THE ROLE OF BLOCKCHAINS AND SMART CONTRACTS IN DECENTRALIZED INDUSTRIAL ENVIRONMENTS

2022 DEI DOCTORAL RESEARCH SEMINARS

PhD Student: Gaetano Volpe
Laboratory of Control and Automation
Department of Electrical and Information Engineering
Polytechnic University of Bari - Italy

OVERVIEW

- In manufacturing environments, Cloud Computing led to «Cloud Manufacturing», a new paradigm that enables companies to provide and receive services over Internet via an intelligent service-oriented architecture
- The range of potential services include software tasks such as complex calculations, 3D rendering, etc.
- These task need to be completed in a reasonable time to preserve production

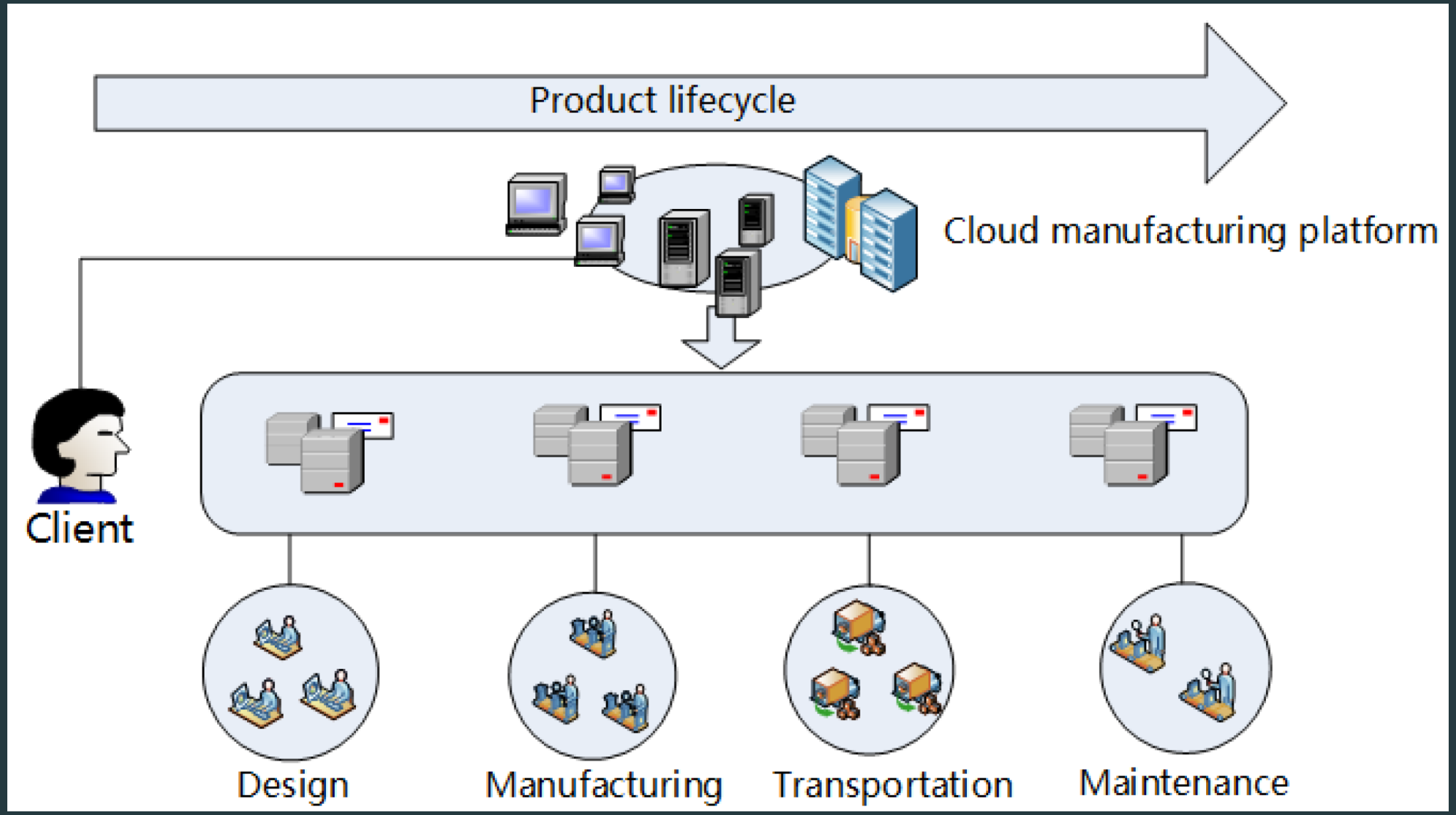
OVERVIEW

- Blockchain is one of last decade's disruptive technologies in distributed architectures
- Smart Contracts are essentially "pieces of code" running on a Blockchain
- Docker is a container-based platform to run applications in isolated environments from lightweight and self-contained images
- Cloud Storage is a well-known technology for storing and retrieving data in the Cloud

OVERVIEW

- Our research concerns a novel distributed architecture for consuming digital processes in a manufacturing environment based on Blockchain, Smart Contracts, Docker & Cloud Storage
- The system is depicted and explained in terms of components, actors and use cases
- A task assignment problem solved with a deep learning approach is also suggested
- Finally, a case study in ophthalmic industry is provided

THE CONTEXT – CLOUD MANUFACTURING



Hu Y, Zhang Z, Wang J, Wang Z, Liu H. Task Decomposition Based on Cloud Manufacturing Platform. *Symmetry*. 2021; 13(8):1311. https://doi.org/10.3390/sym13081311

THE CONTEXT

- There are two preliminary key factors in cloud environments:
  1. Decentralization: to improve reliability, scalability and performance
  2. Security: to prevent data stealing, alteration and preserve integrity
- Blockchain, thanks to its unique features, already satisfies these requirements

THE PROBLEM

- Solutions presented in the related literature are weak in the development of virtualization and servitization technology and they fail to fit the actual enterprise requirements mainly in terms of collaboration
- While implementing, scaling or moving a process, there are often many constraints in terms of programming languages, application dependencies and operating environments
- A standard framework is needed through which the different actors could provide, update and consume any process, in terms of a service, in a more effective way without need of complex reconfigurations

NEW CONTRIBUTIONS

A decentralized, distributed, secure and collaborative platform to consume digital processes in a Cloud Manufacturing environment featuring:

1. An Ethereum blockchain based architecture with dedicated smart contracts
2. Docker integration to efficiently encode any software logic in the process lifecycle and preserve integrity
3. Cloud Storage integration to store and retrieve process inputs and outputs
4. A Deep Learning Process Run-time prediction algorithm to select the “fastest” runner for a task in a specific time band

