system can be described in terms of its main components, functionalities, aspects, etc.
    - Better **cooperation** and **coordination** among developing teams as they can work on "clearly" defined components, functionalities, aspects.
    - Allows to focus on **specific aspects** of the system (e.g., dependability, security).
    - Allows for revealing/detecting errors in the system design by early analysis .

Why Model-Driven Engineering?

1. Low cost
2. Easily understood
3. Easy to test in a controlled environment
4. Easy to modify the model if not satisfied

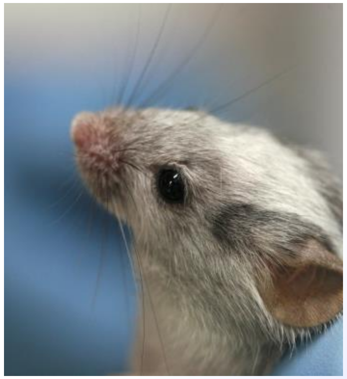
Examples of model (in general) : Model organism



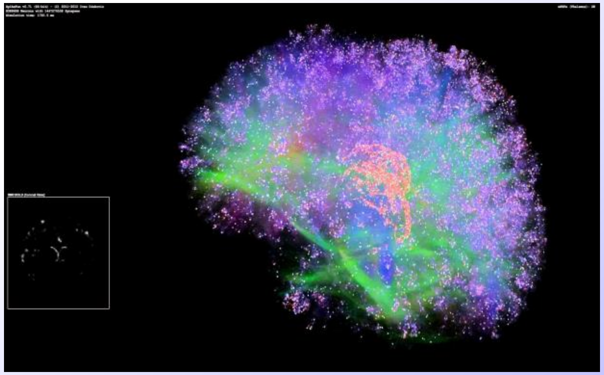
Human (system)

↓

Guinea pig (model)



Lab mouse (model)

