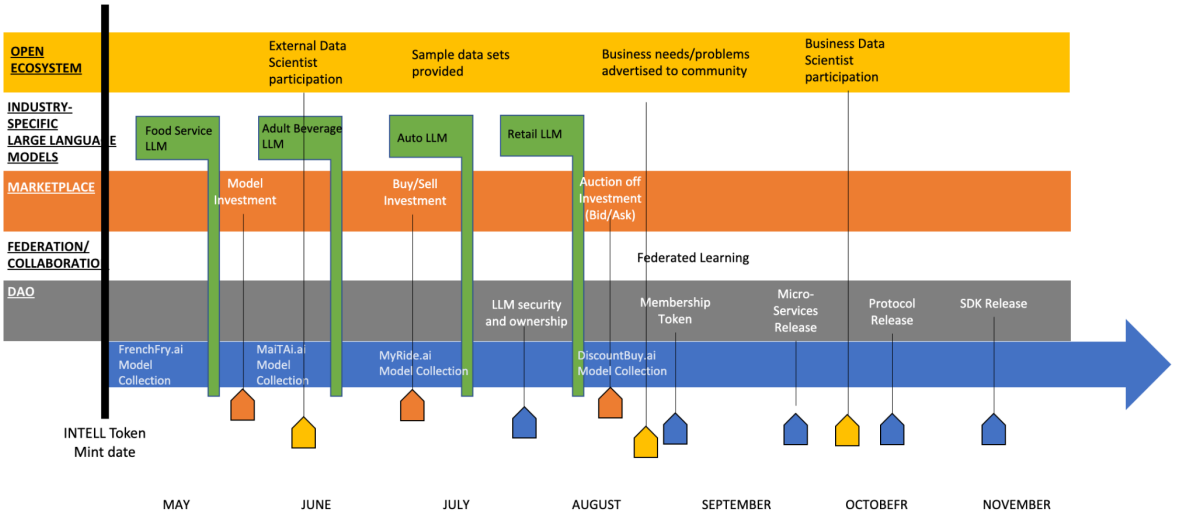
The Intelligence Exchange ramp-up to Web3

[DAO Factory Scenarios]

Phase 1 - Minting, Buying & Selling

As a preliminary project (Phase 1) we will have the minting and creation of NFTs for model creation and updates made by a subsidiary who owns trained models. This tokens are expected to be minted around May, having room until August for deploying the DAO.



INTELL COIN (ERC-20) exist as credits for using the various platforms as well as buying NFTs. They can be bought using FIAT using credit card or other systems.

Each model deployed will be represented by an NFT following the ERC-721 or ERC-1155 token standard.

Marketplace

This Phase 1 cover two use cases: A DataScientist or somebody has a model and he wants to register ownership. Second, a third-party has intelligence and want to register it.

This phase also includes design and building of an NFT marketplace (**TIEX Exchange**)with the following abilities:

- Sign-In and Sign-up with metamask or walletconnect
- See my Models
- Deploy a new Model
    - Define a version
    - Define data lineage
    - Define metadata
    - Define provenance
    - Define a royaltie fee
    - Define how many editions / shares
    - Define collaborators and collaboration % for splits
    - Define ensembling and bundling rules
    - Define how valid and accurate a model is, should be a % of accuracy. This will be updated over the time by performance metrics.
    - Define ongoing performance through performance monitoring
    - Define model performance across multiple customers
    - Define how to access the model, it can be through a UI or an API
    - Define a connected data structure (i.e., this is canonical data structure)
        - It will allow to define how to access. It can be through a UI or an API
    - Define model or data provenance and pedigrees
    - Replacing one or more existing models as an evolution. Two or more models can be combined into a super-model producing a new model.
        - Internally it will burn the old(s) NFT Model contract(s) and mint a new one. The new model should still replicate and consider the splits and rewards from combined Models unless there are changes of each collaboration.
- Buy a Model share
- Sell a Model share
    - Establish a specific model for selling with price. It doesn't support offers.
- Withdraw INTELL COINs
- Buy INTELL COINs
- List available models

Access platform

On a separate platform we should provide model accesses, most probably we are talking about an API with special authentication process and usage.

It needs to cover the usage of cryptocurrency to pay gas and usage fee and also using credit card or other FIAT methods.