NEW CONTRIBUTIONS – MAIN ADVANTAGES

EXISTING BLOCKCHAIN ONLY BASED SOLUTIONS

- Decentralization
- Reliability
- Security

BLOCKCHAIN + DOCKER + RUN-TIME PREDICTION

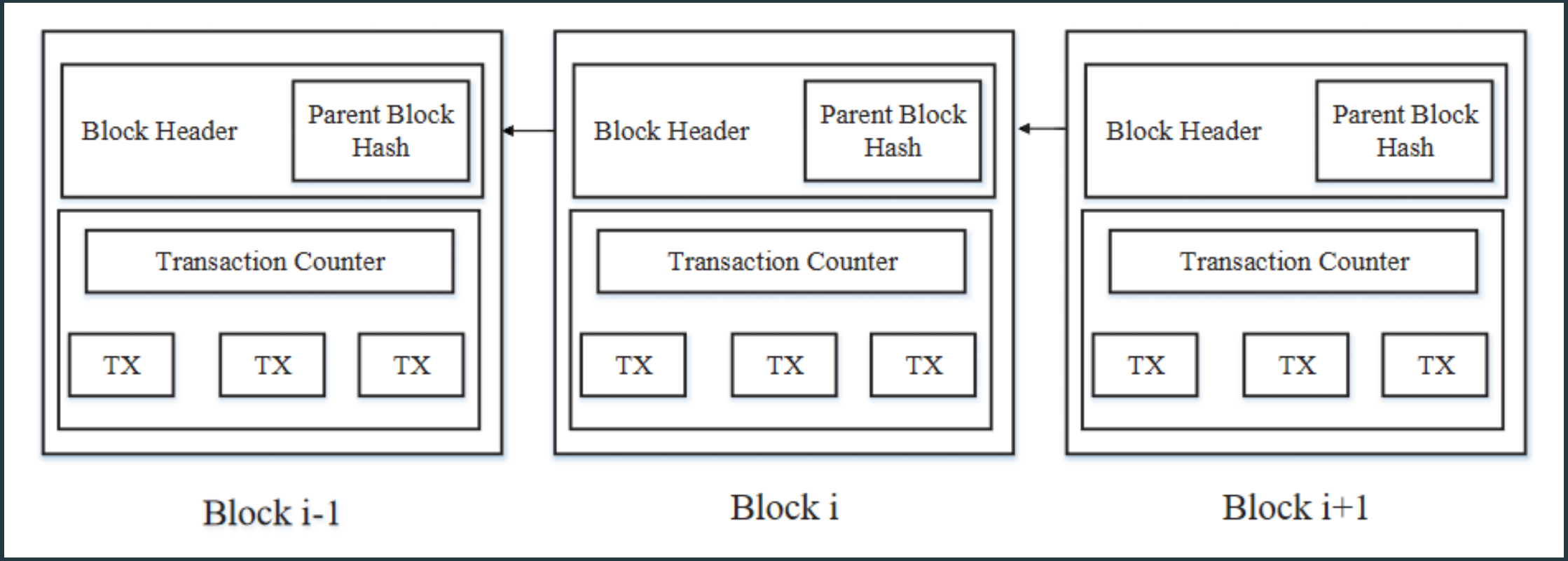
- Decentralization
- Reliability
- Security
- Flexibility
- Performance

G. Volpe, A. M. Mangini and M. P. Fanti, "An Architecture for Digital Processes in Manufacturing with Blockchain, Docker and Cloud Storage," 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), 2021, pp. 39-44, doi: 10.1109/CASE49439.2021.9551633.

BLOCKCHAINS

- Blockchain is a “distributed ledger” technology which is used to record transactions without the need of a central Authority
- Each transaction is signed at least by the issuer and it is verifiable by the nodes participating in the peer-to-peer network
- Transactions are grouped in blocks and each block is connected to the previous one by including its Secure Hash Algorithm (SHA) SHA-256 in the header, thus creating a tamper-proof chain
- All nodes must agree to which transactions are valid and the order in which they have to be stored in the ledger, that is the “consensus” concept

BLOCKCHAINS

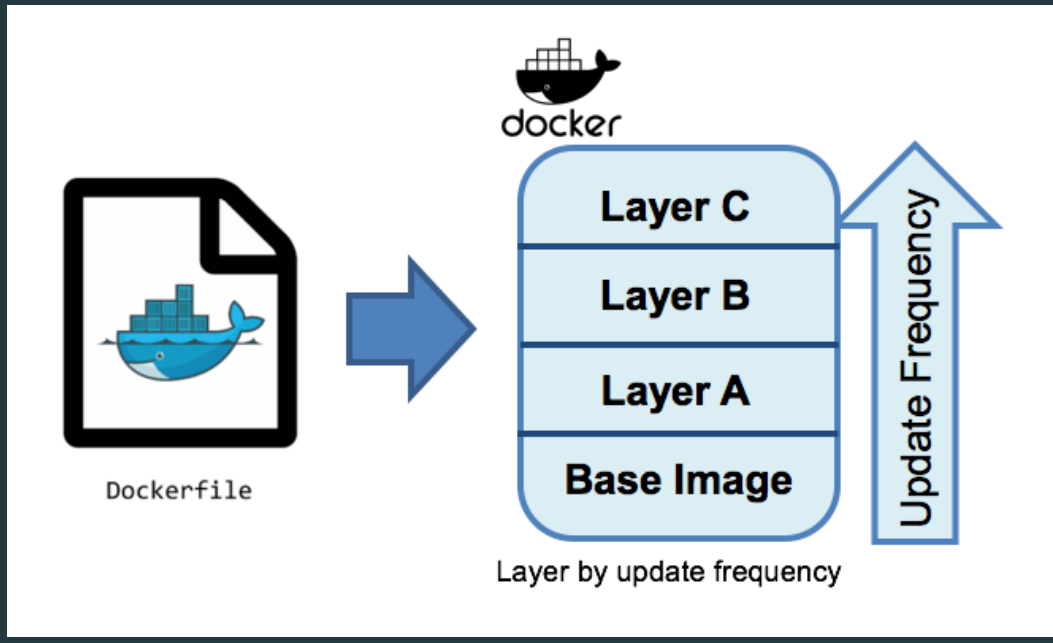


ETHEREUM & SMART CONTRACTS

- Ethereum is an open-source blockchain for creating and using smart contracts and distributed apps
- Ethereum has its own cryptocurrency (ETH)
- A Smart Contract is a decentralized program code deployed in the Blockchain that enforces the terms and conditions of a specific agreement between two or more parties
- SC is executed automatically when triggering conditions are satisfied.
- Two of the most active supported languages for smart contracts are Solidity and Vyper

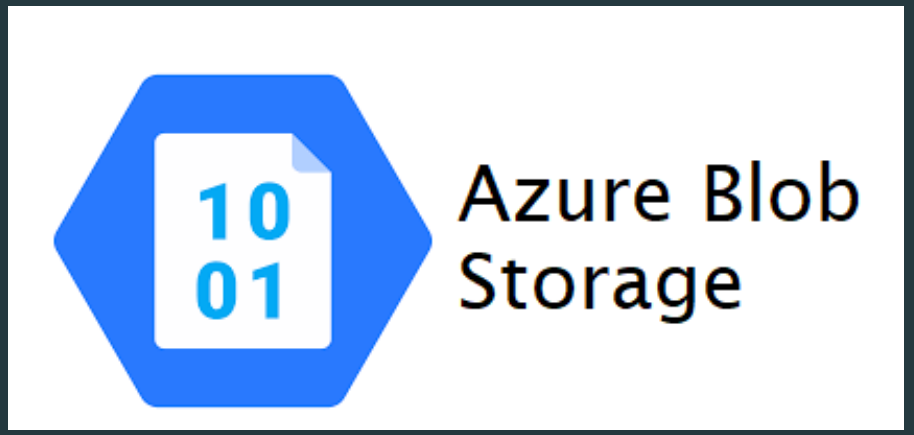
DOCKER & CONTAINERS

- Docker is a platform to run an application in an isolated environment called container
- A hypervisor is not required. The container runs within the host machine's kernel
- The platform provides a client-server architecture to handle the container life-cycle
- An image is a template built on the top of a Dockerfile
- Each instruction in Dockerfile adds a layer to the image filesystem
- Each image could be identified by its SHA-256 hash, the digest, which is calculated on the top of all layers
- A registry is a public or private repository hosting images



CLOUD STORAGE

Cloud Storage is a data storage service. It is one of the most popular cloud technologies and it is offered by some of the major cloud service providers

