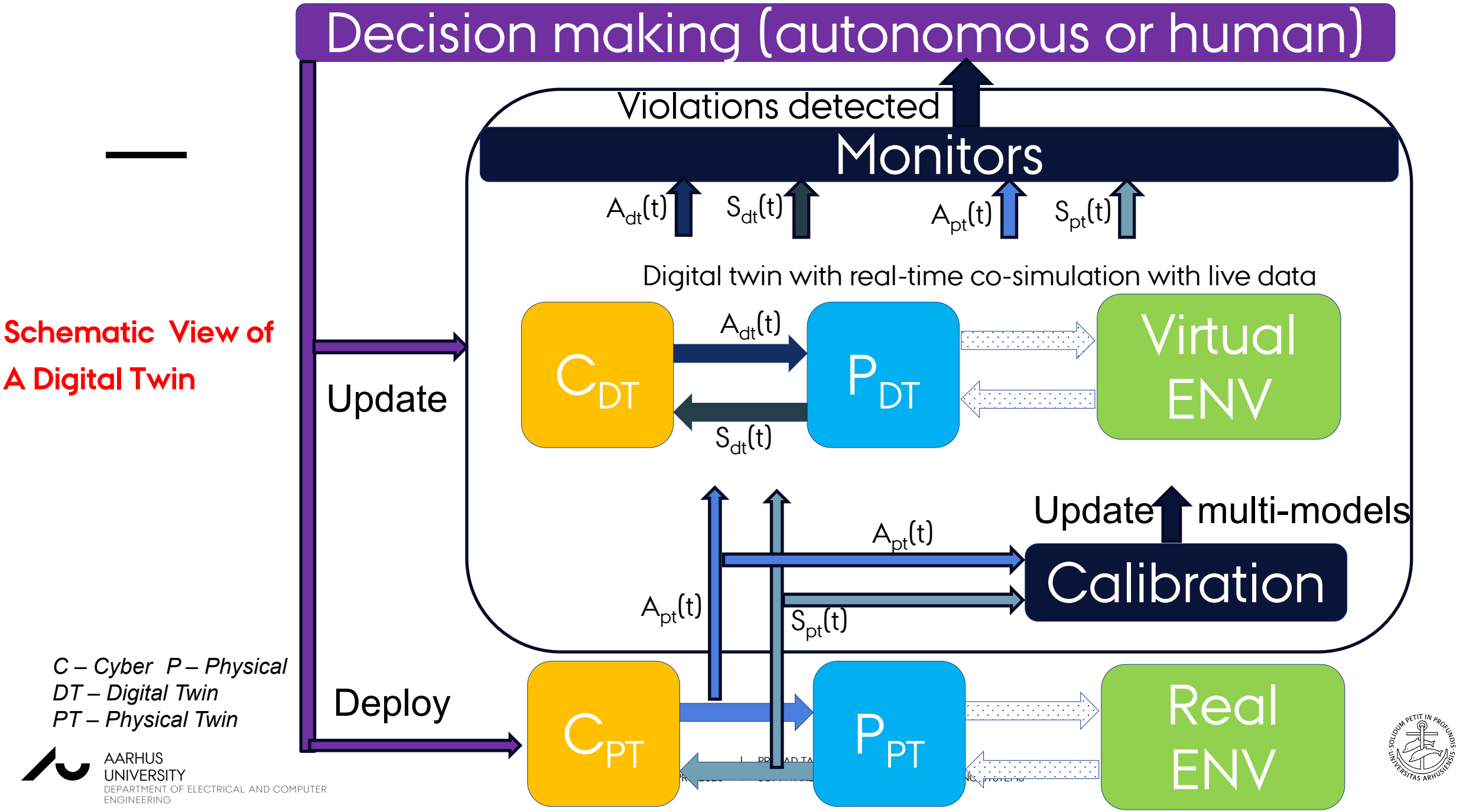
Digital twin as a Service (DTaaS) Software Platform

Prasad Talasila

prasad.talasila@ece.au.dk

PRESENTATION OUTLINE

1) *Requirements for Digital Twin Platforms*
    1) *User Requirements*
    2) Technical Requirements
    3) Lessons from Other Projects
2) A Conceptual Framework
3) DTaaS software platform
4) Implementation Status



Schematic View of A Digital Twin

C – Cyber P – Physical
DT – Digital Twin
PT – Physical Twin

AARHUS UNIVERSITY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

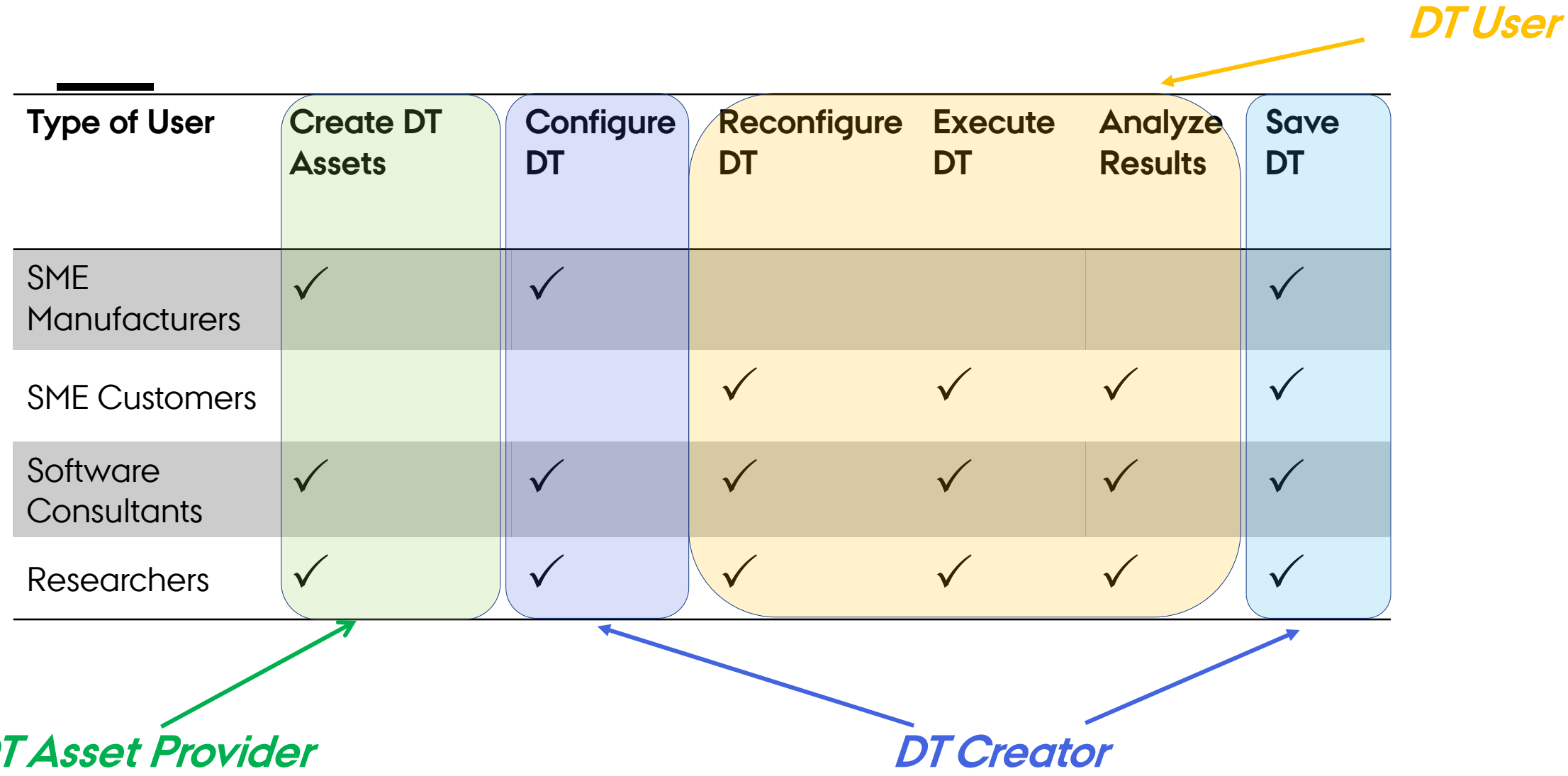
WHO ARE THE USERS?

Type of User	Create DT Assets	Configure DT	Reconfigure DT	Execute DT	Analyze Results	Save DT*
SME Manufacturers	✓	✓				✓
SME Customers			✓	✓	✓	✓
Software Consultants	✓	✓	✓	✓	✓	✓
Researchers	✓	✓	✓	✓	✓	✓

Personas / Roles

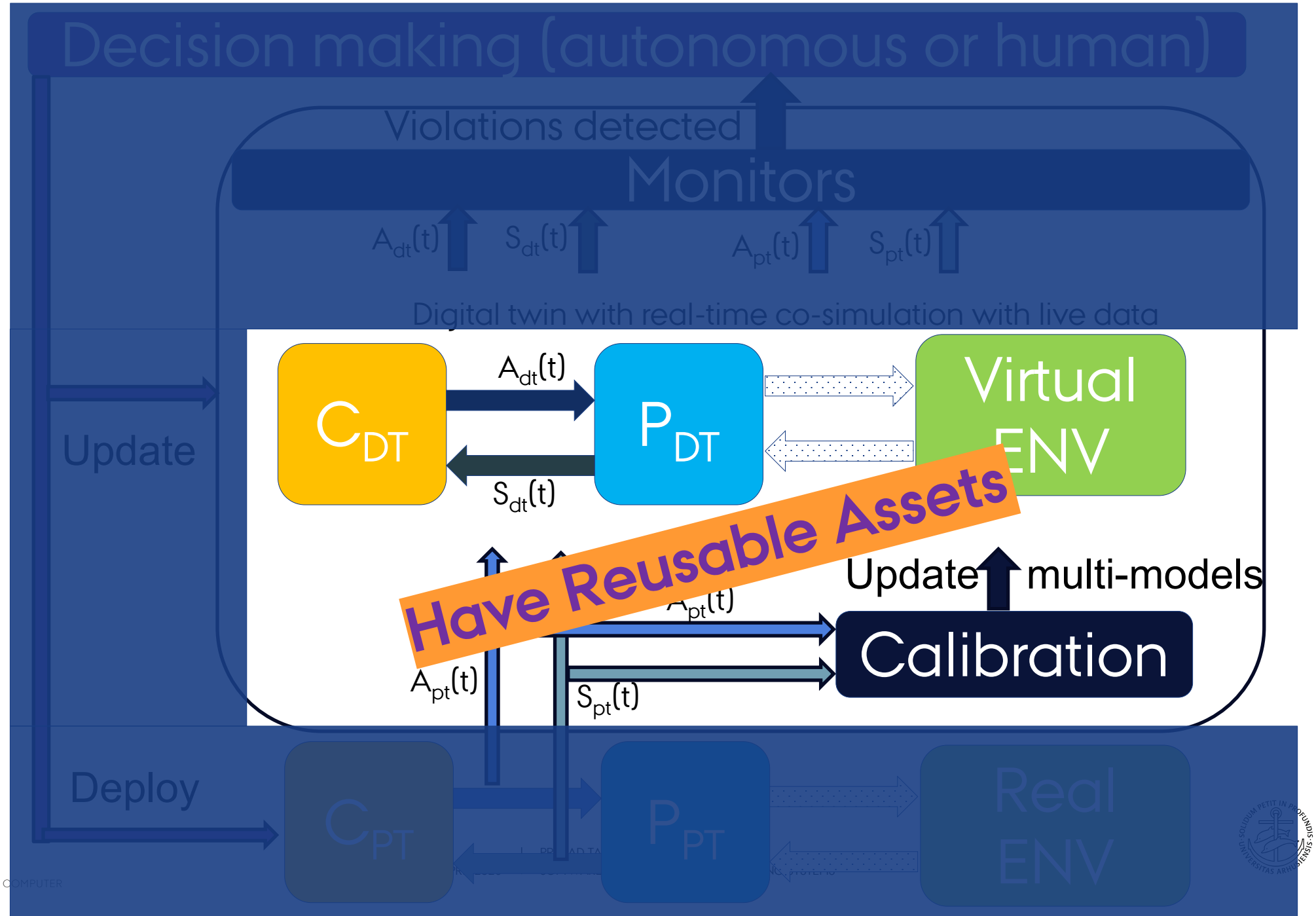
*Save DT – Save a running instance of a DT

A PEEK INTO USER ROLES?