Platform contains functionality to support:

1. Access Control
2. Data management infrastructure (“Ecosystem integration:
3. Registration
4. Pre-processing
5. Model Initialization
6. Model Distribution
7. Model Evaluation
8. Model Deployment
9. Performance Monitoring
10. Value measurement and revenue sharing
11. Data and Model Governance
12. Continuous Improvement
13. Open source and cloud services

One way to keep entry barriers at the minimum is that the user buys credits (INTELL COINS) (montly or when needed) that we internally convert to tokens (cryptocurrency) (not withdrawable) that are next used for paying fees and services access.

These platforms should be also baseline for Phase 2 providing minimum features such us Sign-up and Sign-in.

Phase 2 - DAO

DAO Roadmap and minimum items.

[1]

A short explainer of a DAO and how it will work with “**TIEX Exchange**”.

Definition. Decentralized Autonomous Organization. It’s a new kind of organization that allows running an organization among many collaborators with a common objective. Organizations exist in many forms for decades but in this case, they are formed can allow faster arrangements and decisions as well as actions in a way that was never seen before.

[1] For the purposes of this paper, we are drawing a clear line between the token(s) and infrastructure tied to owning a model, and the DAO. The DAO is represented by TIEX tokens

DAOs can leave entirely in “the blockchain” however there is still a need of connecting with existing instruments to act in the off-chain world.

- A DAO is ruled by its contributors and community according to the laws established by code through Smart Contracts.
- A DAO is funded by cryptocurrency, usually, the one that rules the blockchain (ie. ETH for Ethereum blockchain).
- A DAO is governed by its founders and supporters through a governance vote and voting power given by the utility token created.

TIEX TOKEN (ERC 20 token) is the utility token for participating and being part of the TIEX DAO, it will enable partners to vote over any proposal leveraging their voting power. Partners will be able to buy TIEX TOKENS with INTELL COINS.

The TIEX DAO

The purpose of the TIEX DAO is to manage all activities related to the federation of data for the purposes of creating artificial intelligence and sharing that intelligence across a business ecosystem.

The DAO is not creating models or enabling the creation of the model. The DAO is in the federation business. Everything related to Model creation will be solved and integrated with Phase 1.

TIEX DAO is the infrastructure needed for federation and collaboration around intelligence to happen.

Not TIEX DAO nor TIEX (company) have any DS that work directly, the DAO foundation is the one that request TIEX (company) to work . The DAO has the objective of fostering organic activity among and between ecosystems collaborators and participants.

The TIEX DAO provides infrastructure for the protocols, processes and services to enable the following:

1. Users and access management
    a. Creation
        i. A user will need to define
            1. Profile
                a. Business administrator
                    i. Will be able to invite collaborators and if they can execute or read.
                    ii. Can manage collaborators of his business
                    iii. Needs to be validated
                b. Datascientist/Developer

                    i. Can read
                    ii. Can execute (will require a business administrator approval)
                c. Investor
            2. Name & contact information
            3. Wallet address
            4. Funds / INTELL COIN credits
        b. Users can be grouped as part of a business
        c. Management
            i. List his Models
            ii. List his ecosystems
            iii. List his associated businesses.
            iv. Dashboard
                1. Income
                2. Usage
2. Ecosystem management
    a. Creation
        i. An ecosystem will need:
            1. A name
            2. Shortname for the contract (internal use at the beginning)
            3. List of participants. It can be started by a single partner or many, those who create are considered and marked as founders.
            4. Description of purpose
            5. A minimum fee of XXX INTELL COINS and TIEX Token by each participant. This will be used for creating the contract internally.
            6. Type of ecosystem, open or closed.
            7. Type of governance, controlled or open. In case of controlled, controllers need to be selected among the participants.
        ii. Once created, it will automatically list all public models owned by its members
        iii. New members can be added at any time if defined
    b. Management
        i.
    c. Participate
        i. Any partner can search for an existing ecosystem and ask for participation. This participation might require a membership fee with INTELL COINS or TIEX DAO COINS.
        ii. Any partner can accept terms and conditions outlining their rights and responsibilities.
3. Datascientist / Developers dashboard
    a. SDKs
        i. tiex.py SDK
            1. Register their data
            2. Register their models
            3. Formation of an ecosystem within the DAO
            4. Manage sharing of models