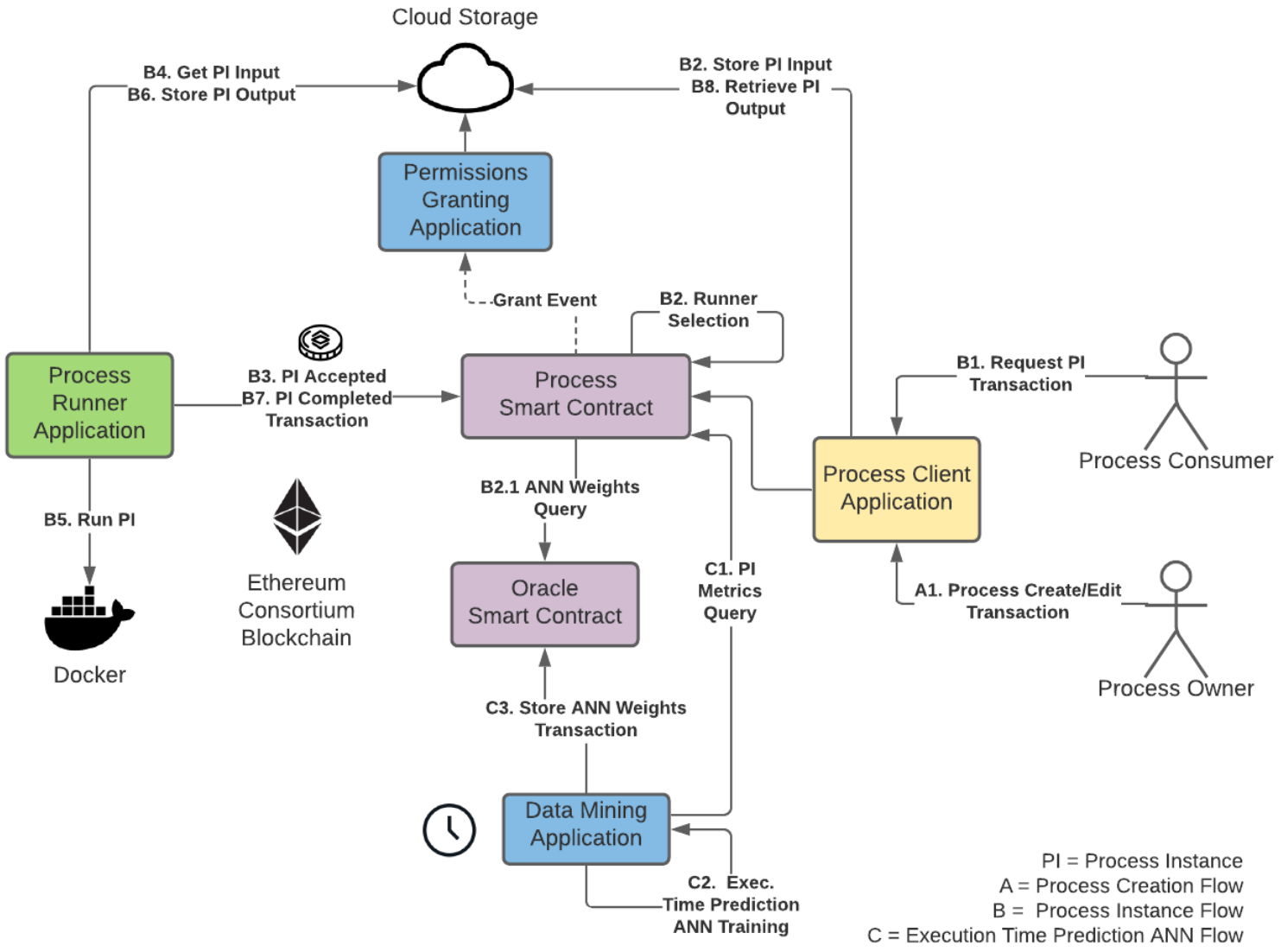


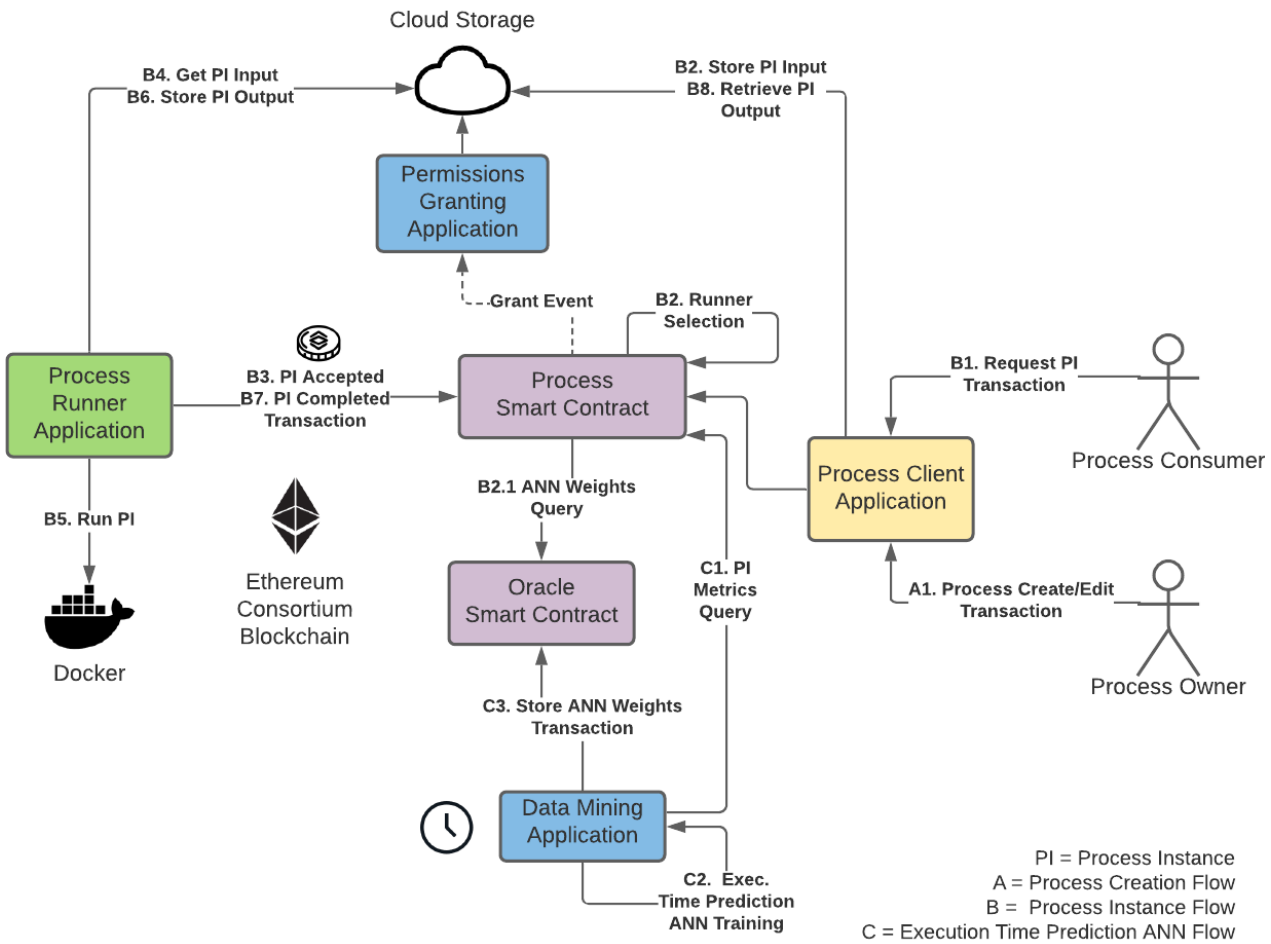
DIGITAL PROCESSES

- In our research, we define a digital process as a task that can be carried out using a software algorithm
- It is designed to take one or more files as input and produce as many files as output
- For example: scientific workflows, 3D rendering, etc.