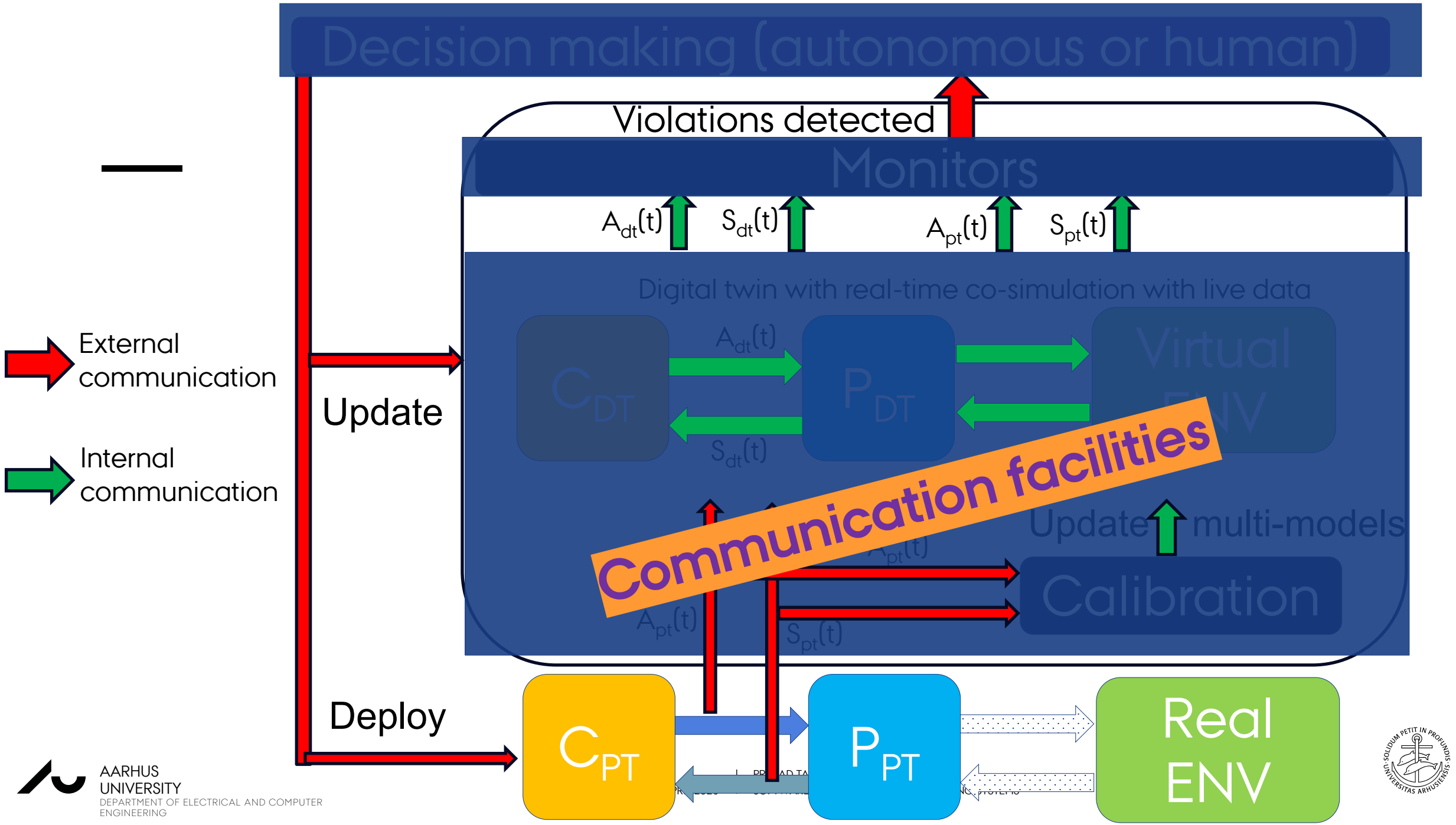
PRESENTATION OUTLINE

1) *Requirements for Digital Twin Platforms*
    1) User Requirements
    2) *Technical Requirements*
    3) Lessons from Other Projects
2) A Conceptual Framework
3) DTaaS software platform
4) Implementation Status

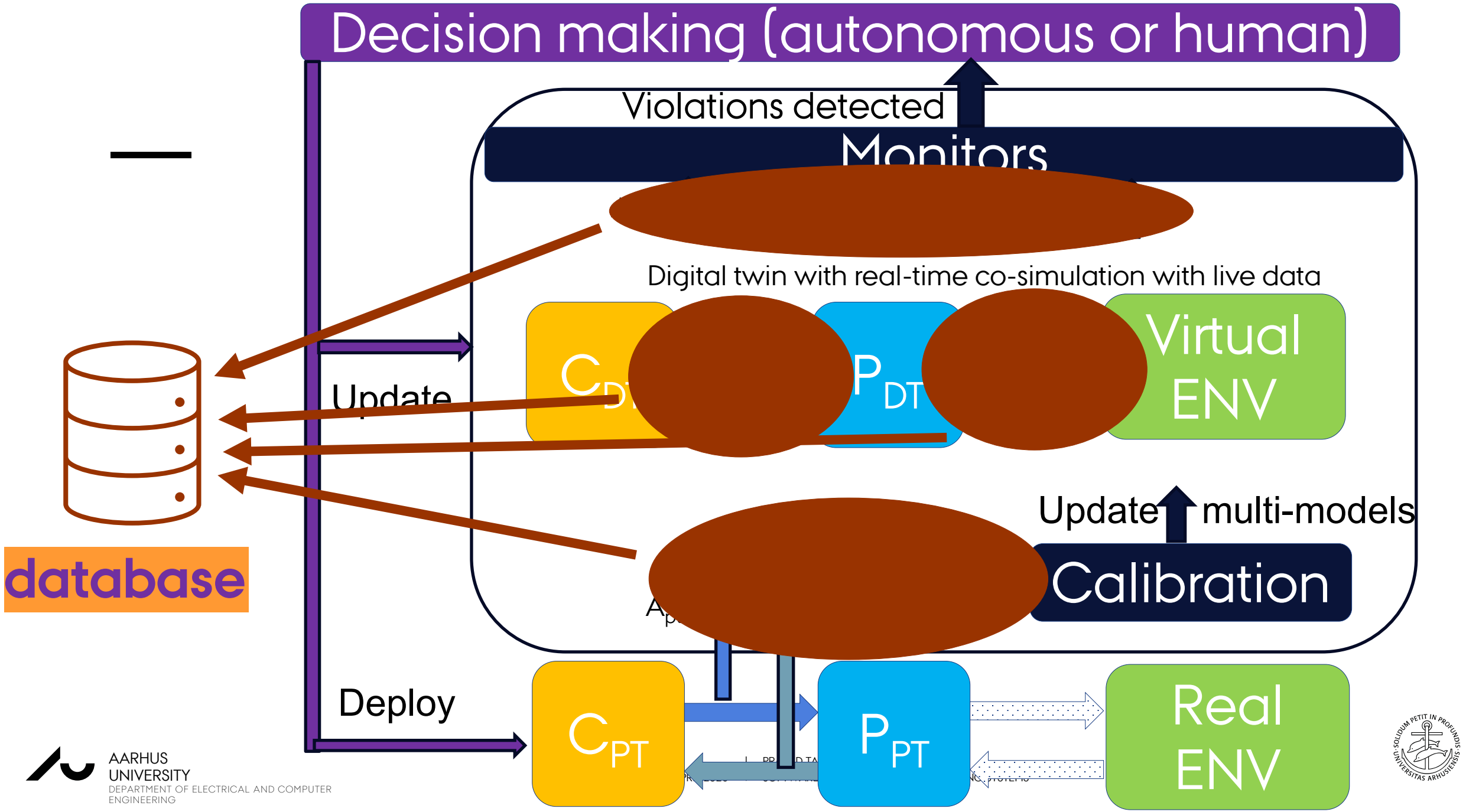

Decision making (autonomous or human)
Violations detected
Monitors
$A_{dt}(t)$
$S_{dt}(t)$
$A_{pt}(t)$
$S_{pt}(t)$
Digital twin with real-time co-simulation with live data
Update
C_{DT}
$A_{dt}(t)$
$S_{dt}(t)$
P_{DT}
Virtual ENV
Have Reusable Assets
Update multi-models
$A_{pt}(t)$
$A_{pt}(t)$
$S_{pt}(t)$
Calibration
Deploy
C_{PT}
P_{PT}
Real ENV

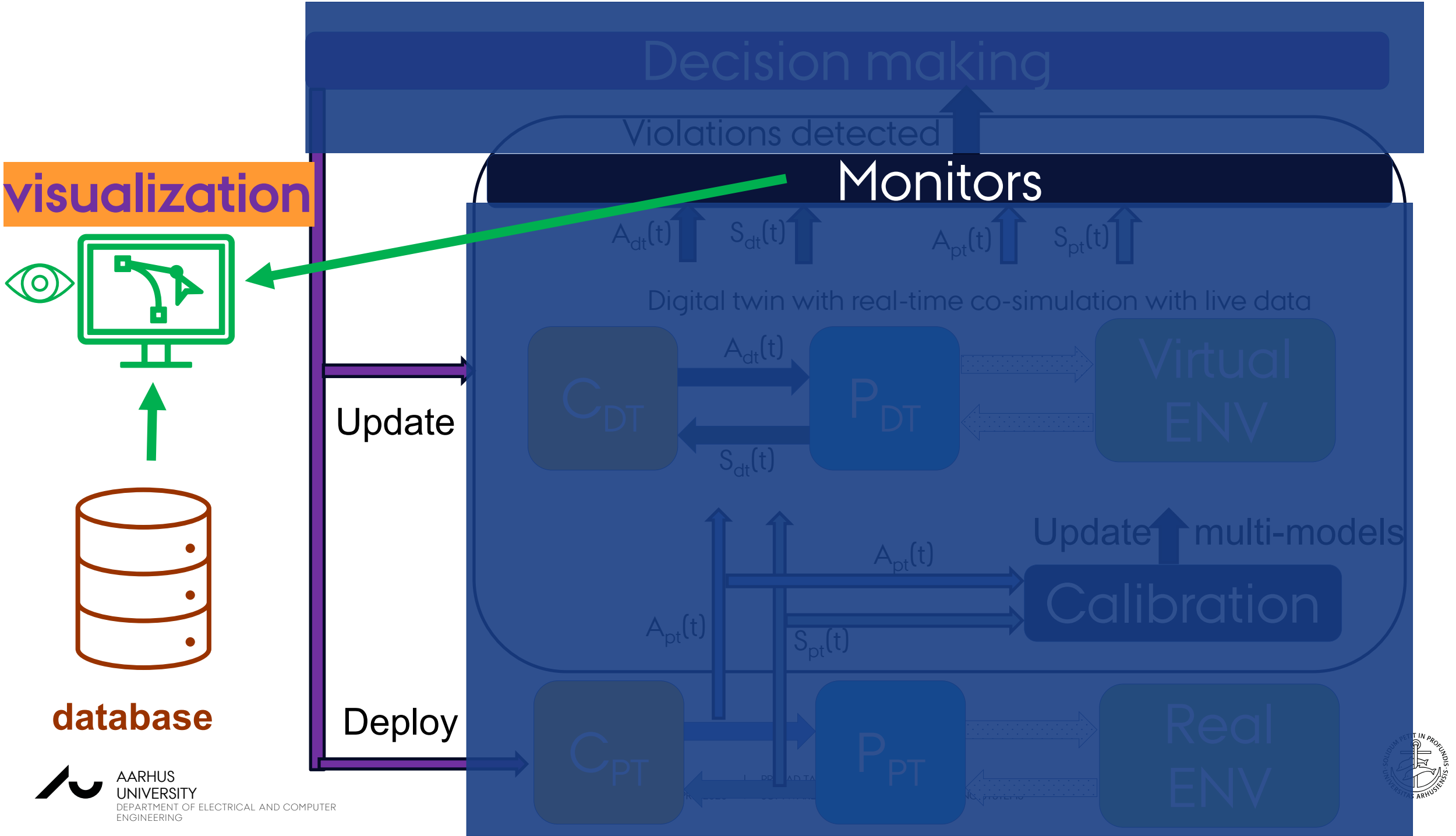
What are all the reusable assets?

Published on the Platform and available for reuse

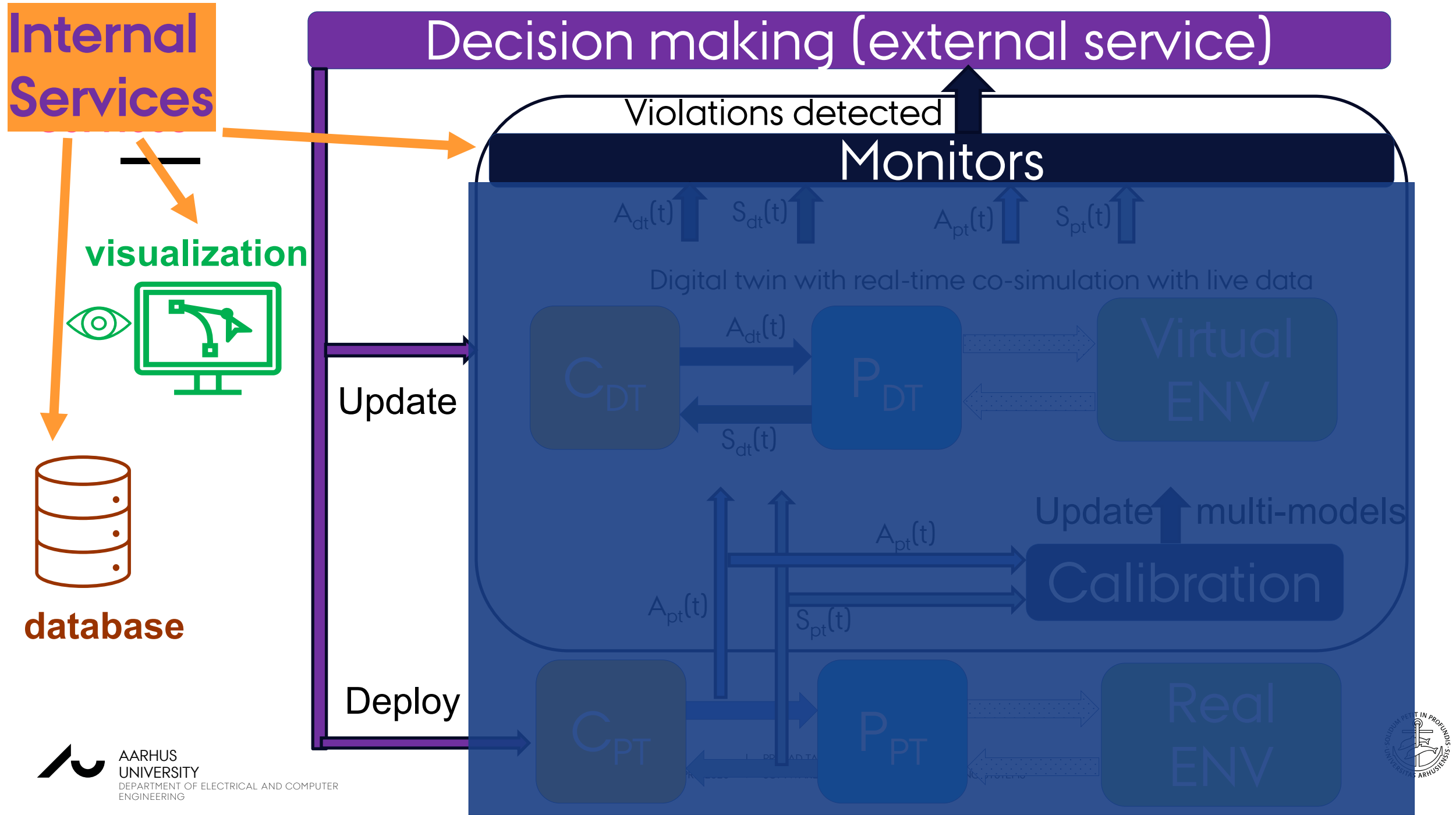


AARHUS UNIVERSITY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING


Decision making (autonomous or human)
Violations detected
Monitors
Digital twin with real-time co-simulation with live data
Update
C_{DT}
P_{DT}
Virtual ENV
database
Update multi-models
Calibration
Deploy
C_{PT}
P_{PT}
Real ENV



AARHUS UNIVERSITY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING



AARHUS UNIVERSITY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

IN SHORT, WHAT DOES A SCALABLE DT AN AUTOMATION PLATFORM REQUIRE?