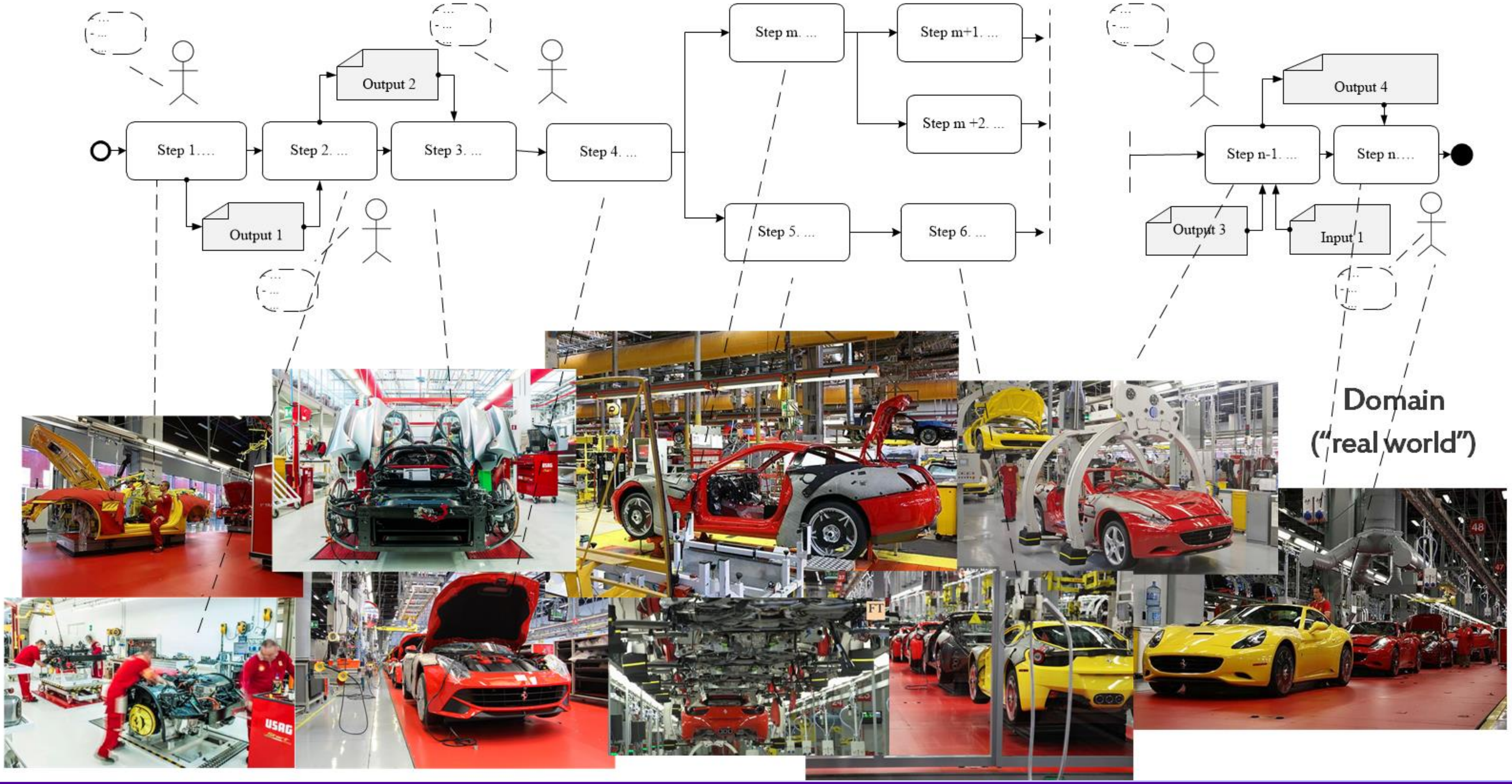


Simulation software (model)

Example 1

Domain
("real world")

Example 1


Step 1....
Output 1
Step 2. ...
Output 2
Step 3. ...
Step 4. ...
Step m. ...
Step m+1. ...
Step m +2. ...
Step 5. ...
Step 6. ...
Step n-1. ...
Output 4
Step n....
Output 3
Input 1
Domain
("real world")

Model Verification and Validation (V&V)

- **Verifying the model**
  - If tests on model passes, its not necessary it will pass on system !
  - However, if tests on model fails, its quite unlikely that it will pass on system.
- **Validating the model**
  - There should be some way to validate a model ! (e.g., simulation, relying on experts, using some logical constraints).
- **In general**

cost and complexity ∝ accuracy

Disadvantages of MDE

- **Modeling** can be a very time consuming activity.
- **Models** are for abstraction and not necessarily **correct** for **implementation**.
- You can Verify and Validate (V&V) the model of the system, not the system itself, i.e., there is a chance that the implimeneted system contains “**errors**” even if the model is “**correct**” with respect to the **designer’s assumptions**.

All models are wrong, but some are useful

- George Box

Modeling and models

Modeling:

- **Modeling** is a cost-effective use of **something** in place of **something** else for some purpose [1].
- **Modeling** allows for the use of something that is **simpler**, **safer**, and/or **cheaper** than **reality** instead of **reality** for some purpose [1].

Modeling and models

➢ Modeling:

- **Modeling** is a cost-effective use of **something** in place of **something** else for some purpose [1].
- **Modeling** allows for the use of something that is **simpler**, **safer**, and/or **cheaper** than **reality** instead of **reality** for some purpose [1].

➢ Models:

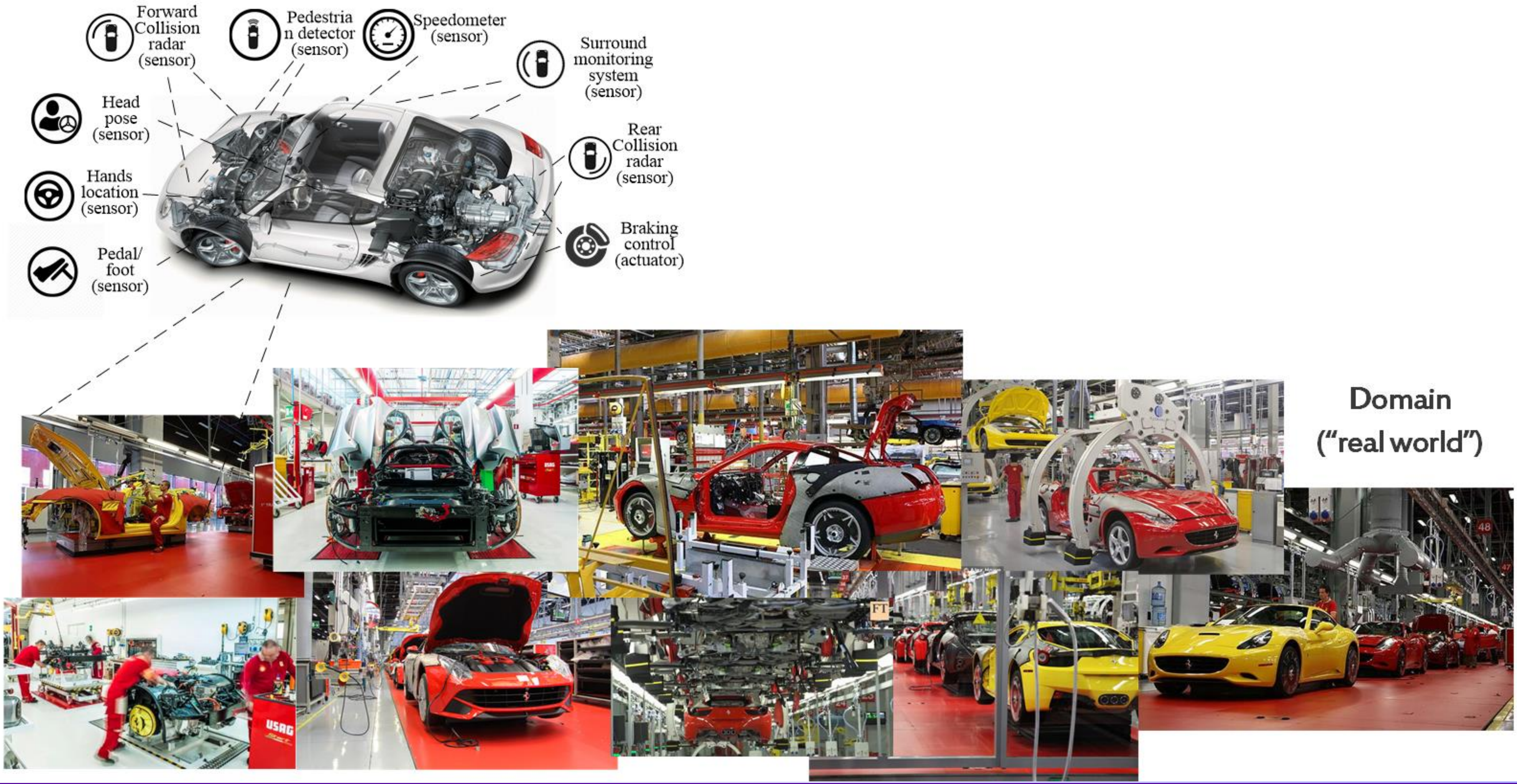
- A **model** represents **reality** for the given purpose, i.e., a **model** is an **abstraction** of **reality**. Usually, **models** cannot represent all aspects of **reality** [1].
- This allows for dealing with the real world in a simplified manner, avoiding the **complexity** and **danger** of reality [1].