THE PROPOSED SYSTEM

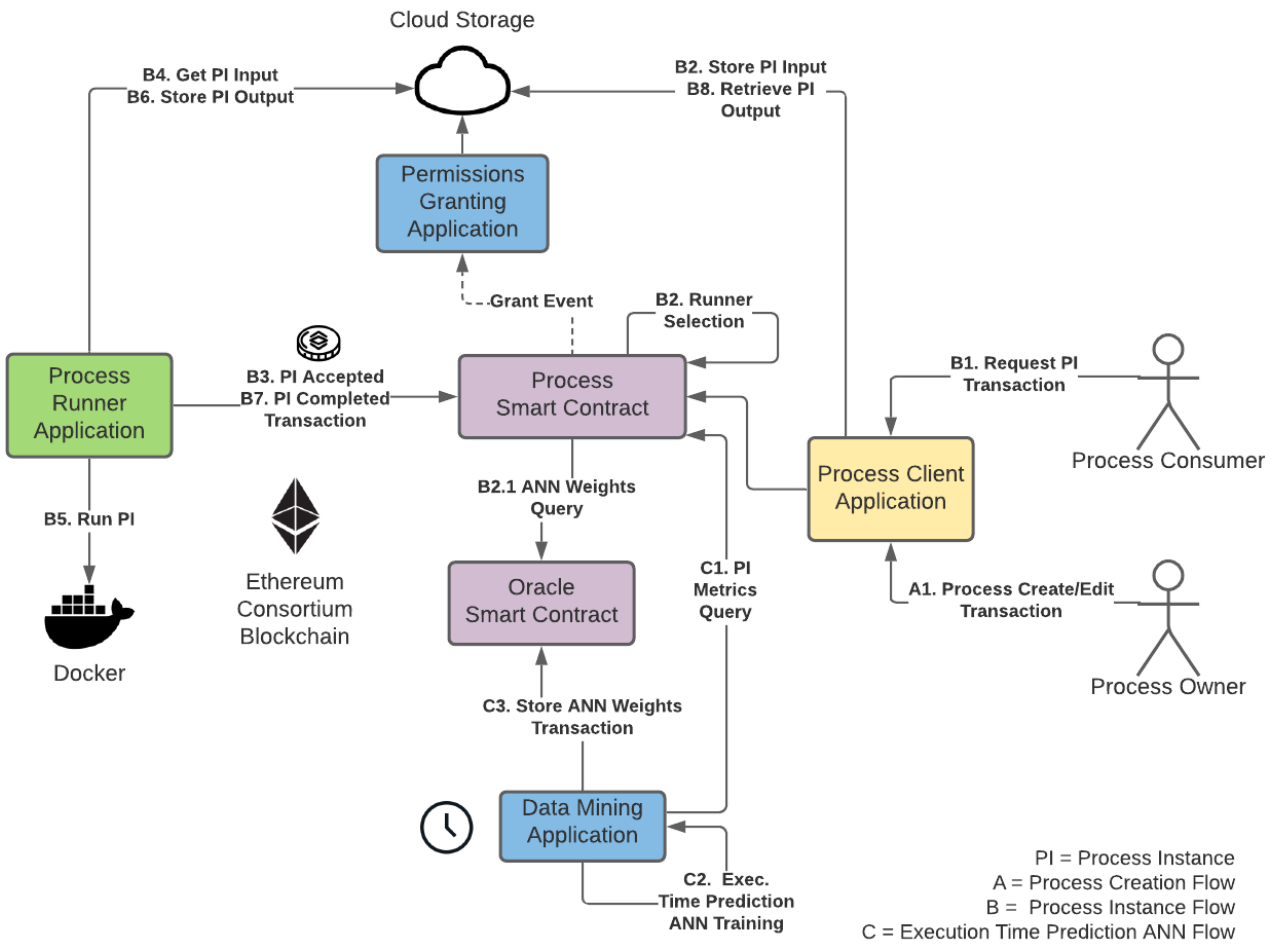
- Ethereum Blockchain & Smart Contracts for process implementation & monitoring
- Docker for task execution
- Cloud Storage to store and retrieve task inputs & outputs
- ANN to predict task run-time
- PoA (Proof-of-Authority) consensus





THE PROPOSED SYSTEM – AT A GLANCE

- Five Actors:
  - Process Owner
  - Process Consumer
  - Process Runner
  - Permissions Granting Application
  - Data Mining Algorithm
- Three main use cases:
  - Process Creation Flow
  - Process Instance Flow
  - Execution Time Prediction Flow



THE PROPOSED SYSTEM – AT A GLANCE

- Two Smart Contracts:
  - Process Smart Contract
  - Oracle Smart Contract
- Four Edge Applications:
  - Process Client Application
  - Process Runner Application
  - Permissions Granting Application
  - Data Mining Application

THE PROPOSED SYSTEM – ACTORS

1. Process Owner: the «creator» of the process logic (the algorithm and the related code and Docker image)
2. Process Consumer: the «consumer» of the process, e.g. the manufacturer
3. Process Runner: the «provider» of computational resources to run an instance of the process (run a container with the selected image)
4. Permissions Granting Application: orchestrates permissions to access files on cloud storage platform
5. Data Mining Algorithm: trains time prediction ANNs for runners and processes and store weights on Blockchain

THE PROPOSED SYSTEM – USE CASES

- Create Process: the Process Owner codes its algorithm, packs everything in a Docker images, pushes it to a Docker Registry and finally creates the process in the Blockchain by executing «CreateProcess» transaction
- Consume Process: the Process Consumer requests a new instance by executing «ProcessInstance» transaction, the Smart Contract assigns the fastest runner using the last calculated ANN models and the instance is finally executed
- Run-Time Prediction: on a regular time basis, past instance metrics are collected by the Data Mining Application and new models are trained for each runner and for each process

THE PROPOSED SYSTEM – PROCESS SMART CONTRACT

- Handles the whole lifecycle of a Process (Create, Consume, Update, Delete)
- Provides *CreateProcess* and *ProcessInstance* transactions
- Stores relevant data of a process such as the owner, the Docker image name and the current hash value
- Process runners are required to use only the image set by the owner and identified by the stored hash value
- Queries the Oracle Smart Contract for ANNs weights to select the fastest runner for an instance

THE PROPOSED SYSTEM – PROCESS SMART CONTRACT

Global Data

```
pragma solidity 0.8.1;

contract Process {
    enum ProcessInstanceState
    {Placed, Assigned, Completed, Refused}
    uint public totalProcesses;
    uint public totalProcessInstances;
    Process[] public processes;
    ProcessInstance[]
    public processInstances;

    struct Account {
        int256 balance;
        bool accountType;
    }
    struct Process {
        string dockerImage;
        // Docker SHA-256 Hash
        bytes dockerId;
        uint clickFee;
        address owner;
        ProcessRunner[] runners;
    }
    struct ProcessRunner {
        address runner
    }
    struct ProcessInstance {
```

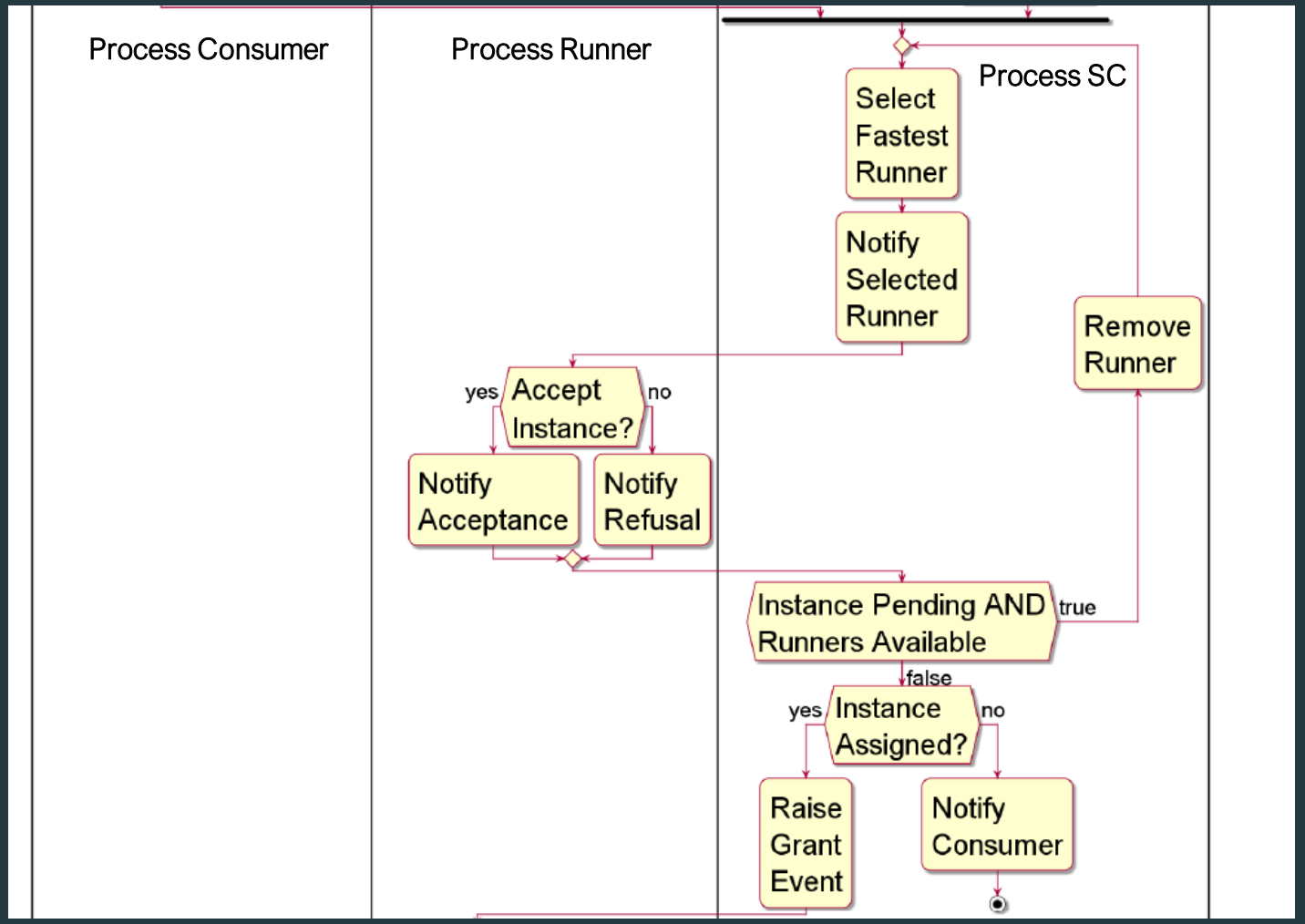
Process Data

```
        uint processId;
        address runner;
        address consumer;
        uint executionTime;
        uint executionFee;
        uint totalFee;
        bytes inputId;
        bytes outputId;
        ProcessInstanceState state;
    }
    function createProcess(...)
    function processInstance(...)
    ...
```