Reusable Assets

Communication Facilities

Databases

Visualization

Execution Manager

Lifecycle Manager

***Usual Suspects*:**

Unified Web Application

Gateway (for single point of entry)

Security

Accounting

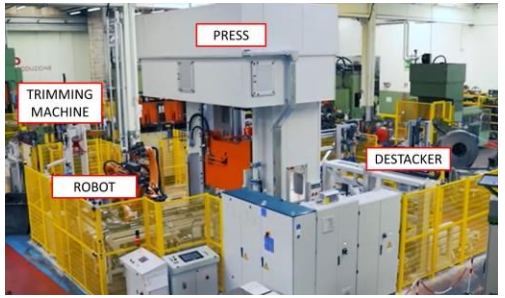
PRESENTATION OUTLINE

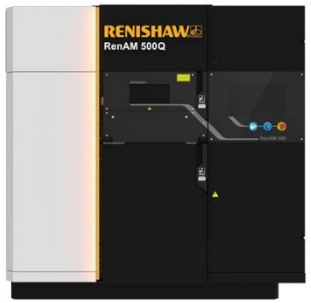
1) *Requirements for Digital Twin Platforms*
    1) User Requirements
    2) Technical Requirements
    3) *Lessons from Other Projects*
2) A Conceptual Framework
3) DTaaS software platform
4) Implementation Status

DIGITbrain Project

What are the disjoint questions People ask about Digital Twins?

- what is on the shopfloor?
- Do the machines have sensors?
- Which data in which format do they provide?
- Do behaviour models (or any other types of models) exist?
- Which tools exist to evaluate such models?
- Where to execute these tools?
- How to transport the data to that location?
- How to monitor the execution?
- How to get feedback to the user?
- .....



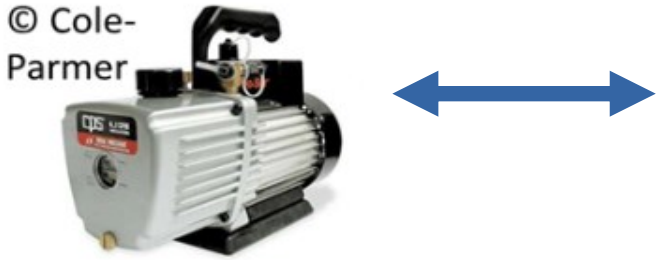


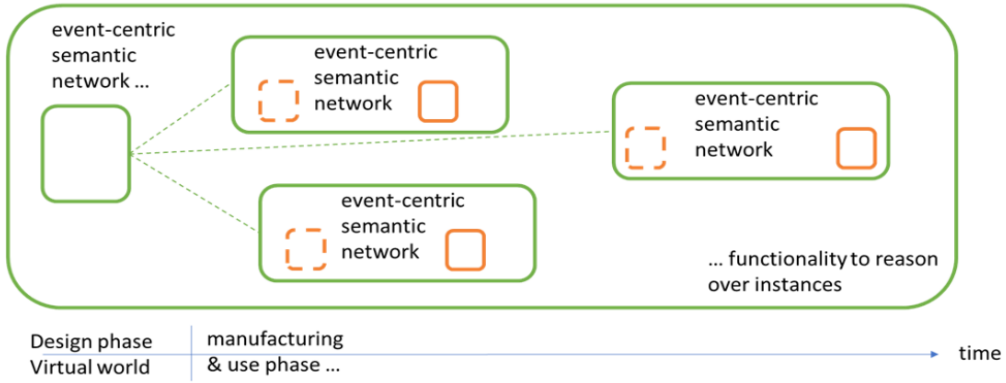
What can be one way of thinking about the digital twins for manufacturing?

Ref: Prof André Stork, Task 1.2 – Technical Project Management, WP 1 – Project Coordination and Technical Management, DIGITbrain project, 15-Feb-2023

The Notion of Digital Product Brain

Extend the traditional digital twin concept towards the **Digital Product Brain** that **steers the behaviour and performance of an industrial product** (mechatronic system or manufacturing machine) by coalescing its physical and digital dimensions and by **memorising the occurred (physical and digital) events throughout its entire lifecycle**.





Key Aspects:

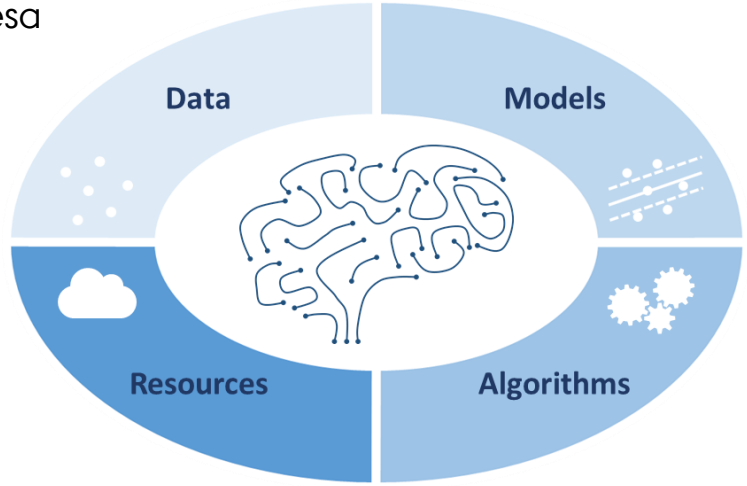
1. DPB evolves to track all significant events in the lifecycle of an IP
2. Evolution of DPB unique to each Industial Product sold (which is called Industrial Product Instance – IPI)
3. Semantic network representation for keeping track of events
4. DPB can be adapted to an IP family – product series of a company

Ref: 1) D3.3 - Second Update of the Testbed Infrastructure, DIGITbrain project
2) Andre Stork, Report on DIGITbrain (Db) Concepts and Terminology Definitions, DIGITbrain Review Meeting, July, 2021.

What goes into creation of DPB

Data: (users)

Farmers using Agrointelli Robotti
Customers of FabMetrics
Wood drying plants using Prodesa equipment



Models: (manufacturers)

Agrointelli (agricultural robot manufacturing factory and 3D CAD of agricultural robot)
Prodesa (manufacturer of wood pellet dryers)
FabMetrics (manufacturer of glass making equipment)

Execution Platform and Infrastructure:

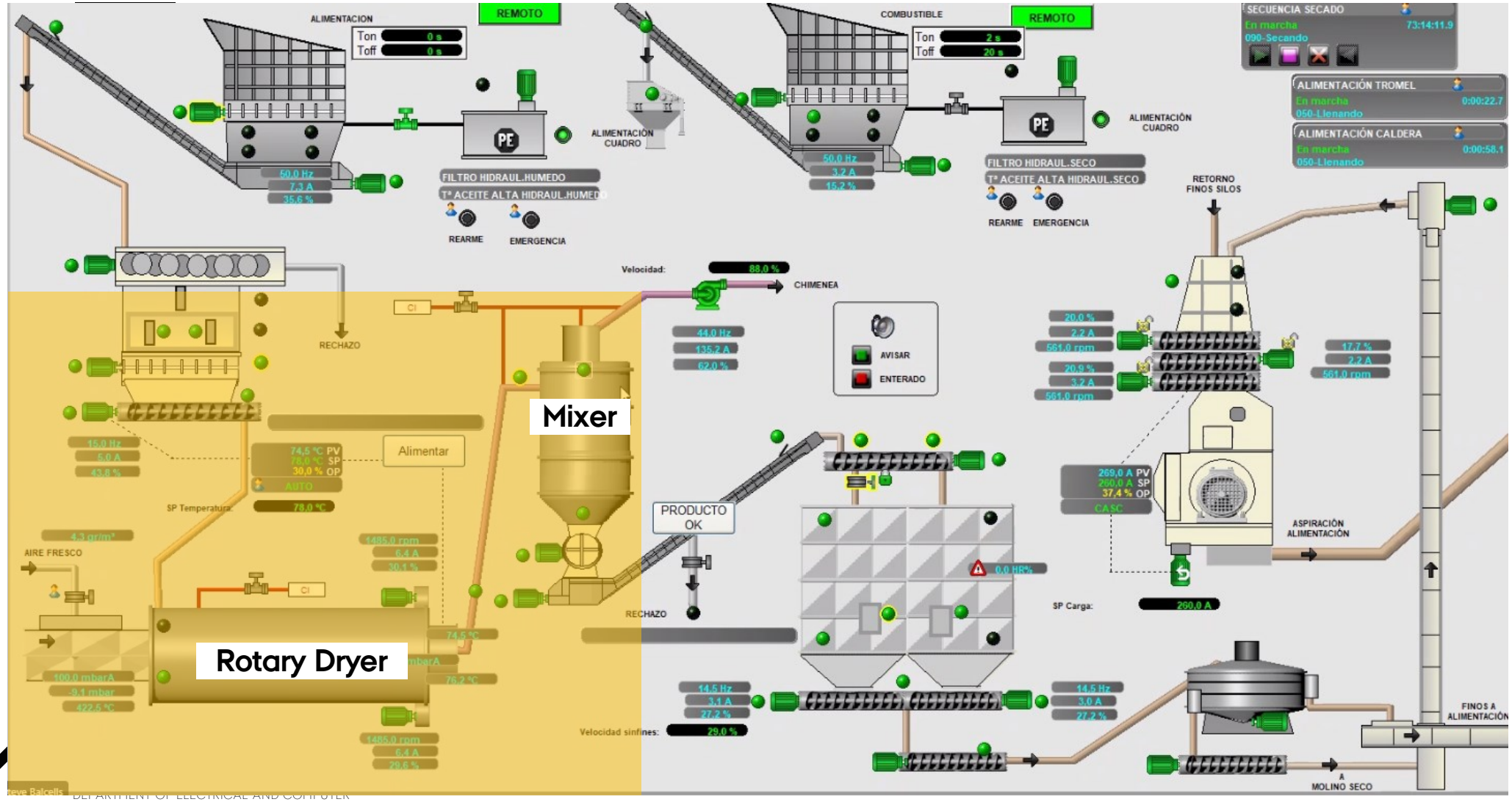
University of Westminister (MiCADO)
SZTAKI (CloudBroker)
EGI (Distributed Cluster Computing Infrastructure)
Public Cloud Services (Azure, AWS)
On-premises (FabMetrics)

Microservice and Algorithms: (software vendors)

Maestro - co-simulation orchestration engine by Aarhus University
CAELIA –ROM solver and Data Analysis by ITA-INNOVA
DDDSimulator – discrete event simulator and 3D visualizer by TTS
Ristra – Numerical simulator for mechanical structures by IGD, Fraunhofer
SAS a lifecycle assessment tool by SUPSI