5. Federate their models
6. Access model performance and monitoring services

b. Resources
   i. How to guides
c. Playground
   i. Try a model

4. Data and model registration proposals
5. Model creation request(Proposal or ecosystem need proposal)
   a. Any user can issue a proposal for a model creation for any member(s) to work on it.
      i. How far do we go on-chain? Do we escrow through a contract the funds while Datascientists are working? Ideally yes as it will already solve the royalty and splits logic.
   b. Once done, this models will be fingerprinted , encrypted and registered using Phase 1, **TIEX Exchange**
6. Model fund requests. Any model owner can mark his models available for investment.
   a. This should reuse the model creation request adding a minimum amount needed and a time limit.
      i. All payers should “approve” model as done to unlock the funds.
      ii. In case of reaching the time limit funds will be released automatically, deducting tx fees from the funds.
      iii. At any time the model owner can raise a dispute for the following topics
         1. Unrealistic requests
         2. Model redefinition
         3. others.
7. Service fund requests. At any time a service can be requested to any partner (in particular to TIEX Company) through foundation or directly. ie. A new ecosystem is created, foundation will need to create a new contract and deploy it. That can be done by TIEX company. It will be transparent for the ecosystem partners.
   a. This should reuse the model creation request establishing fixed amounts and minimum amounts needed and a time limit.
      i. At least 51% of payers should “approve” the service as done to unlock the funds.
      ii. In case of reaching the time limit funds will be released automatically, deducting tx fees from the funds.
8. Model completeness validation (Establish a validation council)
   a. Any validation can issue an update/improvement/fixing request. Such update will be considered as an evolution that will replace the old model (in case it’s deployed already).
   b. Validation council will check the model against the description provided by the creators.
   c. Such council will be chosen by the DAO, at the begining it can be done by TIEX as the first council.
9. DAO Platform Dashboard
   a. “My ecosystems”
   b. “My Models”

    c. “My profile”
    d. “Search a model”
    e. “Searh an ecosystem”
    f. Give access to visualization tools for exploring, analyzing, and interacting with data, models, and insights.
10. Model access or subscription
    a. Any partner can utilize one or more models by combining them. For such action the business user accessing wallet should own one or more shares/editions of each involved model.
    b. Any partner will use credits (INTELL COIN) for using the platform and access models. This will lower the barrier for actions such as signing, buy , selling and listing through the various features of the platforms.
    c. Each payment will be splitted on : tx fee, platform fee, feature specific fee (which can also be splitted among the collaborators and investors through the reward and split system).
        i. In case of user = investor there will be a percentage discount over the fee.
    d. Each model can be shared by its creators and investors for other collaborators/ecosystems for improvement, ensemblement. Direct usage is possible but has no relevance on pricing logic. This is extended to datasets and visualizations. An ecosystem X member can see all public models of any X ecosystem members by default.

11. Federated model creation
    a. Each ecosystem partner can provide his federated intelligence data through
        i. Service for differential privacy and homomorphic encryption data protection
        ii. Model creation
    b. Each model can be shared to a specific group of members or be publicly available.
    c. Each federated model will be available through a service (ideally generic access) ie. API Endpoint [Action=GET] /federation/?Ecosystem=Pizza&Forecast=March/2023 (with Access token and Signature associated with owner wallet
12. Federated and no federated dataset creation
    a. Any partner can connect and import external information through APIs and data file processing. This will be solved by ETLs?
    b. Storage of such data will be following an hybrid model TBD. Hybrid with Cloud storage + IPFS, data should be encrypted with every owner private key. Sharing of data will be done by delegating access to other keys. Will need to find the right encryption algorithm to allow this.
13. Federated analysis across multiple members or an ecosystem
    a. Add federated information to an existing ecosystem
    b. Subscribe to a model output from federated analysis
14. Model monitoring and performance reporting
    a. Validation through performance metrics and performance. This a recurring task to establish the validity of the model. Metadata, such as data lineage,

provenance, and schema information, to ensure data quality and trust needs to be present and maintained.

15. Access control and payments
    a. Each partner will need to pay at different stages for accessing the several services available
        i. Entry fee. Each DAO member will pay a fee with FIAT through creditcard or other method. That will be turned into crypto and assigned to his wallet in INTELL COINS tokens.
        ii. Recurrent fee. Each DAO member will buy credits with FIAT for been able to use the platform and access models. This will lower the barrier for actions such as signing, buy , selling and listing through the various features of the platforms.
        iii. Each fee payment will be splitted on : tx fee, platform fee, feature specific fee (which can also be splitted among the collaborators and investors, this last one will receive a reward for his staking % on the model).