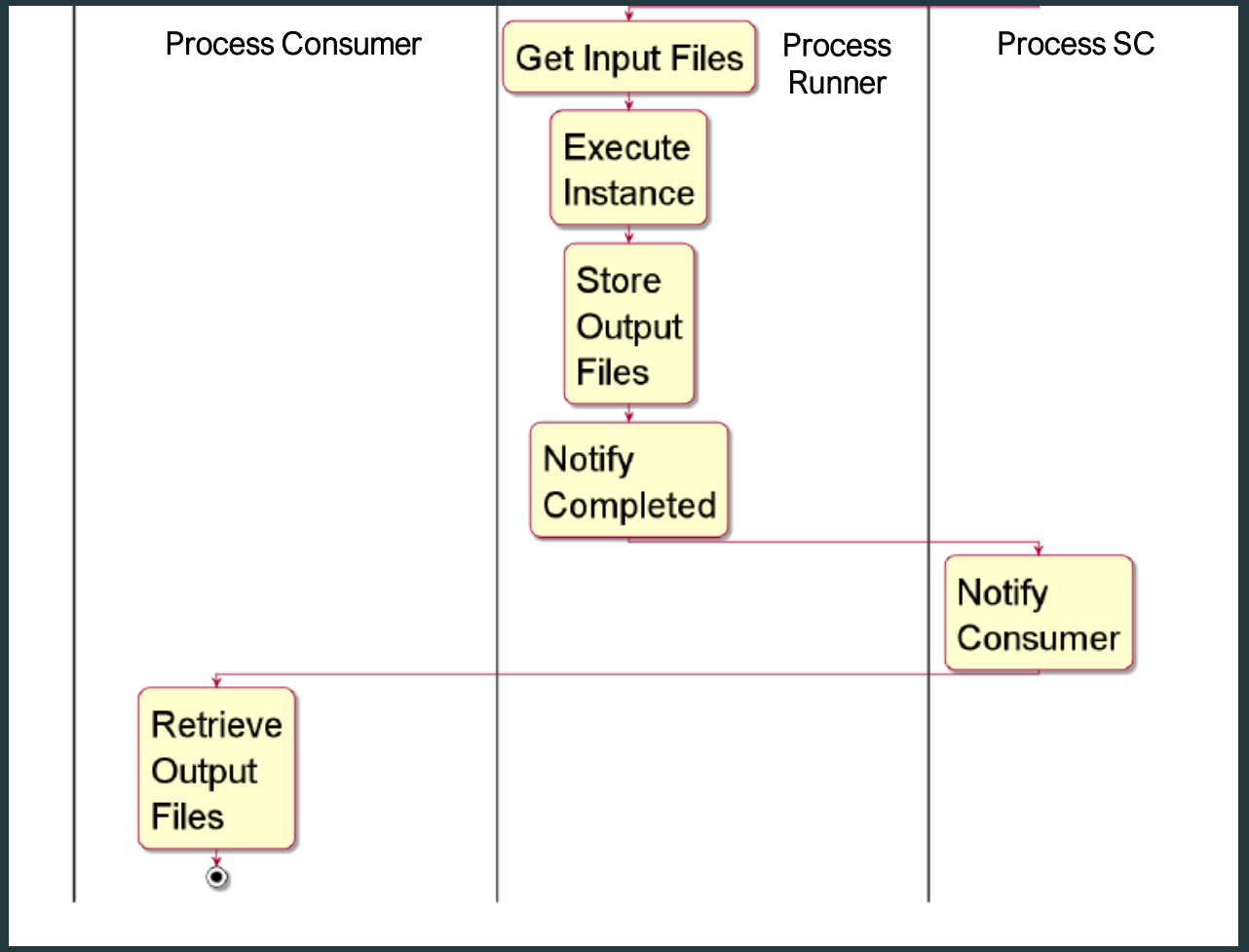
Instance Data

THE PROPOSED SYSTEM – PROCESS INSTANCE TRANSACTION - UML ACTIVITY DIAGRAM - 1

THE PROPOSED SYSTEM – PROCESS INSTANCE TRANSACTION - UML ACTIVITY DIAGRAM - 2

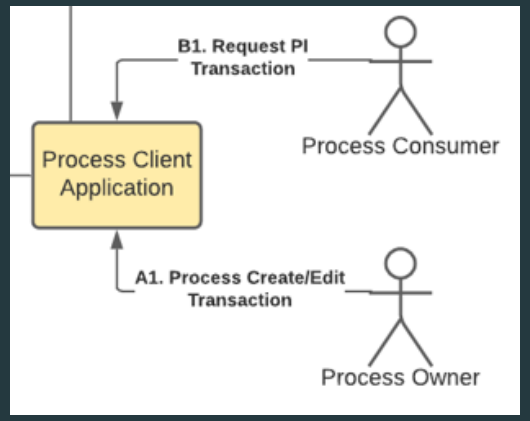


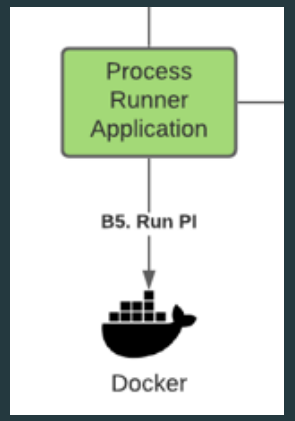
THE PROPOSED SYSTEM – PROCESS INSTANCE TRANSACTION - UML ACTIVITY DIAGRAM - 3

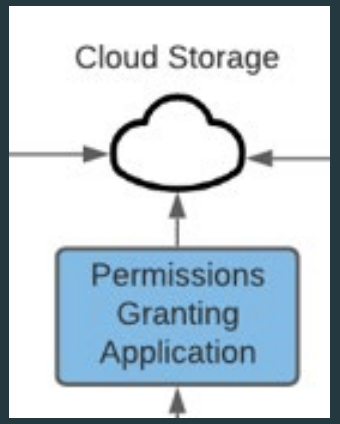


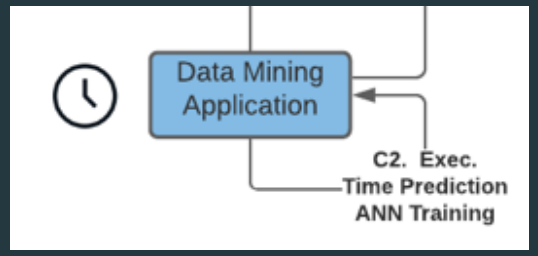
THE PROPOSED SYSTEM – ORACLE SMART CONTRACT & EDGE APPLICATIONS

- Oracle Smart Contract: stores the calculated weights of each ANN model. The use of such special smart contract makes the task assignment problem deterministic and compliant with the Blockchain architecture
- Edge Applications: they interact with the Blockchain and Smart Contracts. They can be coded in any compatible language (Java, .Net, Go, etc.)