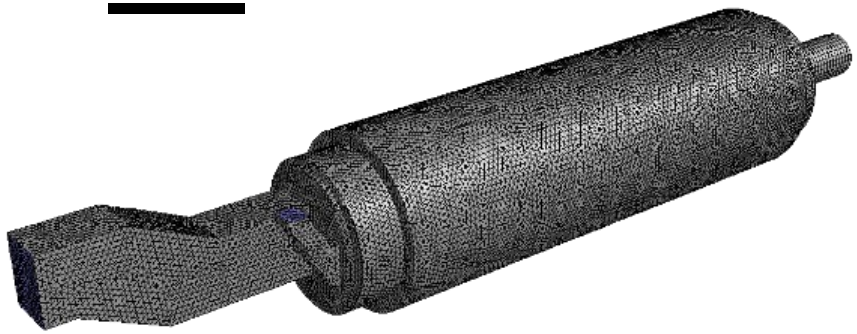
Example-1

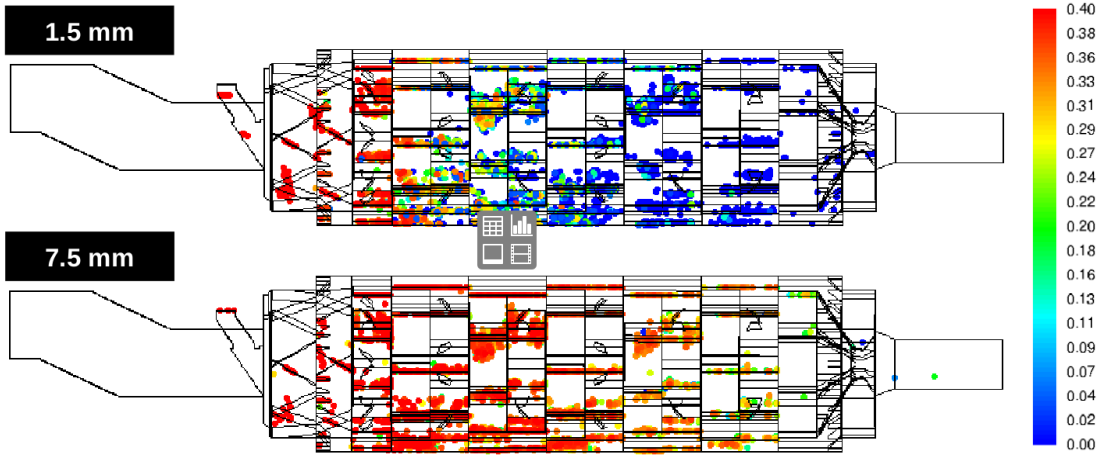
WOOD PELLET DRYING PLANT – PRODUCTION OPTIMIZATION



Example-1

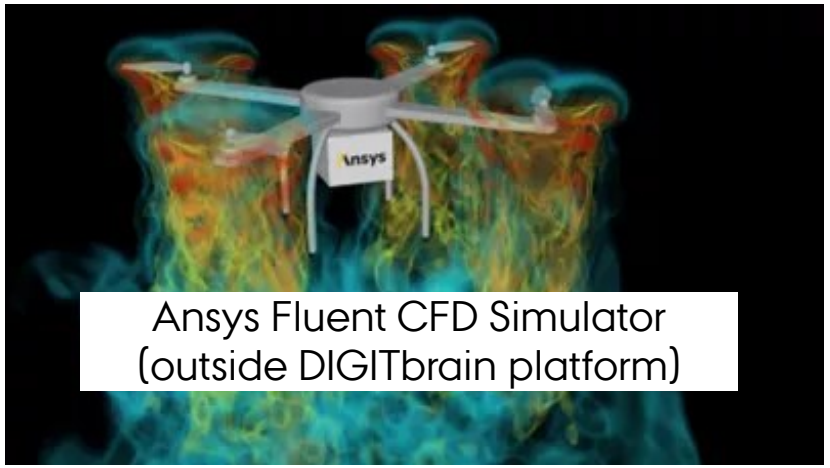
WOOD PELLET DRYING PLANT – COMPETING MODELS: CFD MODEL





Rotary Dryer CFD Model

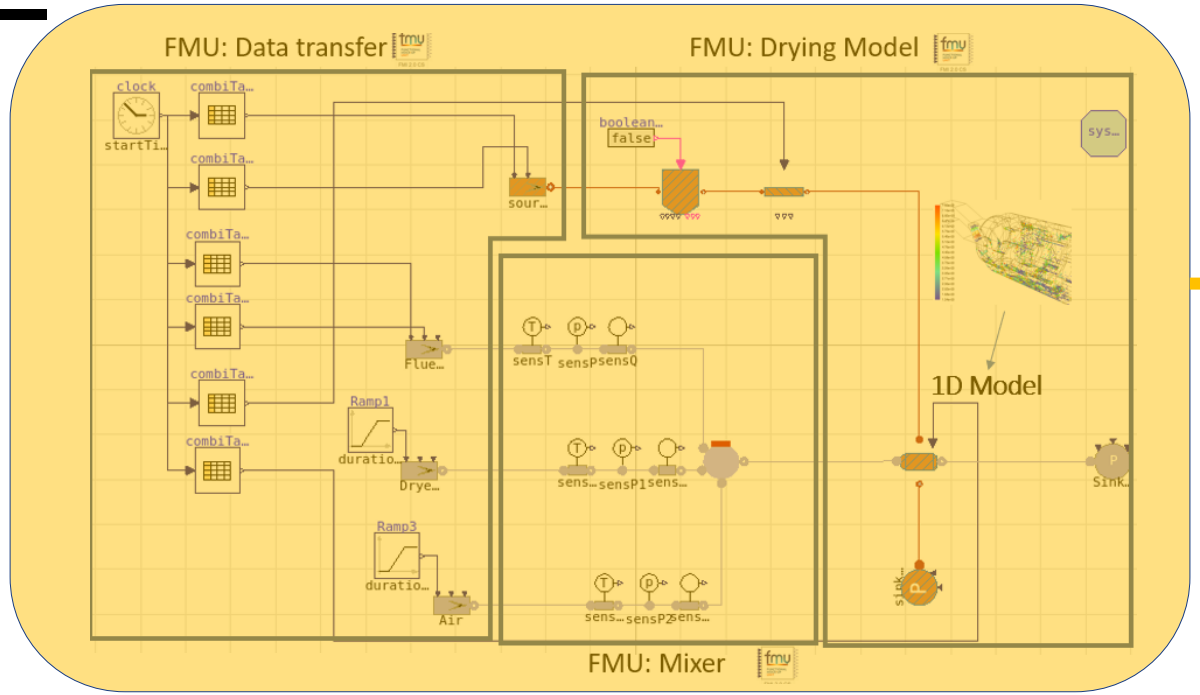
Plant data

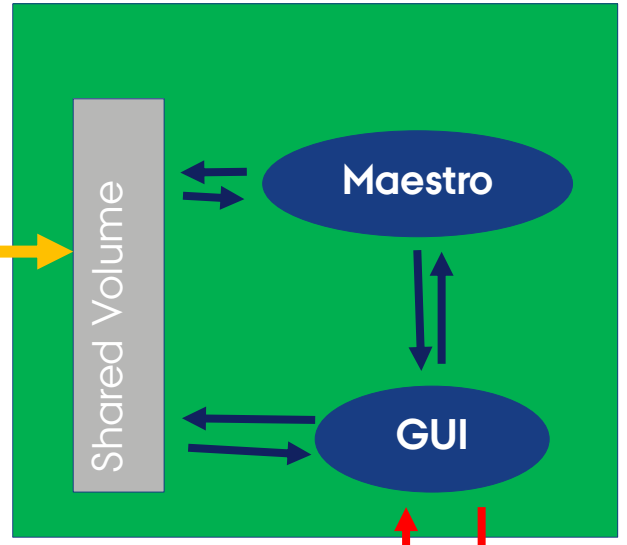


Simulation Results

Example-1

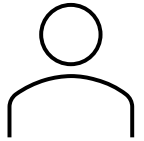
WOOD PELLET DRYING PLANT – COMPETING MODELS: CO-SIMULATION MODELS





Plant data

Optimized production plan



Production Planner

Model Data Microservice Algorithm

Example-1

SYNCHRONIZATION ALGORITHM BETWEEN MICROSERVICES

GUI Microservice	Maestro Microservice
1. Create output/SYNC.LCK 2. Update model 3. Remove model/SYNC.LCK 4. Busy wait on output/SYNC.LCK 5. Copy results from output/ directory 6. Go to step-1	1. Busy wait on non-existence of model/SYNC.LCK 2. Creates model/SYNC.LCK 3. Delete all files in output/ directory 4. Run co-simulation and saves results in output/ directory 5. Deletes output/SYNC.LCK 6. Goes to step-1

Such a synchronization is a requirement in an algorithm composed from microservices.
But this is unique to each algorithm

Example-1

WORKFLOW FOR THE USE OF DIGITAL TWIN

1. The user uploads input data to GUI web application which places the data in a shared volume.

2. The user also updates model parameters and the web application updates the model placed in shared volume.

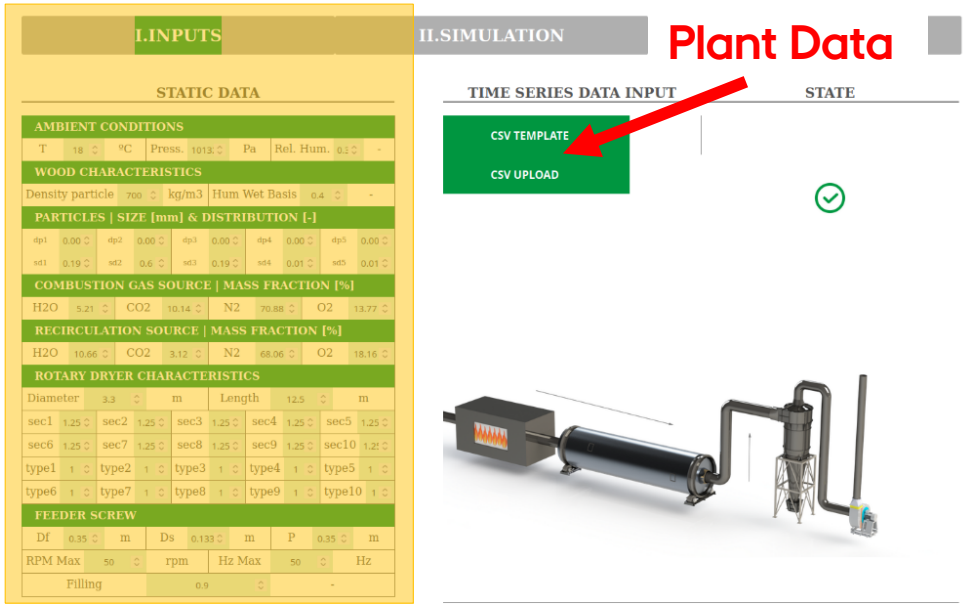
3. Maestro completes co-simulation and gives the results back to GUI

4. Co-operation between microservices happens uses synchronization algorithm

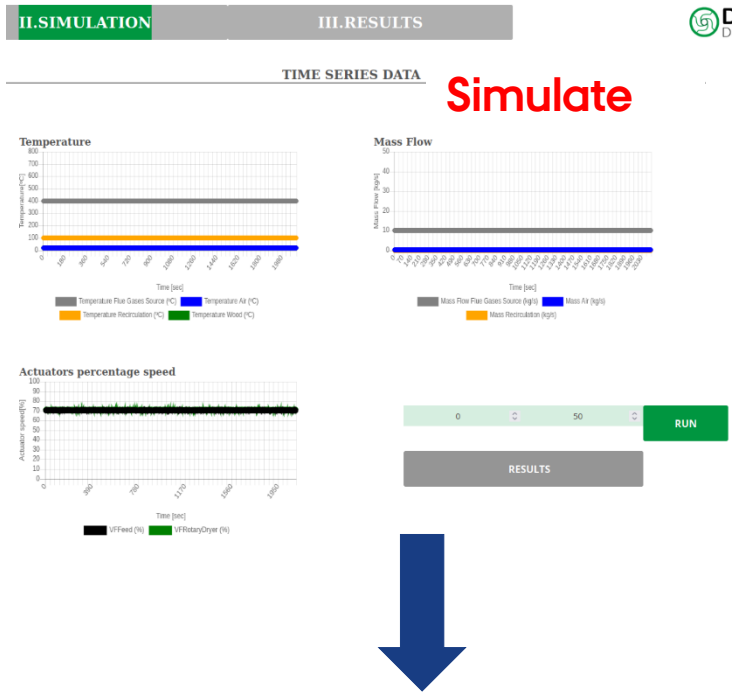
5. Uses sees visualization of results in GUI web application.

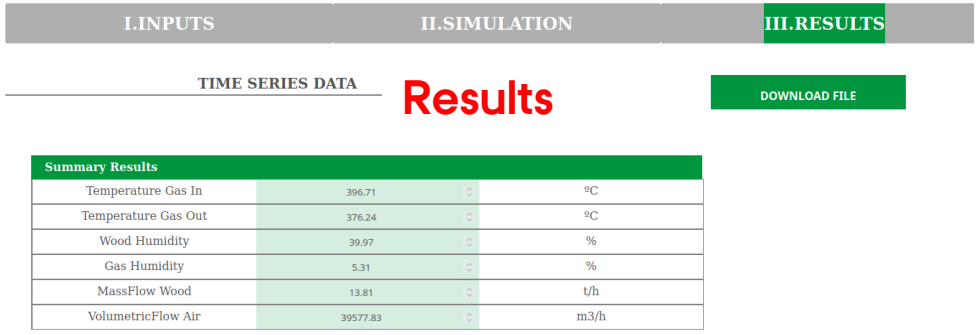
Example-1

What is the end result?
Result of Planned Production Based on Model Updates



Model Parameters





PRESENTATION OUTLINE

1) Requirements for Digital Twin Platforms
2) *A Conceptual Framework*
3) DTaaS software platform
4) Implementation Status
5) Onboarding Users

DIGITAL TWIN LAYERS: A PROPOSAL

NOTE: This is not a strictly layered architecture

WHAT IS INSIDE THE DIGITAL TWIN LAYERS?