Example 2

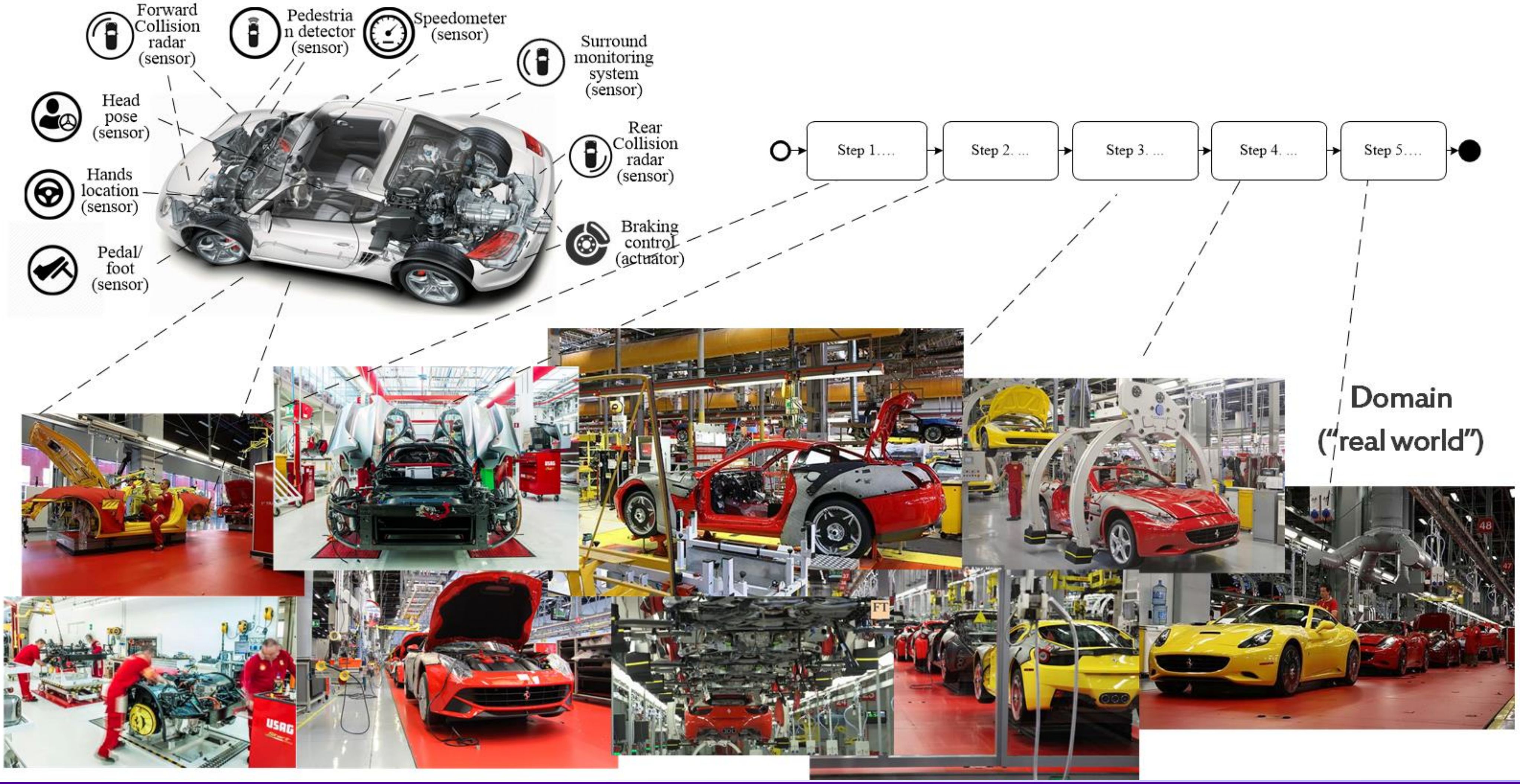
Domain
("real world")