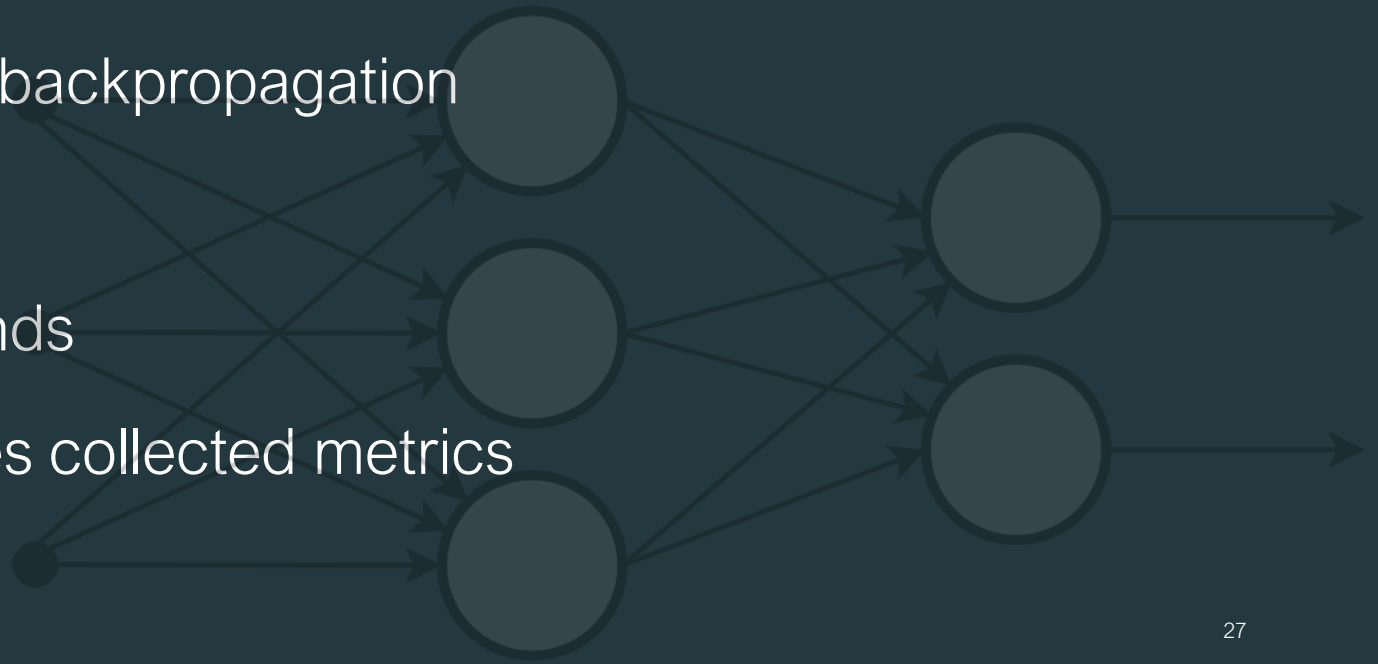






THE PROPOSED SYSTEM – TIME PREDICTION ANN

- The goal: predict instance run-time given a process, a runner and a time band
  - Instance is assigned to predicted fastest runner
- Type: Artificial Neural Network with backpropagation
- Inputs: Hour of day, Day of Week
- Output: predicted run-time in seconds
- Training/Test Dataset: past instances collected metrics



OPHTHALMIC LENSES CALCULATION – A CASE STUDY – THE CONTEXT

CALCULATION

Surfacing

Polishing

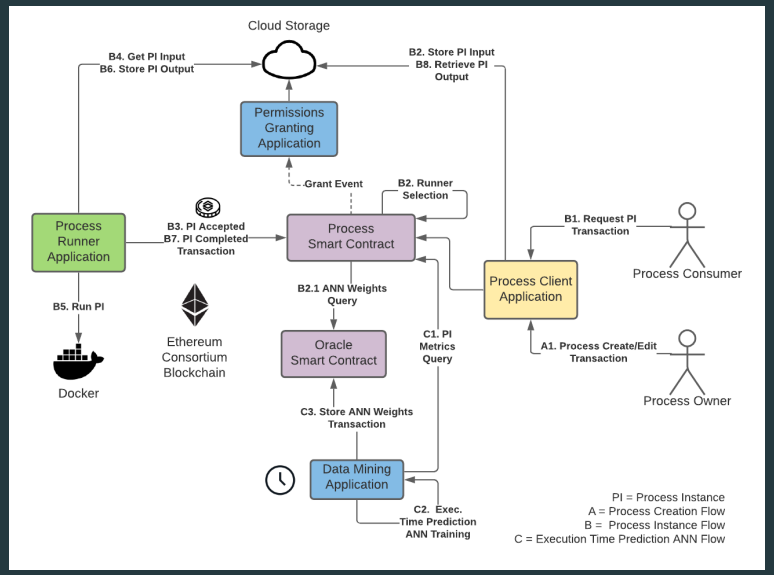
Coating

Edging

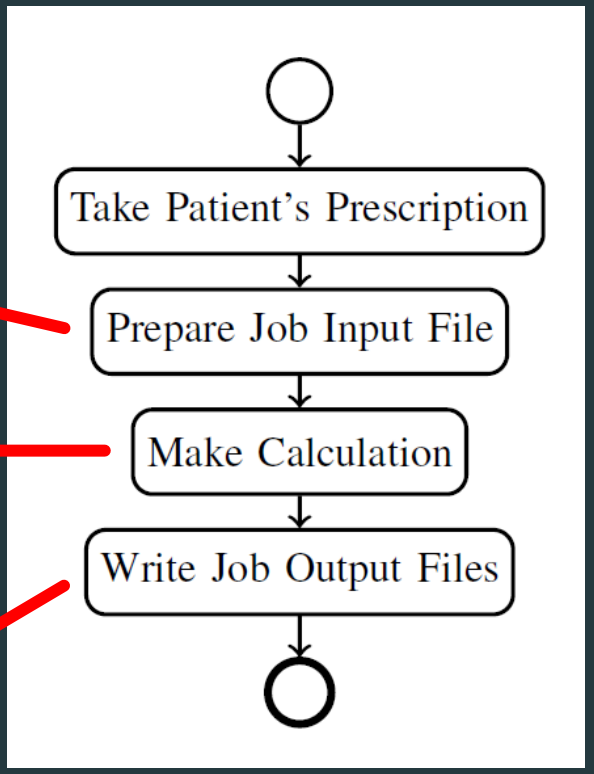
Final Spectacles

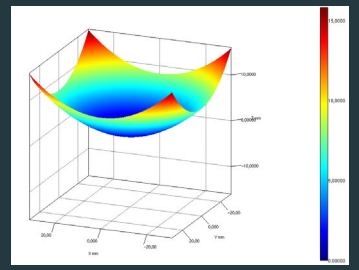
- ✓ Calculation is a full «digital process»
- ✓ The process calculates the back surface of the lens
- ✓ Calculation time varies depending on lens type: single vision, progressive, etc…
- ✓ Process is defined as LDS (Lens Design System)

OPHTHALMIC LENSES CALCULATION – A CASE STUDY – THE IMPLEMENTATION



JOB=1111
DO=B
REQ=LDS
LNAM=LensDesignName;LensDesignName
SPH=+1.00;+1.00
CYL=+0.0;+0.00
AX=0.0;0.0
ADD=2.0;2.0
CRIB=60;60
LIND=1.5;1.5
FRNT=5.00;5.00
MINEDG=1.0;1.0
…..





REQ=SDF
JOB=1111
SURFMT=1;R;B;85;85;84;84
…
ZZ=16.688;16.176;15.682;15.206; …
ZZ=16.310;15.800;15.308;14.833; …
…

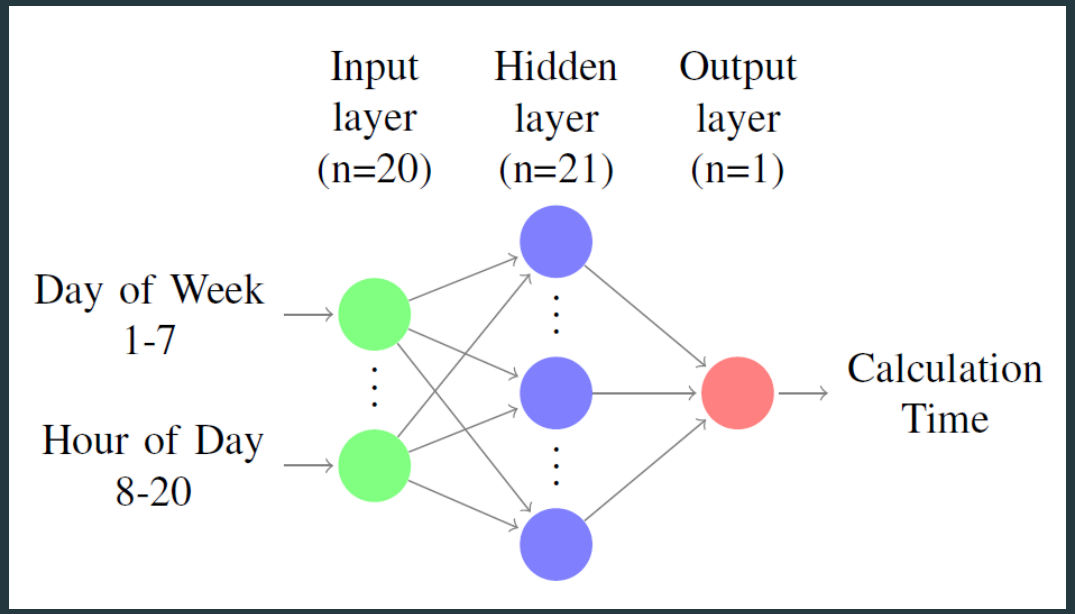
JOB=1111
DO=B
REQ=LMS
LNAM=LensDesignName;LensDesignName
CRIB=60;60
LIND=1.5;1.5
GBASEX=-4.43;-4.43
GCROSX=-4.60;-4.60
……