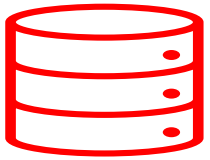
Twin Management

- Configuration of DT
- Analysis/queries capabilities
- Decision support
- Selection of what to visualise

Service Management

- Actual model simulation
- Experimental model simulation
- Monitors
- Calibration
- Reconfiguration
- Trend analysis
- 2D visualisation
- 3D visualisation

Asset Management


Data


Models


Functions


Tools

Data Ingestion & Processing

- Data collection
- Data transmission
- Data storage
- Data processing
- Data fusion
- Data visualization
- Adapters and brokers

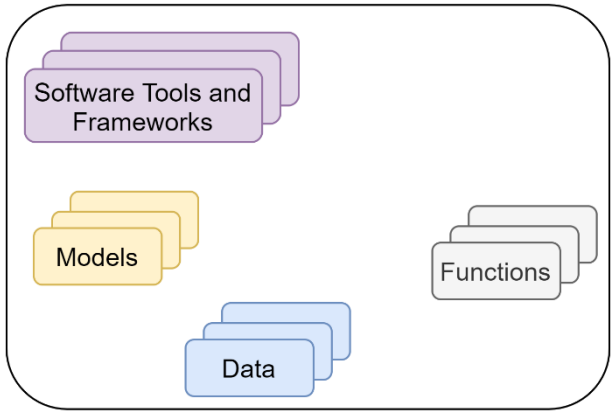
Physical Entities

- System of Systems
- System
- Subsystem
- Component
- Part

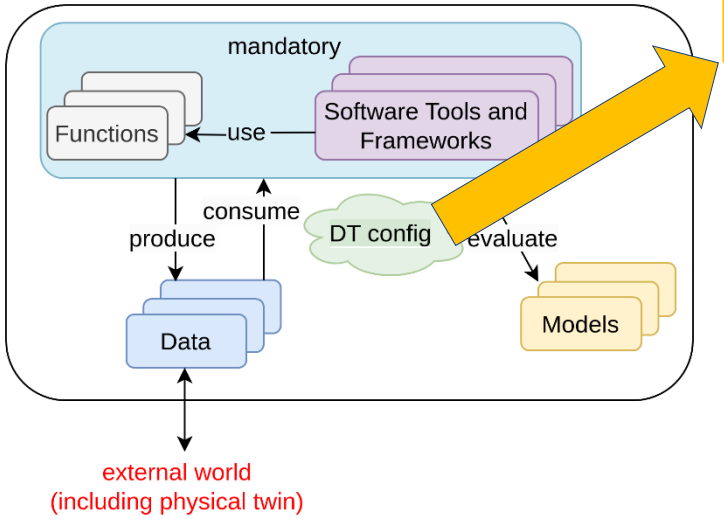
User Interaction

A STEP TOWARDS DEFINING DIGITAL TWINS...

Basic Components



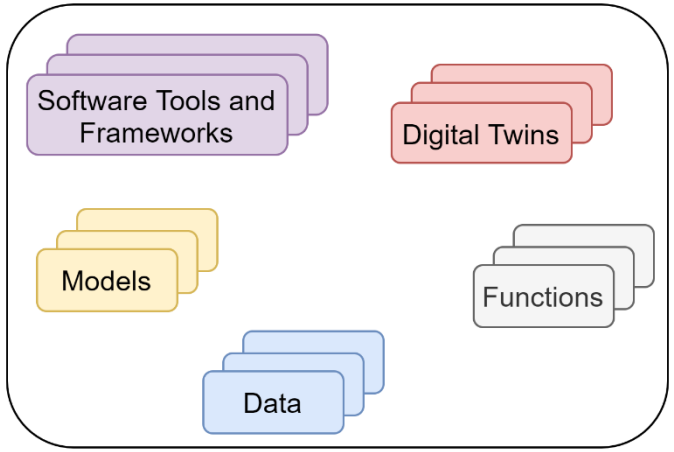
Create DT



DT Config C_{dt}^{l+1}

- Components C_c
- Infrastructure C_i
- Physical twin C_{pt}
- External world C_e
- Internal DTs C_{dt}^{l}
  (flat / hierarchical config)

Into the Component Library



HOW IS A DIGITAL TWIN DEFINED?

$$D_t = \{D^*, M^*, (FT)^+\}C_{dt}$$

Where,

D_t *= digital twin*

D = data *M = Model* *F = Function* *T = Tool*

C_{dt} *= DT configuration*

WHAT IS IN C_{dt}?

$$C_{dt} = \{C_a, C_i, C_e, C_{pt}\}$$

Where,

C_a = configuration of data, model, function and tool assets

C_i = infrastructure and the platform services configuration

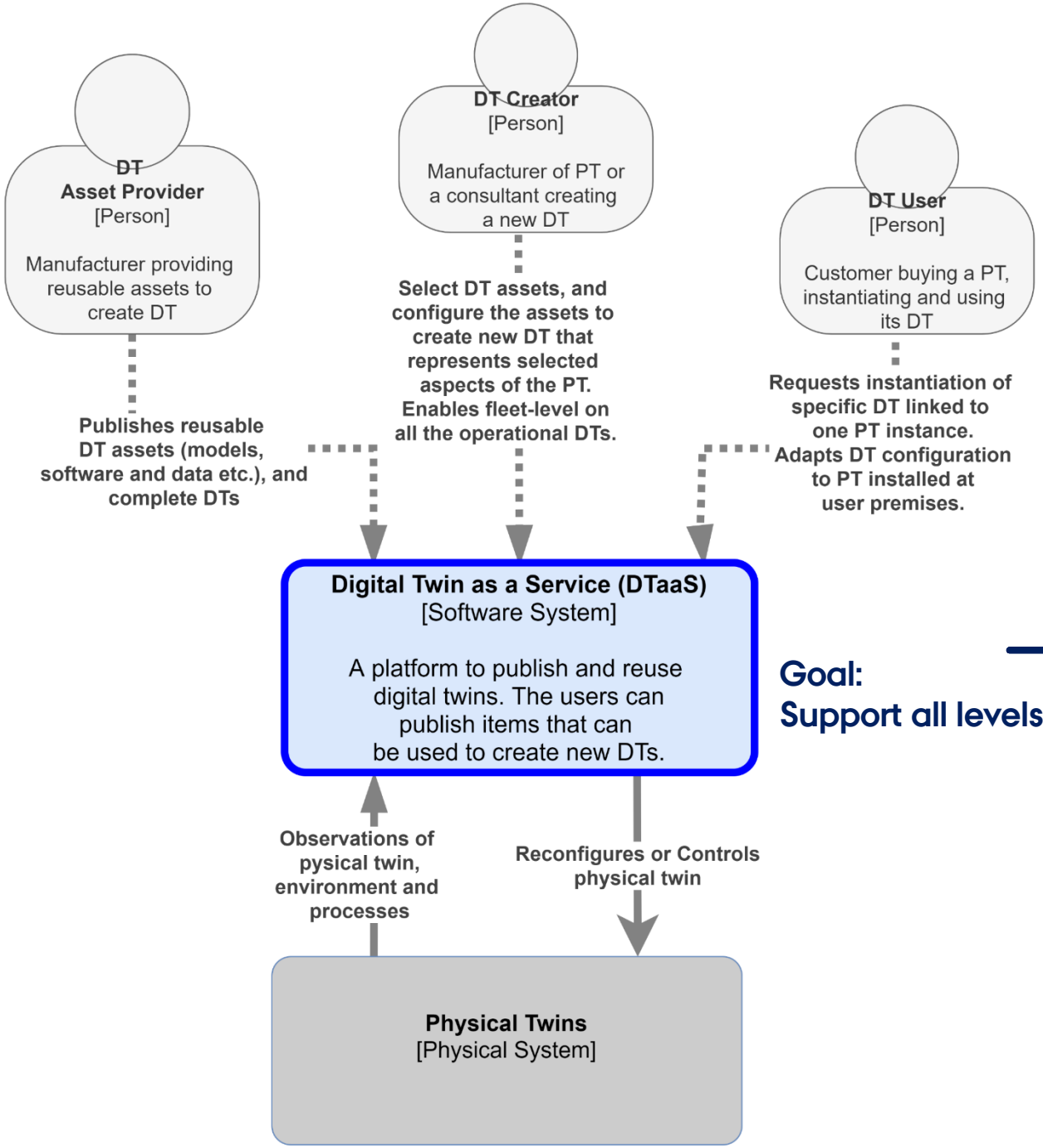
C_e = configuration for integrating with external systems

C_{pt} = physical twin configuration

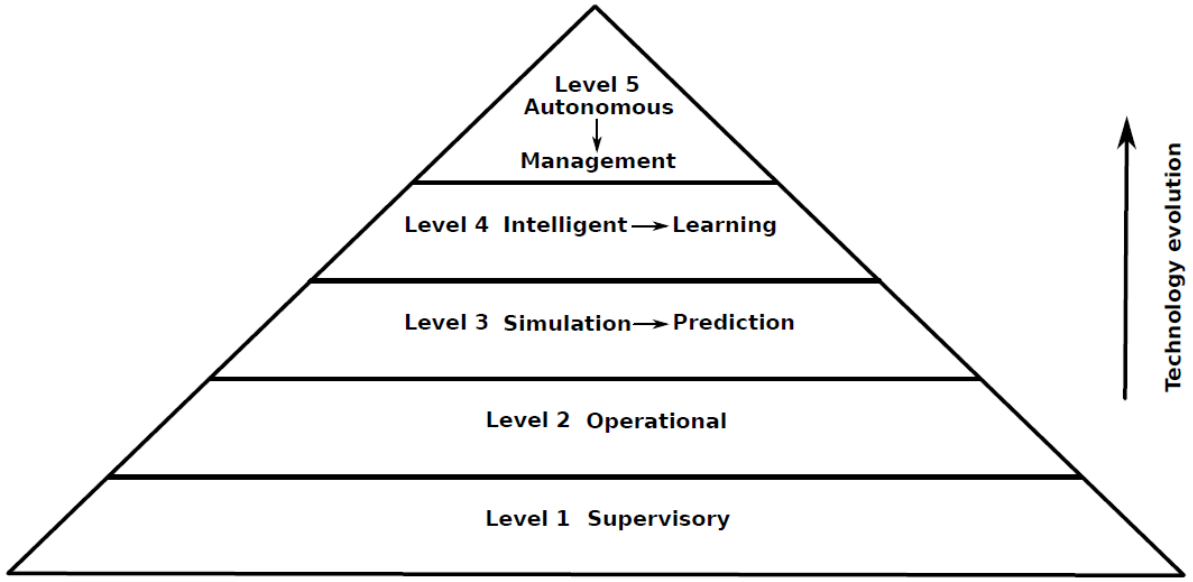
PRESENTATION OUTLINE

1) Requirements for Digital Twin Platforms
2) A Conceptual Framework
3) *DTaaS software platform*
    1) *System Architecture and Design*
    2) Digital Twin Lifecycle Manager
    3) Execution Manager
    4) Comparison with Other Platforms
4) Implementation Status

VERY HIGH LEVEL SYSTEM CONTEXT DIAGRAM



Goal: Support all levels



Hierarchy of DT Capabilities

PRASAD TALASILA
SOFTWARE ENGINEERING AND COMPUTING SYSTEMS

WITH RECAP OF TECHNICAL REQUIREMENTS

Reusable Assets

Communication Facilities

Databases

Visualization

Execution Manager

Lifecycle Manager

Usual Suspects:

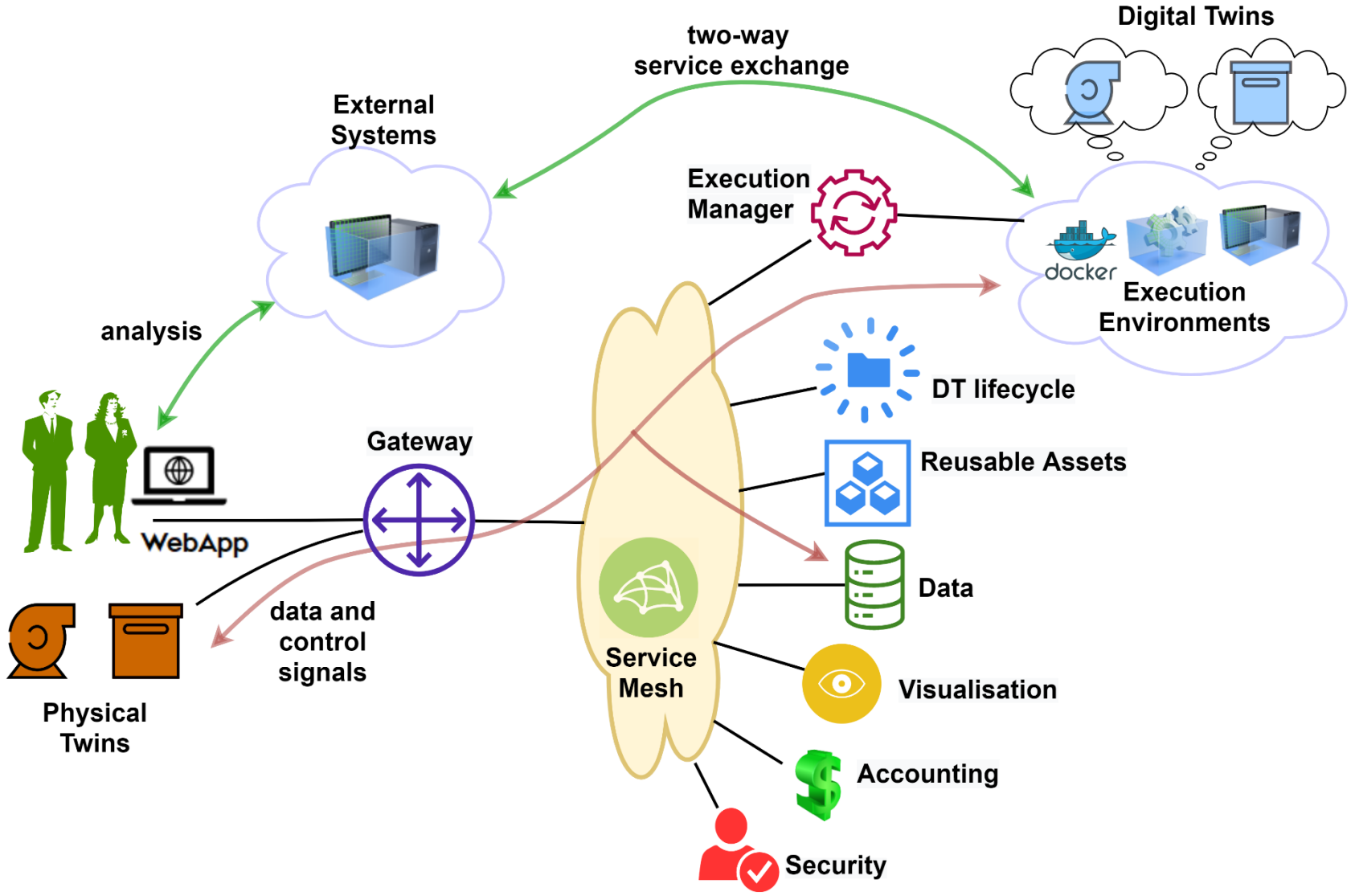
Unified Web Application

Gateway (for single point of entry)