Example 2

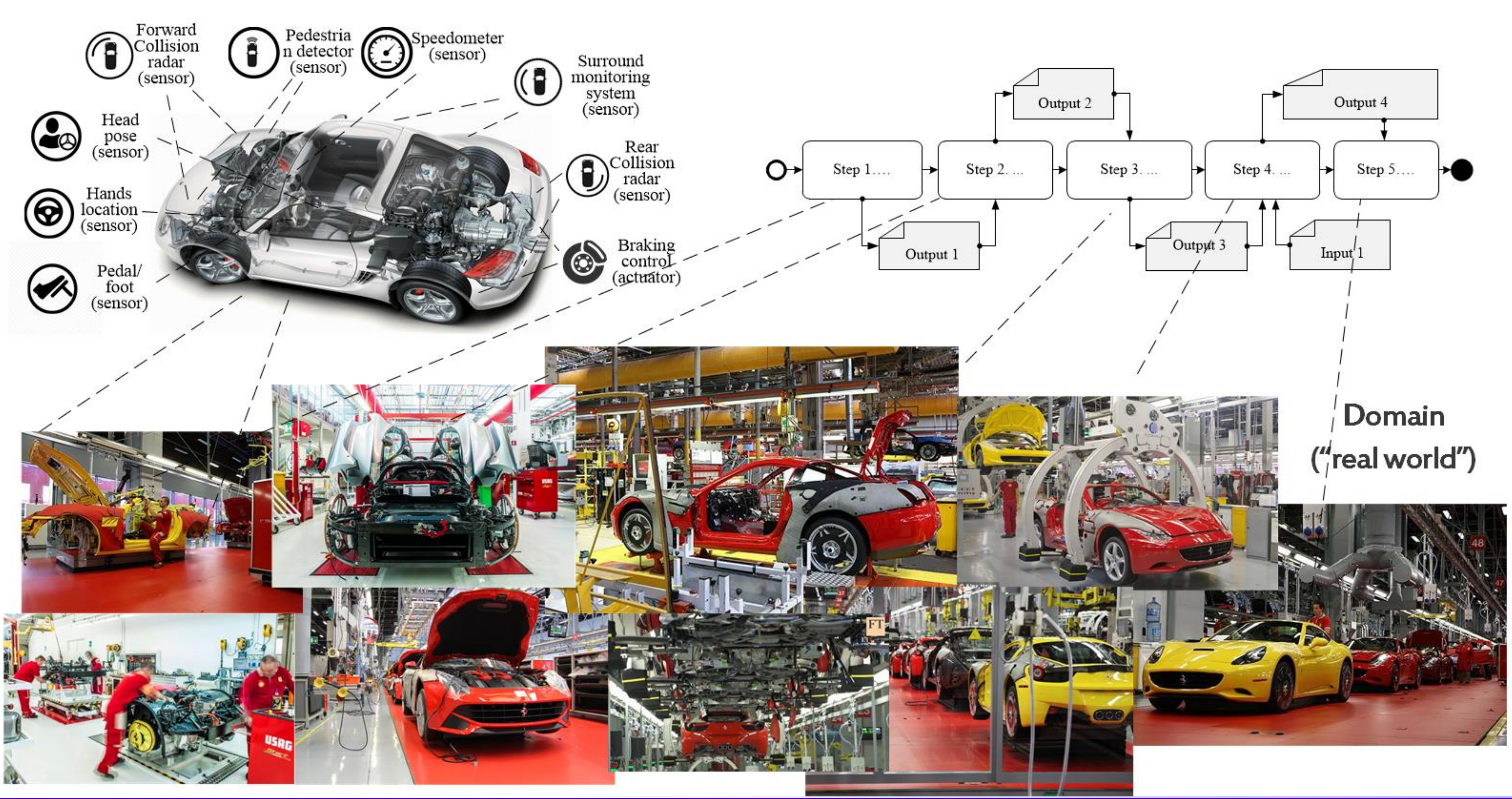
Example 2


Forward Collision radar (sensor)
Pedestrian detector (sensor)
Speedometer (sensor)
Surround monitoring system (sensor)
Head pose (sensor)
Rear Collision radar (sensor)
Hands location (sensor)
Braking control (actuator)
Pedal/ foot (sensor)
Domain ("real world")