OPHTHALMIC LENSES CALCULATION – TIME PREDICTION ANN EXPERIMENT

- Training/Test Simulation Dataset:
  - One Progressive Lens Design
  - Two reference calculation times: 7s (SD=0.5) under normal state and 15s (SD=1.0) under busy state (simultaneous processes running on the runner)
  - 200 random samples/hour calculated with normal distribution, 8.00-20.00, Mon-Sun
  - Two-hours heavy load time band randomly selected for each weekday for which busy state parameters are used. Normal state parameters used elsewhere
  - 18.200 total samples

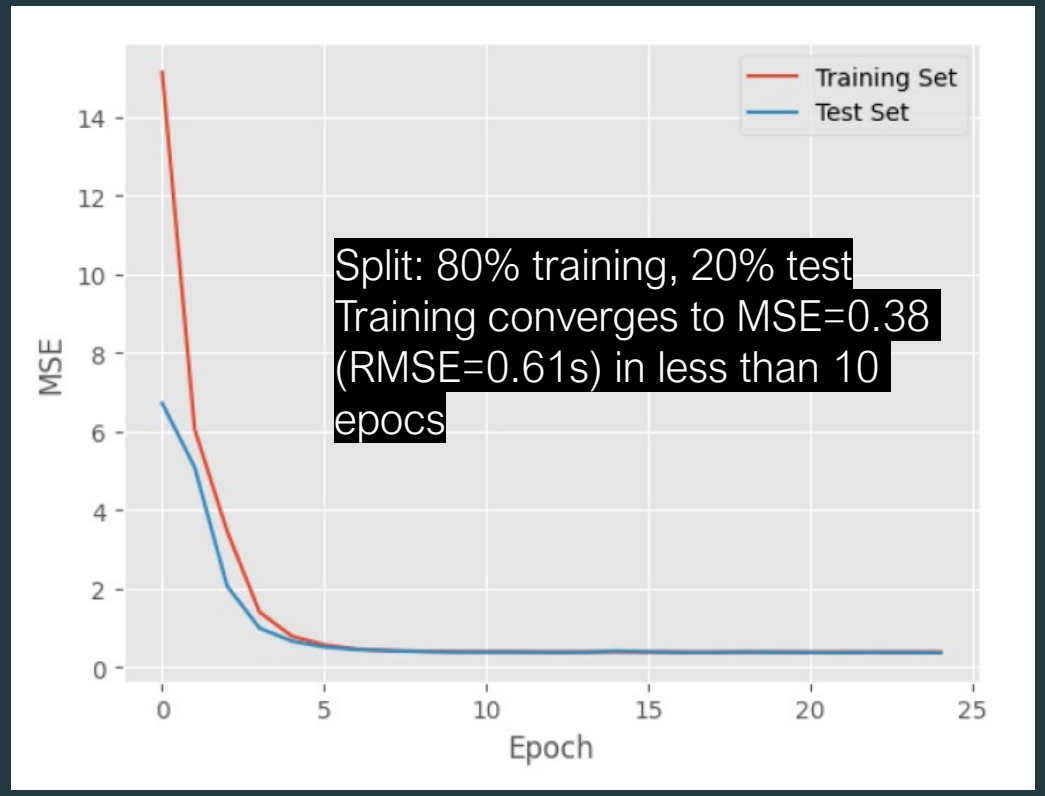
OPHTHALMIC LENSES CALCULATION – TIME PREDICTION ANN EXPERIMENT

One-Hot Encoding used for Input Parameters (010…)

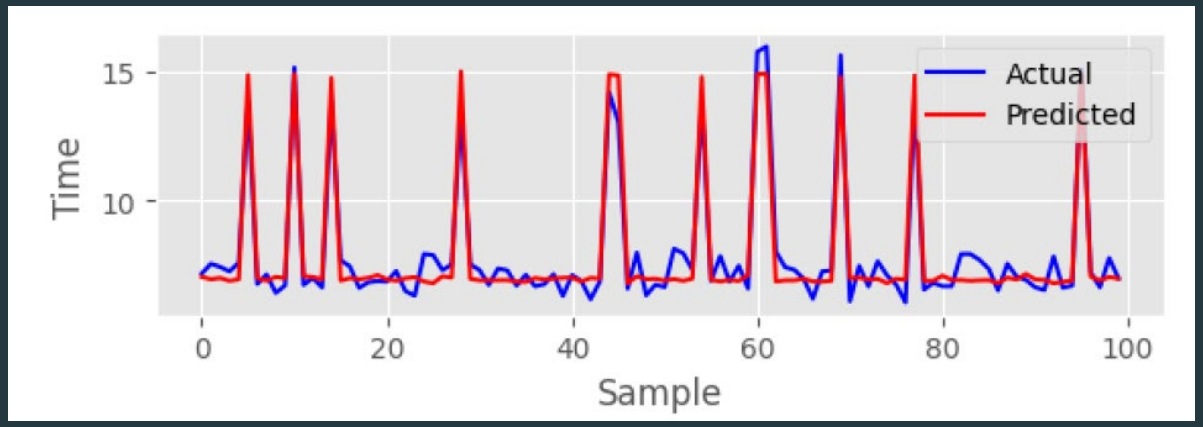


- ✓ MLP Network built with one hidden layer
- ✓ MSE (Mean Squared Error) loss function
- ✓ ReLu (Rectified Linear Unit) activation function
- ✓ Adam optimizer
- ✓ Keras/Tensorflow implementation in Python

OPHTHALMIC LENSES CALCULATION – TIME PREDICTION ANN EXPERIMENT



Split: 80% training, 20% test
Training converges to MSE=0.38 (RMSE=0.61s) in less than 10 epocs