Security

Accounting

SYSTEM ARCHITECTURE



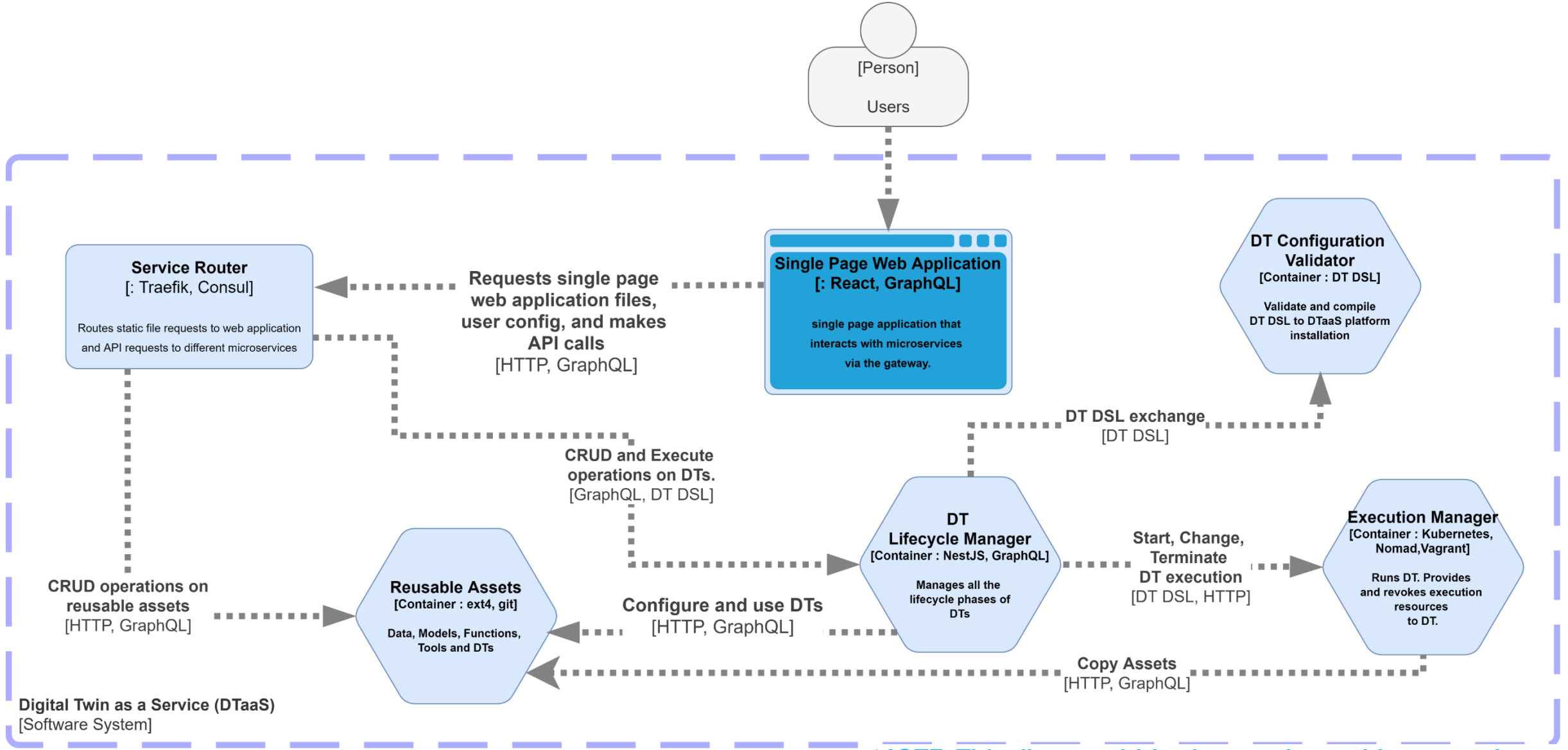
WHAT DO EACH OF THE SYSTEM COMPONENTS DO?

System Component	Responsibilities
security	Authentication and authorization
Accounting	Use of reusable assets and the platform resources
Visualization	Pre-defined or custom visualizations, dashboards etc.
Data	Data archives, Databases etc.
Reusable Assets	Models, Tools, Functions and Digital Twins available for reuse

WHAT DO EACH OF THE SYSTEM COMPONENTS DO? (2)

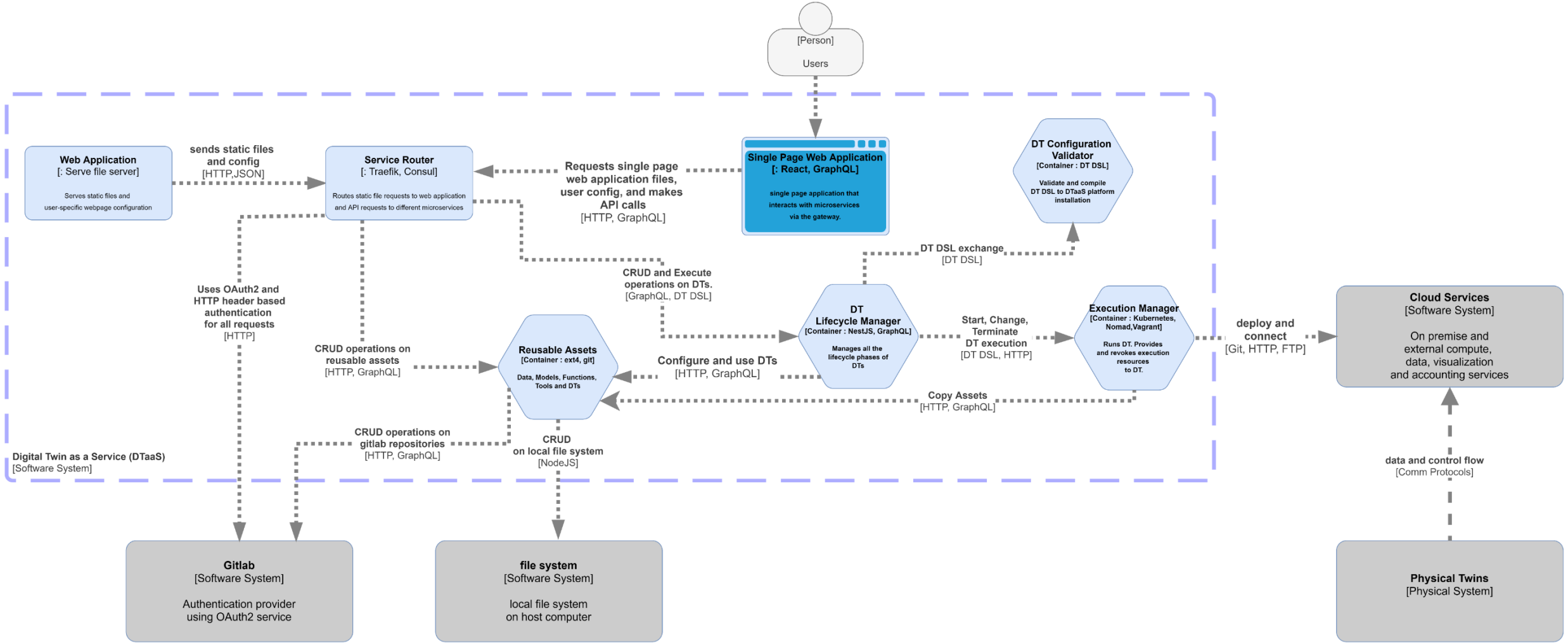
System Component	Responsibilities
DT Lifecycle Manager	Manage digital twins through all their lifecycle stages
Execution Manager	Instantiate and execute digital twins on selected virtualized environments like docker containers, virtual machines, cloud infrastructure.

C4 Level 2 Diagram for Reusable Assets, DT Lifecycle Manager and Execution Manager

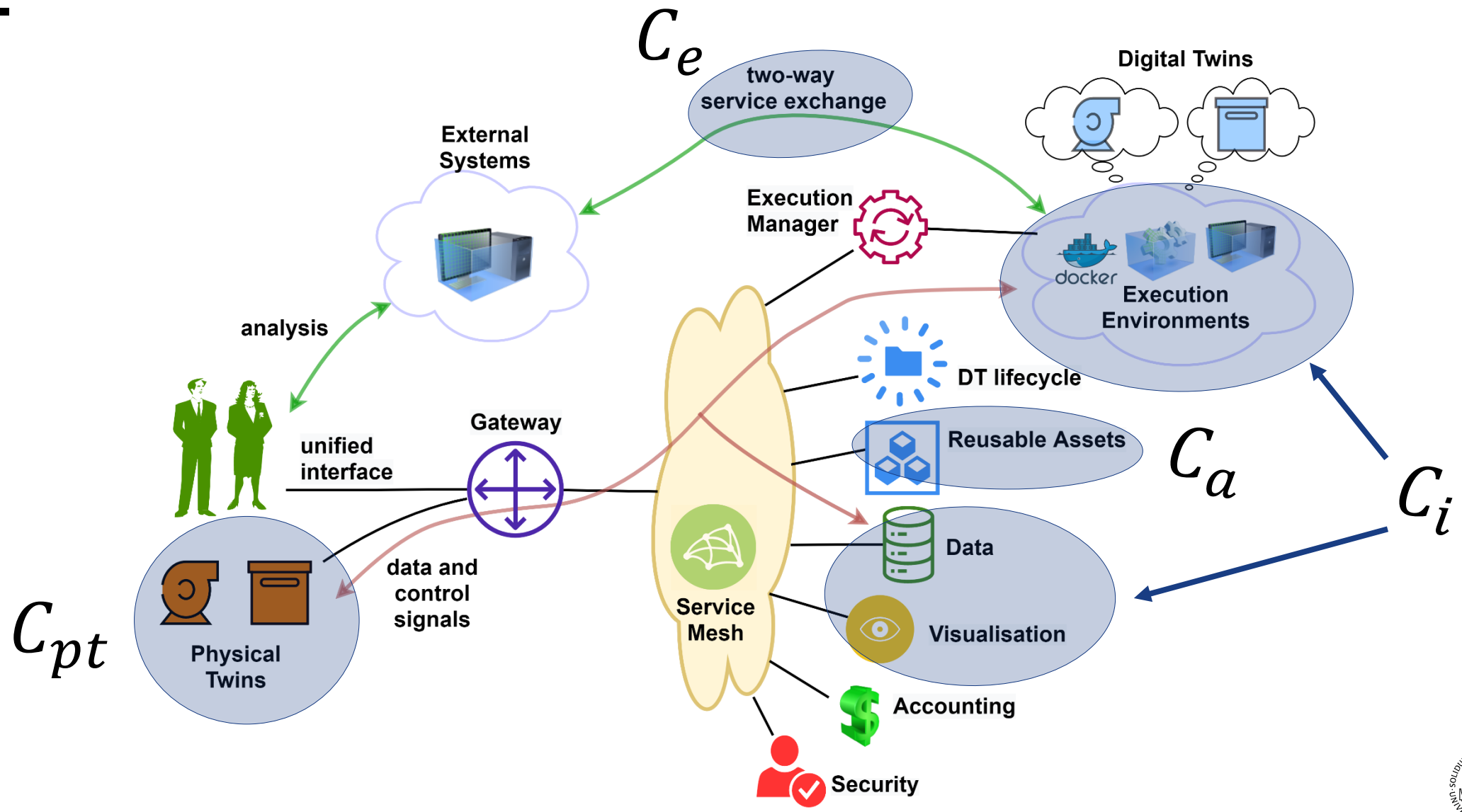


NOTE: This diagram hides interactions with external systems

C4 Level 2 Diagram for Reusable Assets, DT Lifecycle Manager and Execution Manager



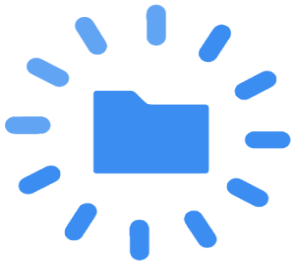
WHICH COMPONENTS ARE SPECIFIED IN C_{dt}?