            1. MODEL USAGE TOTAL PRICE = (TX FEE + PLATFORM FEE + NFT MODEL USAGE PRICE)

               NFT MODEL USAGE PRICE = CREATOR BASE PRICE + INVESTOR(1|0) * (INVESTOR_TOTAL_NFTS/TOTAL_NFTS) * CREATOR BASE PRICE (Will depend for each investor)

    b. Each partner will be authenticated through his wallet address for using customer facing platforms and by signing transactions while using and executing models. This will happen in many cases while paying a specific feature or service execution.
16. Basic (and not so basic) set of models ready to be used, improved and ensembled with other models through services.
    a. KnowledgeGRAPH
    b. BNOMAI
    c. Canonical data structures tied to each impact
    d. Product, store, or customer identity management capabilities
       This models will be fingerprinted , encrypted and registered using Phase 1, **TIEX Exchange**
    e. Descriptive, predictive, and prescriptive analytics

DAO functionality includes (such functionality can be implemented manually, through software or a combination):

1. API gateway and service orchestration: The DAO provides micro-services that route requests to appropriate micro-services, ensuring efficient communication between components and supporting API-driven integrations.

2. Authentication and identity management services: The DAO manages user authentication and identity, integrating with existing authentication providers, such as Single Sign-On (SSO) or OAuth2. The DAO defines the protocol to give users control over the data sources that are used to improve the model.
3. Collaboration and sharing services: The DAO enables users to share datasets, models, visualizations, and insights with other members of the ecosystem, fostering collaboration and innovation.
4. Communication Protocols: The DAO manages the communication protocol that allows devices to communicate with each other securely and efficiently.
5. Data access and security services: The DAO defines fine-grained access control and data encryption to ensure data privacy and compliance with relevant regulations. ThA secure, distributed storage infrastructure ensuring that data is stored privately, and access is granted only to authorized parties
6. Data analytics and machine learning services: The DAO provides a suite of analytics and machine learning capabilities, such as descriptive, predictive, and prescriptive analytics, as well as model training, deployment, and monitoring.
7. Data catalog and discovery services: The DAO enables users to search, discover, and access relevant datasets, models, and other resources within the shared ecosystem.
8. Data ingestion and integration services: The DAO facilitate the secure and efficient ingestion of data from various sources, such as databases, APIs, and file systems, into the shared ecosystem.
9. Data Management: The DAO defines the ‘best practice’ architecture for data management that allows devices to share relevant data with each other while maintaining user privacy. This can include techniques such as differential privacy, data anonymization, and data sharding. The DAO defines storage and management of data in a scalable and distributed manner, supporting various data formats and structures, such as relational, non-relational, time-series, and graph databases.
10. Data preprocessing: The DAO provides tools and services that preprocess and clean data, ensuring compatibility across different data sources and formats.
11. Data transformation and processing services: The DAO defines or performs data cleaning, enrichment, transformation, and aggregation tasks to prepare data for analysis and machine learning.
12. Documentation and Support: The DAO provides documentation and support for the protocol and SDK. This should include tutorials, sample code, and a community forum.
13. Dynamic Model Aggregation: The DAO defines the protocol to allow for dynamic aggregation of models from different devices.
14. Evaluation and Monitoring: The DAO provides the system that evaluates and monitors the performance of the federated learning system. This can include metrics such as model accuracy, convergence time, and system throughput including system performance, log events, and trigger alerts in case of anomalies, ensuring the health and stability of the shared ecosystem.
15. GANs for data sparsity: The DAO defines Generative Adversarial Networks (GANs) can be used to address data sparsity issues by generating synthetic data that mimics the properties of real data, helping to improve model performance.