The model predicts run-time with sufficient accuracy and catches peak times

OPHTHALMIC LENSES CALCULATION – THE ADVANTAGES

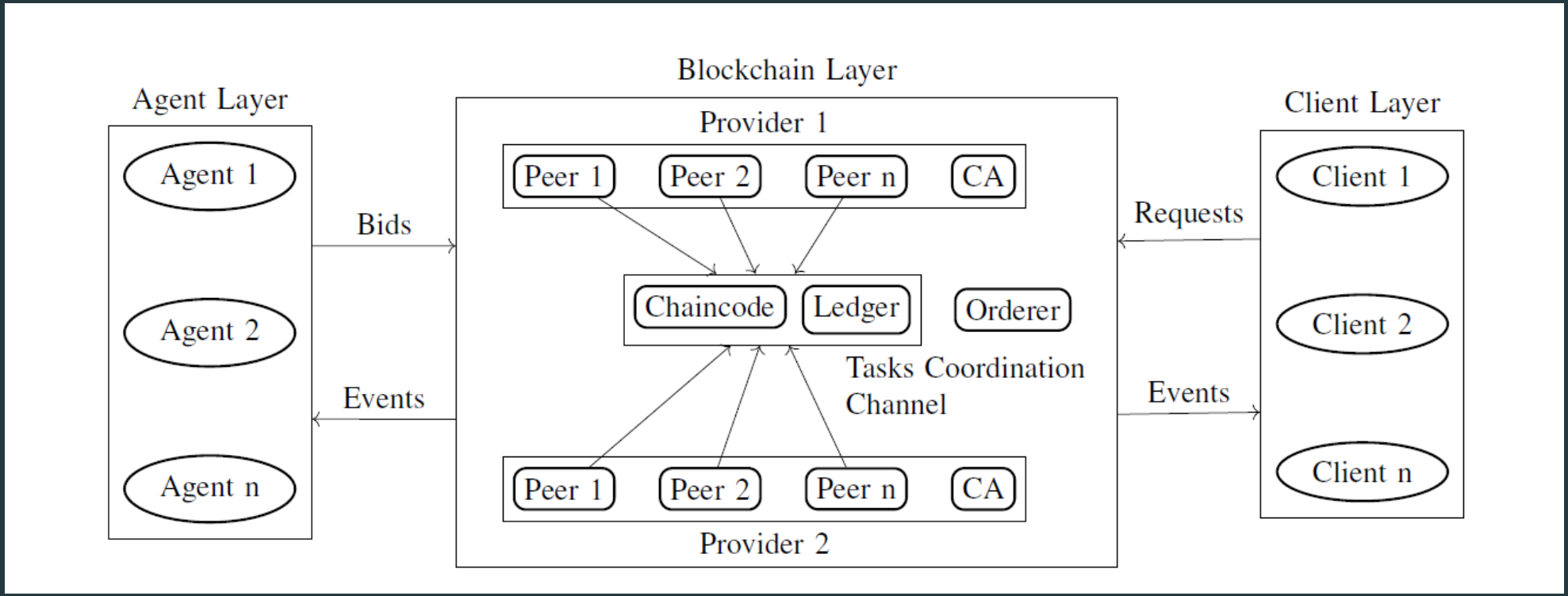
Existing solutions:

- No flexibility in adding new designs
- Calculation queue
- Software and Hardware constraints
- Low reliability
- Low security level as transactions are stored in a single database

Proposed System:

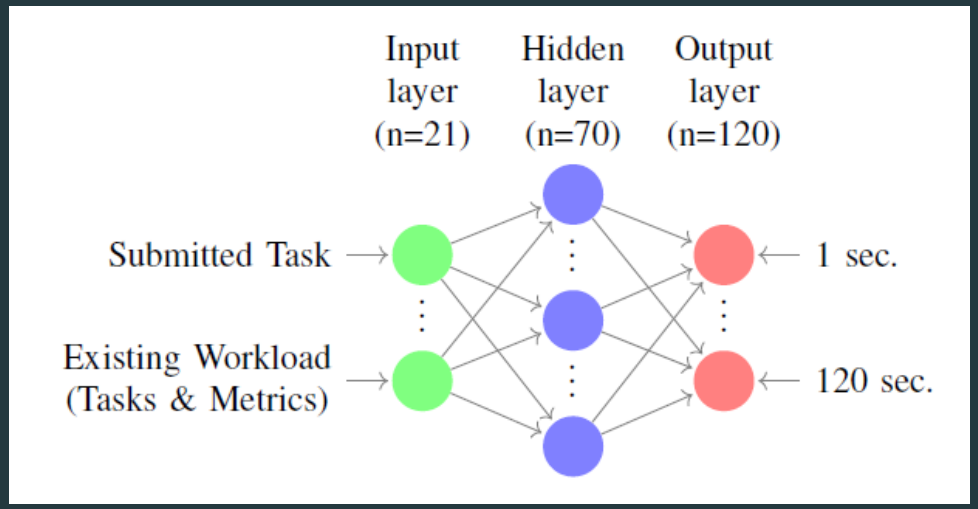
- Collaboration:
  - New designs simply added as processes with Docker images
  - Only one client
- Reliability and Performance:
  - Calculations are executed in a parallel fashion
  - Always fastest runner is selected
  - Distributed system is tolerant to runner faults
- Security:
  - Blockchain is tamper-proof in nature
  - Each blockchain node holds a copy of the ledger
  - ERC-20 token integration for economic transactions

ALTERNATIVE APPROACH – HYPERLEDGER FABRIC & DEEP REINFORCEMENT LEARNING

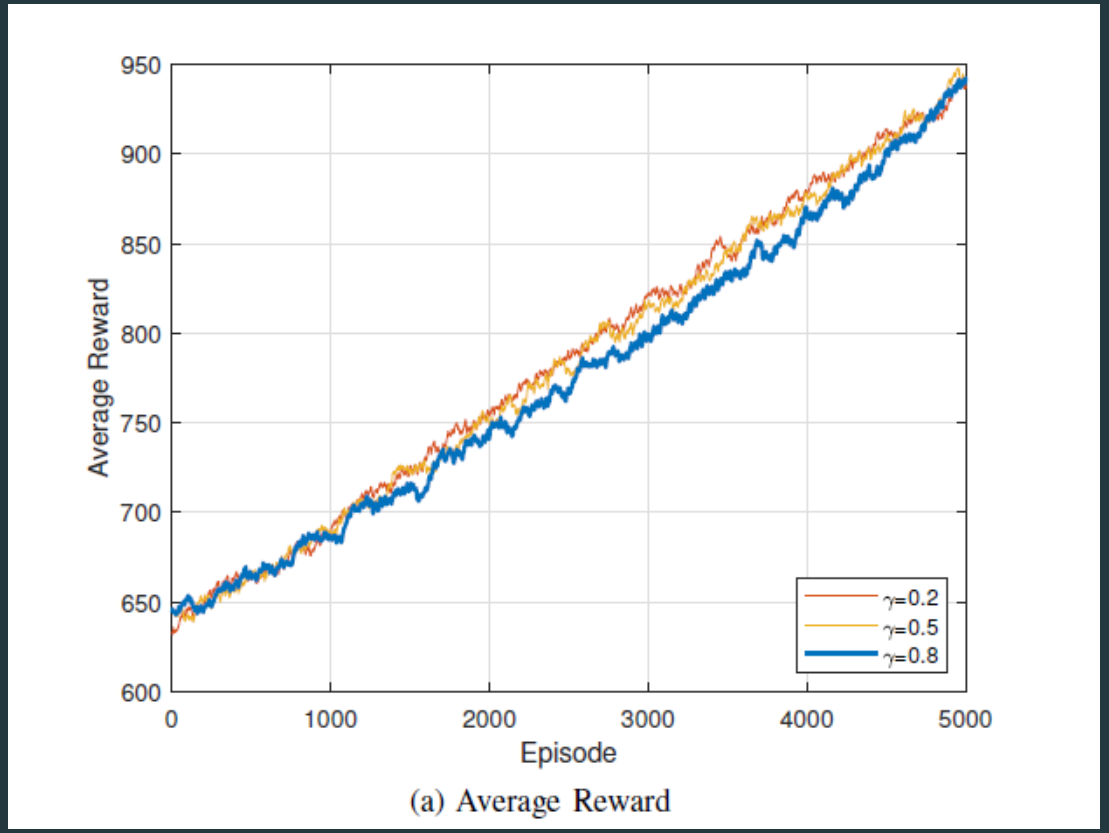


Thanks to its modularity and flexibility, Hyperledger Fabric is more indicated for industrial applications and consortium networks

ALTERNATIVE APPROACH – HYPERLEDGER FABRIC & DEEP REINFORCEMENT LEARNING



- We consider the case in which the agents can provide multiple simultaneous calculations
- We leverage on DRL to have agents «learn» how to predict task runtime based on their current workload



(a) Average Reward

CONCLUSIONS & FUTURE WORK

- A novel distributed architecture for consuming digital processes in a manufacturing environment based on Ethereum Blockchain, Smart Contracts, Docker & Cloud Storage has been introduced
- Advantages related to the integration of Blockchain and Smart Contracts with Docker technology have been explained to face the weakness of existing solutions in terms of collaboration and flexibility
- A time-prediction algorithm with deep learning approach has been also introduced to predict instance run-time for a process on a runner and solve a task assignment problem
- A practical case study in the ophthalmic lens industry has been presented
- The efficiency of time prediction algorithm and the advantages of the implementation of the proposed system over existing solutions in terms of Collaboration, Reliability, Performance and Security have been illustrated
- Future work: implementation on different Blockchain platform, task assignment problem improvement, analysis of transactional privacy and intellectual property problems

THANKS FOR YOUR ATTENTION