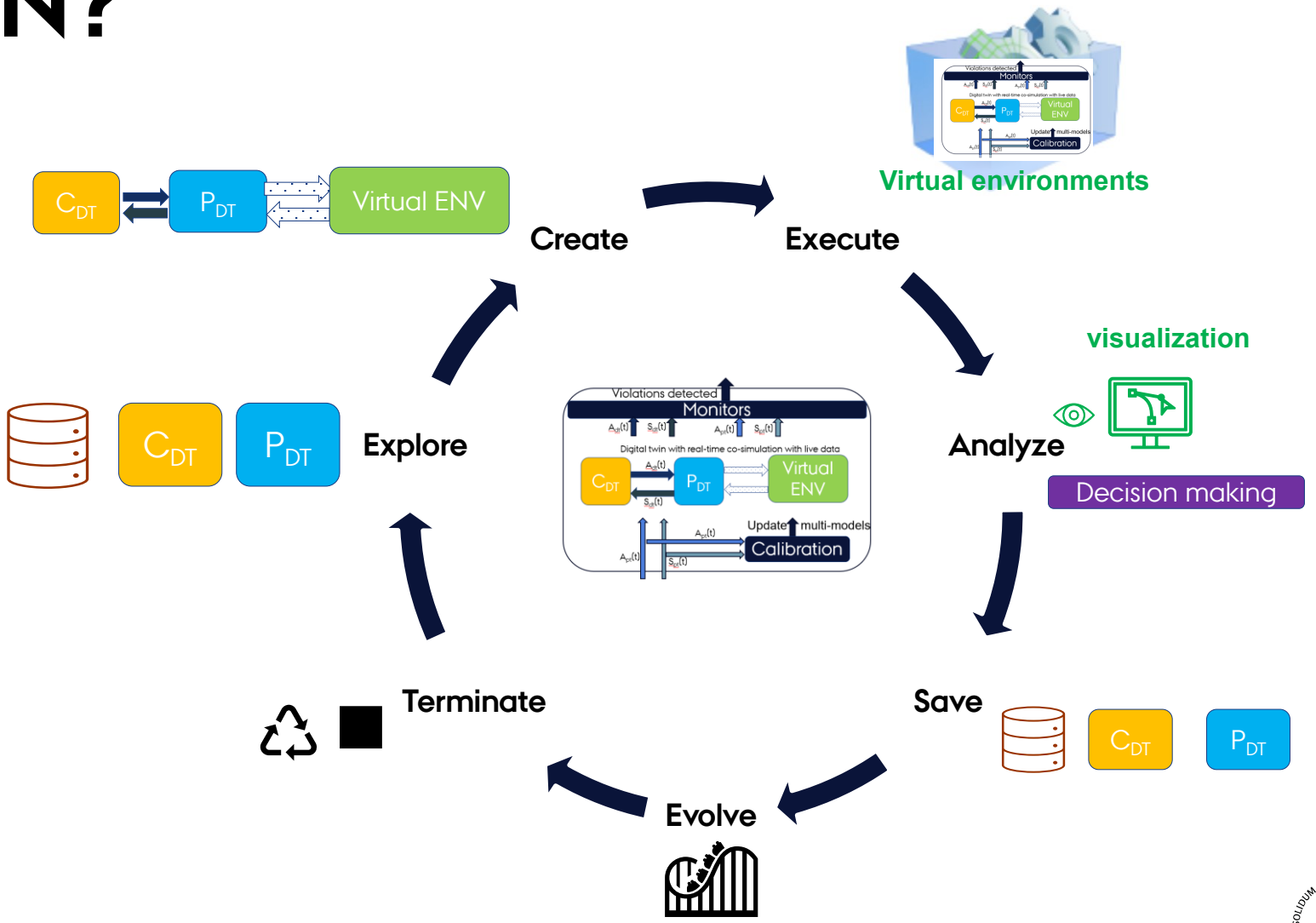
PRESENTATION OUTLINE

1) Requirements for Digital Twin Platforms
2) A Conceptual Framework
3) *DTaaS software platform*
    1) System Architecture and Design
    2) *Digital Twin Lifecycle Manager*
    3) Execution Manager
    4) Comparison with Other Platforms
4) Implementation Status

WHAT ARE DIFFERENT LIFECYCLE PHASES A DIGITAL TWIN?



lifecycle manager



IN DEPTH LOOK AT DIGITAL TWIN LIFECYCLE

Author DT Components (on or off platform)

Consolidate and Explore DT Components (like a market place)

Create / Configure new DT (like a Lego playground)

Execute one DT
(with a click)

Scenario Analysis
(execute many DTs with a click)

Analyze (using data science tools)

Save(any of DT components)

Evolve

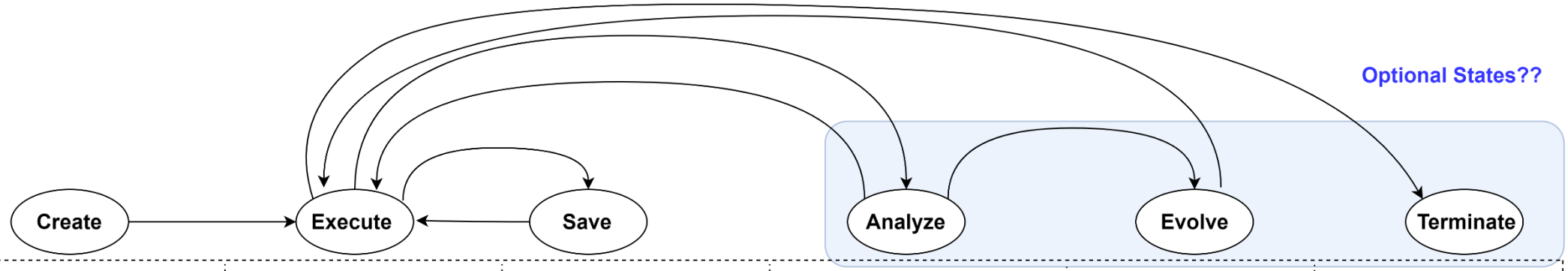
Terminate

What's the cache?

Not so linear transition

LIFECYCLE PHASES FOR INCUBATOR - AN EXAMPLE

What about state transitions and outputs?

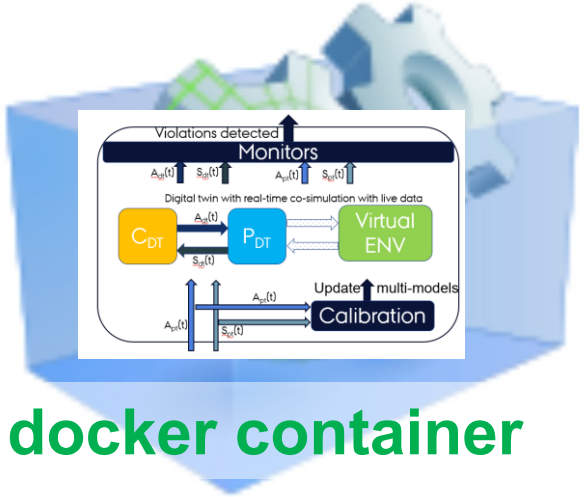


DT Lifecycle Phase	Create	Execute	Save	Analyze	Evolve	Terminate
Activities	Data, models, functions, and tools selected and DT configuration specified.	DT is executing and adds value to the PT. The multiple DT services are running: monitoring the system, storing data, visualising it, and any other services.	The DT is paused. All relevant state is stored, and made ready to restart later. The PT continues to operate but without the support from the DT.	All relevant state of PT and DT are monitored ana analyzed. The PT continues to operate.	The DT is updated thanks to collected insight of its analyze phase. The PT is reconfigured based on analysis results.	All the communication from the DT to the DT services stops. The DT stops execution
Incubator Example	State estimation and anamoly detection are used as single moel. Data input comes from physical incubator and output goes to incubator. Visualisation functions are InfluxDBqueries.	The DT (really just a digital shadow at this point) displays the full state of the system (recorded and estimated)	The DT is paused. No recording of new temperature measurements is done, and no state estimation is done. The PT continues to operate without the support from the DT	When the lid of the incubator is opened, anamoly detector generates alert, the need for recalibration in model is recognised and model calibration starts.	Recalibrated model is deployed in DT.	Communication with the physical incubator stops. The connection to InfluxDB is terminated. The incubator DT process is terminated

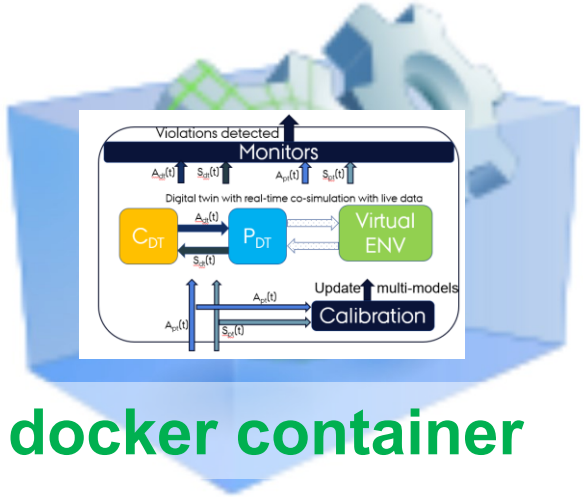
PRESENTATION OUTLINE

1) Requirements for Digital Twin Platforms
2) A Conceptual Framework
3) *DTaaS software platform*
    1) System Architecture and Design
    2) Digital Twin Lifecycle Manager
    3) *Execution Manager*
    4) Comparison with Other Platforms
4) Implementation Status

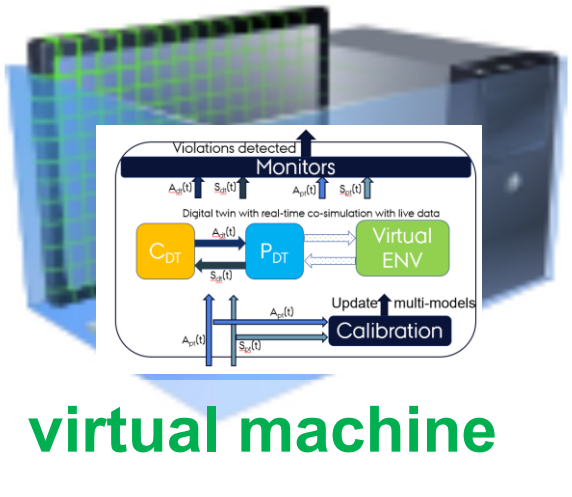
WHAT DOES EXECUTION MANAGER DO?



docker container



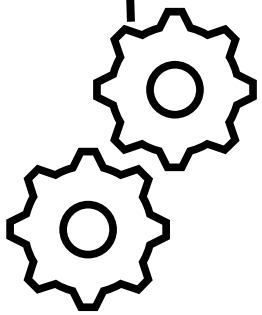
docker container



virtual machine

create and destroy

execution manager



PRESENTATION OUTLINE

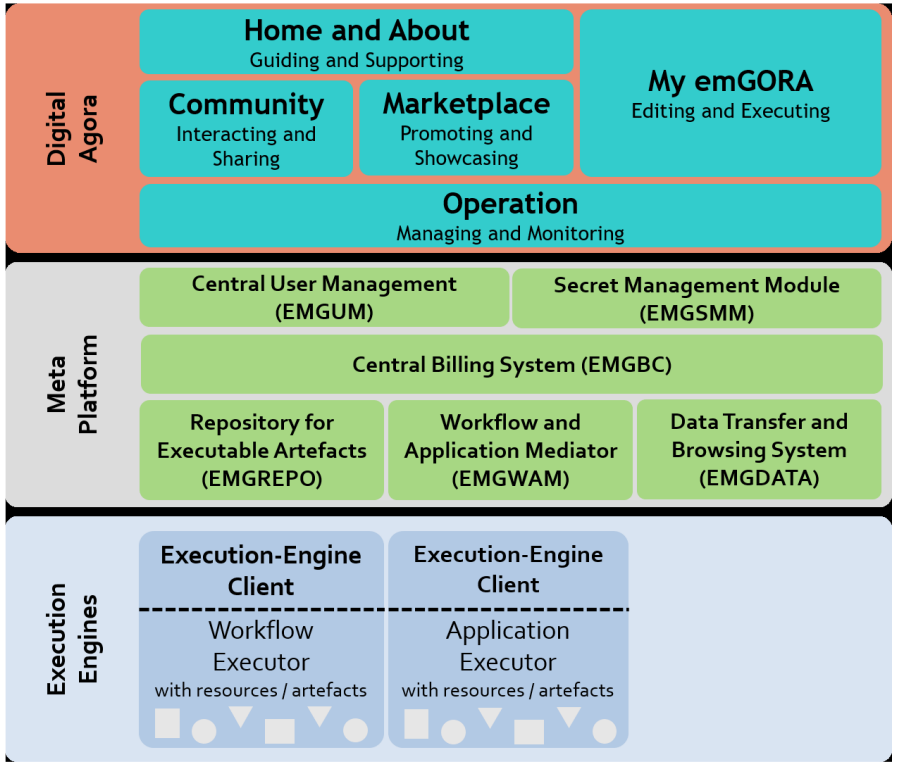
1) Requirements for Digital Twin Platforms
2) A Conceptual Framework
3) *DTaaS software platform*
    1) System Architecture and Design
    2) Digital Twin Lifecycle Manager
    3) Execution Manager
    4) *Comparison with Other Platforms*
4) Implementation Status