Example 2

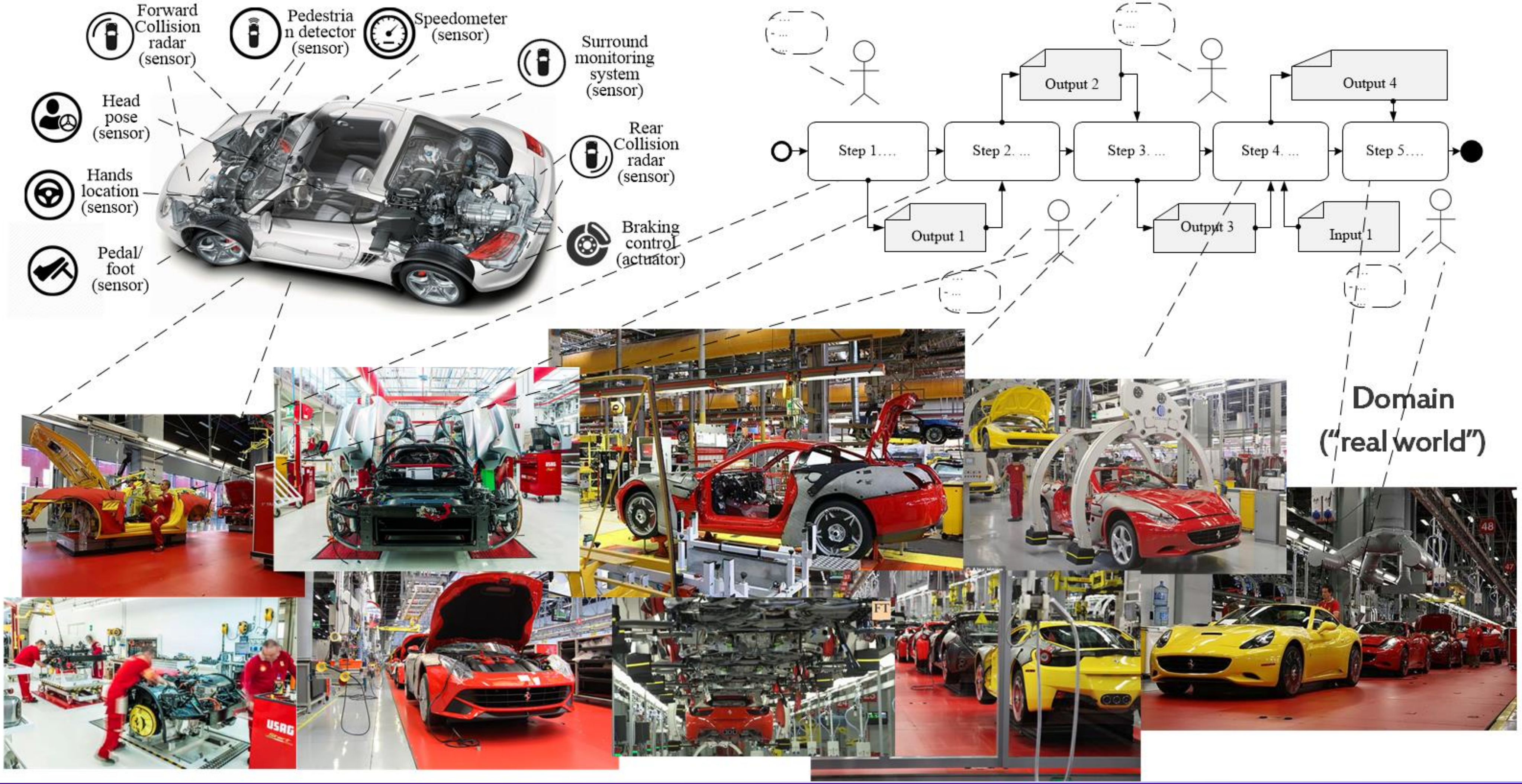

Forward Collision radar (sensor)
Pedestria n detector (sensor)
Speedometer (sensor)
Surround monitoring system (sensor)
Head pose (sensor)
Hands location (sensor)
Pedal/ foot (sensor)
Rear Collision radar (sensor)
Braking control (actuator)
Step 1.…
Step 2. ...
Step 3. ...
Step 4. ...
Step 5.…
Domain (“real world”)

Example 2



Domain ("real world")

Example 2


Forward Collision radar (sensor)
Pedestria
n detector (sensor)
Speedometer (sensor)
Surround monitoring system (sensor)
Head pose (sensor)
Hands location (sensor)
Rear Collision radar (sensor)
Pedal/ foot (sensor)
Braking control (actuator)
Step 1….
Step 2. ...
Step 3. ...
Step 4. ...
Step 5….
Output 1
Output 2
Output 3
Output 4
Input 1
- ...
- ...
- ...
Domain
("real world")

Modeling and models

- **Modeling** is supposed to **represent** a particular reference ("domain") for a particular purpose in a cost-effective (feasible) way [1].

- **Open questions:**
  - How to represent the reference ("domain") ?
  - How to define the purpose of the modeling?
  - How to guarantee it is cost-effective ?

Modeling and models

- **Modeling** is supposed to **represent** a particular reference (“domain”) for a particular purpose in a cost-effective (feasible) way [1].

- **Open questions:**
  - How to represent the reference (“domain”) ?
    - WHAT are we modeling?
  - How to define the purpose of the modeling?
    - WHY we are modeling?
  - How to guarantee it is cost-effective ?
    - HOW the model can be of high-quality and still feasible to be used?

Selected References for Reading

[1] Rothenberg J, Widman LE, Loparo KA, Nielsen NR. *The nature of modeling*. in Artificial Intelligence, Simulation and Modeling. 1989. (Simple and very interesting paper).

[2] D. C. Schmidt, *Guest Editor's Introduction: Model-Driven Engineering*, in Computer, vol. 39, no. 2, pp. 25-31, Feb. 2006. doi: 10.1109/MC.2006.58

[3] Van Der Straeten, Ragnhild, Tom Mens, and Stefan Van Baelen. "Challenges in model-driven software engineering." *International Conference on Model Driven Engineering Languages and Systems*. Springer, Berlin, Heidelberg, 2008.