Major Advantages of DTaaS

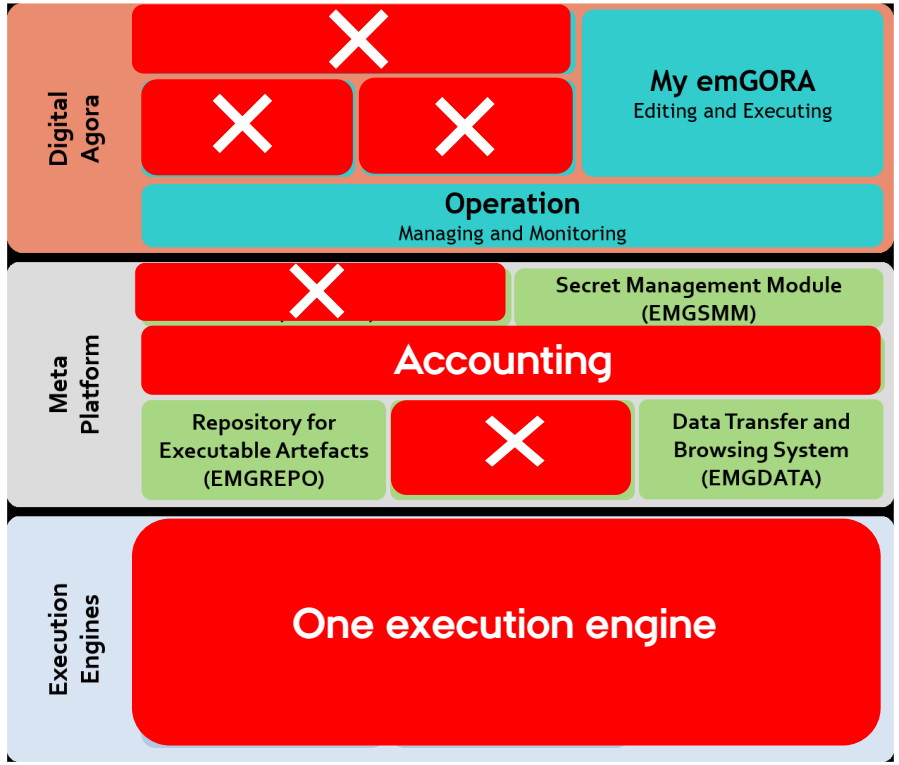
1. Support for different kinds of Digital Twins - CFD, Simulink, co-simulation, FEM, ROM, ML etc.
2. Support other Digital Twin frameworks (need an example here)
3. Facilitate availability of Digital Twin as a service
4. Collaboration and reuse
5. Private workspaces for authoring and verification of reusable assets, trial run DTs
6. Cost effectiveness

Scope vis-à-vis DIGITbrain Project

DIGITbrain



DTaaS



Comparison with Commercial Platforms

Various Parts \ Comprehensive Tools		DTaaS	Predix	PTC's ThingWorx	Siemens MindSphere	ANSYS	Dassault 3D Experience	Foxconn's Beacon
DT evolution	Knowing the physical world	✓				✓	✓	
	Changing the physical world	✓	✓		✓			
Modeling	Geometry modeling	✓					✓	
	Physical modeling	✓				✓	✓	
	Behavior modeling	✓				✓		
	Rule modeling		✓					

Comparison with Commercial Platforms (2)

Various Parts \ Comprehensive Tools		DTaaS	Predix	PTC's ThingWorx	Siemens MindSphere	ANSYS	Dassault 3D Experience	Foxconn's Beacon
DT Data Management	Data collection	✓	✓	✓	✓			✓
	Data transmission	✓		✓	✓			
	Data Storage	✓		✓			✓	✓
	Data processing		✓				✓	✓
	Data fusion		✓				✓	✓
	Data visualization	✓					✓	✓

Comparison with Commercial Platforms (3)

Various Parts \ Comprehensive Tools		DTaaS	Predix	PTC's ThingWorx	Siemens MindSphere	ANSYS	Dassault 3D Experience	Foxconn's Beacon
Services	Simulation Services	✓	✓		✓	✓	✓	✓
	Optimization Services		✓		✓			✓
	Diagnosis and Prognosis Services		✓	✓	✓	✓		✓

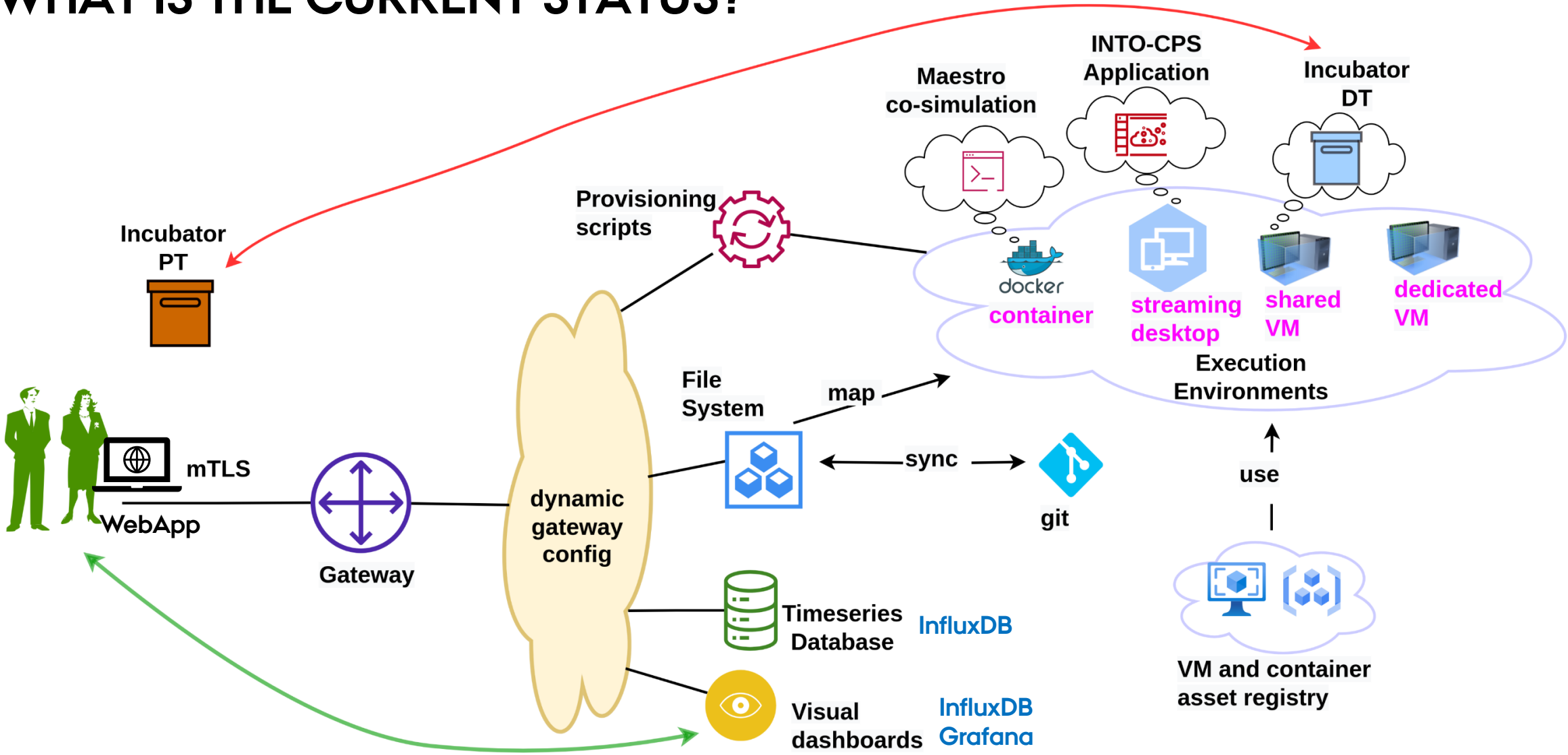
Comparison with Commercial Platforms (4)

Various Parts \ Comprehensive Tools		DTaaS	Predix	PTC's ThingWorx	Siemens MindSphere	ANSYS	Dassault 3D Experience	Foxconn's Beacon
Connections	Connection in digital world	✓	✓		✓		✓	
	Connection between physical and digital world	✓	✓	✓	✓		✓	✓

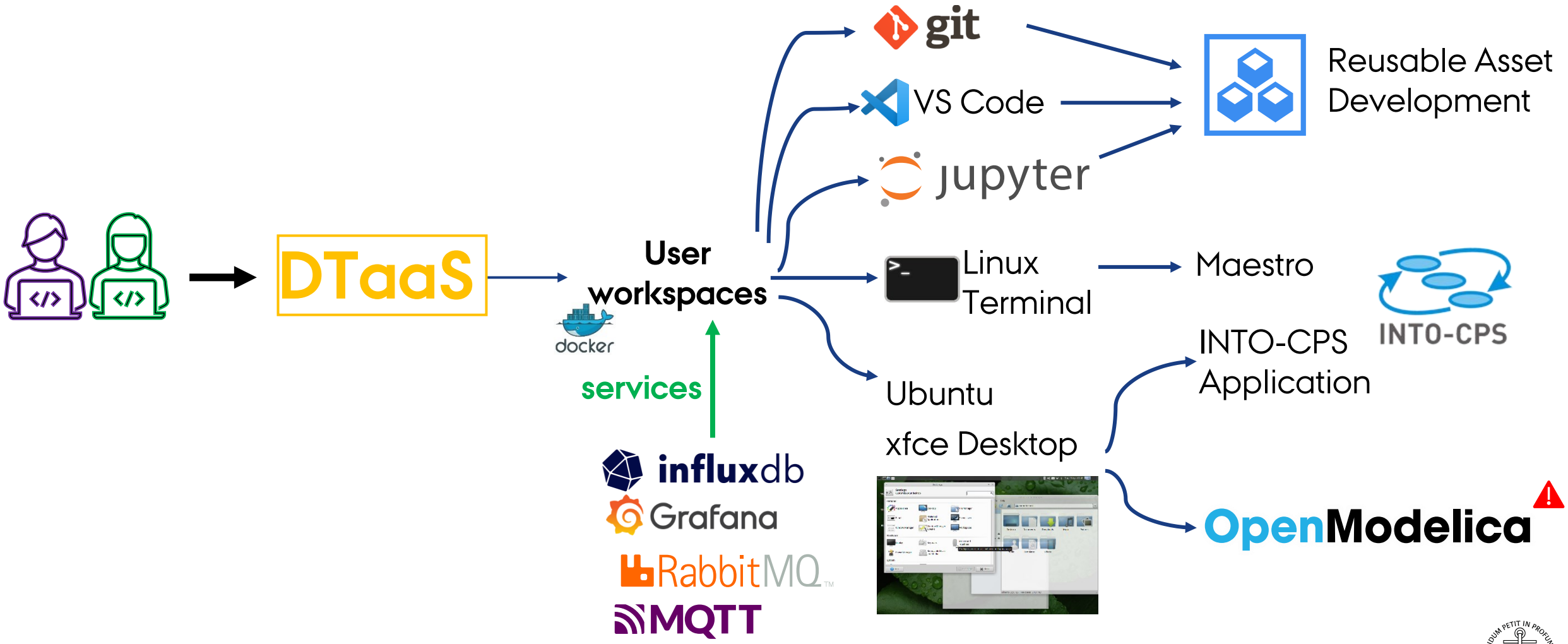
PRESENTATION OUTLINE

1) Requirements for Digital Twin Platforms
2) A Conceptual Framework
3) DTaaS software platform
4) *Implementation Status*

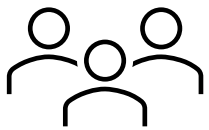
WHAT IS THE CURRENT STATUS?



EXISTING FEATURES OF DTAAS



WORK PROGRESS IN DTAAS PROJECT



DTaaS

Active

- Clean UI
- Software packages
- Multi-user security
- DT Configuration DSL (YAML schema)
- Integration of External Services

Not Started Yet

- Execution Manager
- One click management of DTs
- Lifecycle Manager
- Dynamic virtual machine integration

REFERNCES

- Peter Gorm Larsen, Increasing Dependability of Cyber-Physical Systems by using Digital Twins (Presentation), March 2023.
- John Fitzgerald, Claudio Gomes and Peter Gorm Larsen (Editors), The Engineering of Digital Twins (book draft), May 2023
- Wagg, D., Worden, K. orcid.org/0000-0002-1035-238X, Barthorpe, R. et al. (1 more author) (2020) Digital twins: State-of-the-art future directions for modelling and simulation in engineering dynamics applications. ASCE - ASME Journal of Risk and Uncertainty in Engineering Systems, Part B. Mechanical Engineering, 6 (3). 030901. ISSN 2332-9017.
- Images: Docker, InfluxDB, Grafana, RabbitMQ, MQTT, VSCode, Jupyter, xfce desktop, Linux Terminal, git, OpenModelica
- DIGITbrain deliverables – A EU Horizon 2020 project under grant number 952071.
- Data Ingestion and Processing and platform comparison
  Qi, Q., Tao, F., Hu, T., Anwer, N., Liu, A., Wei, Y., Wang, L., & Nee, A. Y. (2021). Enabling technologies and tools for digital twin. Journal of Manufacturing Systems, 58 (PB), 3–21. https://doi.org/10.1016/j.jmsy.2019.10.001
- Fei Tao, Meng Zhang, Yushan Liu, A.Y.C. Nee, Digital twin driven prognostics and health management for complex equipment, CIRP Annals, Volume 67, Issue 1, 2018, Pages 169-172, ISSN 0007-8506 (5 dimensional digital twin)
- Tekinerdogan, Bedir, and Cor Verdouw. "Systems architecture design pattern catalog for developing digital twins." *Sensors* 20.18 (2020): 5103.

AARHUS
UNIVERSITY