16. Governance and monitoring: The DAO defines rules, policies, and monitoring mechanisms to ensure the proper functioning of the federated ecosystem and maintain trust among its members.
17. Metadata management services: The DAO maintains and manages metadata, such as data lineage, provenance, and schema information, to ensure data quality and trust.
18. Model sharing and deployment: The DAO provides marketplace functions for sharing and deploying AI models, allowing members to access and use the federated intelligence.
19. Model Specification: The DAO defines the algorithms for federated learning that can work with the specified communication protocol and data management system. This may include techniques such as federated averaging, federated learning with secure aggregation, and federated learning with differential privacy.The DAO defines the federation analytic model architecture, including the number of layers, the activation functions, the optimizer, and other hyperparameters are defined.
20. Model training and evaluation: The DAO provides services for training, validating, and evaluating AI models within the federated ecosystem.
21. Model Versioning: The DAO defines the protocols for model versioning into the protocol so that new versions of the model can be created as users append new data sources.
22. Opt-in mechanism: The DAO defines services for members to opt into the federation, with clear terms and conditions outlining their rights and responsibilities.
23. Privacy-preserving techniques: The DAO provides privacy-preserving techniques such as federated learning, differential privacy, and secure multi-party computation to maintain data privacy and security while sharing intelligence.
24. SDK: The DAO provides the SDK that enables developers to easily integrate federated learning into their applications. The SDK should include client-side and server-side components that can be integrated into existing machine learning frameworks such as TensorFlow or PyTorch.
25. Transfer learning services: The DAO provides tools and services that enable cross-ecosystem intelligence by leveraging pre-trained models and applying them to new, related tasks through transfer learning.
26. User Data Privacy: The DAO provides the services for ensuring that user data privacy is maintained as new data sources are added.
27. User interface and visualization services: The DAO provides user interfaces and visualization tools for exploring, analyzing, and interacting with data, models, and insights.

Federation DAO Mechanism

This is a short explanation on how the federation will take place, and how smartcontracts and blockchain will ensure such Federation providing warranties for all the parties.

Existing architecture

- ➔ DAO registry
- ➔ Vault
  - ◆ Stores actual key access to various platforms delivering temporary access keys
- ➔ Datasources validator Oracle
- ➔ Data provider registration
  - ◆ data provider registration process library and sc contract that will work with the VAULT
- ➔ Generic integration engine
  - ◆ Integration engine composed of a contract template and a generic job that basically asks the VAULT for a specific temporary access key to a specific registered data provider, that will store a result set of information on a specific output format and location.Implementation might change based on encryption, and data result in storage needs.
  - ◆ Encryption method selector
    - Homomorfic, watermark, diferential privacy
- ➔ Data provenance stamper
- ➔ Performance monitor
- ➔ Models
  - ◆ DS_1_Model_Operator_Predictor_Gen it's a generic sales model that receives a sales normalized information and returns Predictions for a specific period of time for product and can accept any product as a training parameter
  - ◆ Basic_Model_Operator_Predictor it's a basic sales model that can be inherited by other models for specialization
- ➔ Oracles
  - ◆ OR_Sales_Model_Gen_output it's a generic sales model Oracle connected to the DS_1_Model_Operator_Predictor_Gen Model

For this example we asume DAO and the sub-DAO as one, in the future some elements migh be available cross sub-DAOs throuhg the main DAO.

Scenario

The Food Ecosystem sub-DAO composed by Tyson, Smithfield Ham and US Foods is looking to create intelligence about product sales, specifically ham and chicken, that will help both on knowing where there is the greatest opportunity to sell more chicken or ham, and

how Tyson, Smithfield Ham and US Foods can coordinate sales and marketing activities over the next 6 months.

Given that Tyson, Smithfield Ham and US Foods have areas of specialization and competency in sales and marketing coordination of efforts will increase efficiency and maximize impact.

For such work all need to create intelligence from their data in order to:

1. Determine historical sales targets
2. Determine historical sales and marketing effectiveness
3. Calculate restaurant operator level demand for chicken in the future
4. Determine the best targets for Tyson and US Foods based on sales and marketing competencies
5. Optimize sales and marketing for individual operators including:
    a. Specific product
    b. Personalized messaging
    c. Best channel
    d. Likelihood or propensity to purchase
    e. Best sales force
    f. Best marketing force

Through the DAO they are able to:

1. Secure their data
2. Mask the actual data by using algorithms to hide sensitive data
3. Produce a global model of operators based on US Foods and Tysons' experience
4. Produce a Tyson and US Foods specific plan consisting of sorted opportunities, actions, and projected impact

Users involved

Data Scientists who provide products or services as part of DAO

- ☐ Member_DS_1 is a datascientist part of the DAO.
- ☐ Member_DS_2 is a datascientist part of the DAO.

Data or 3rd party product/service providers as part of DAO

- ☐ Member_DP_1 is a data provider part of the DAO

Consumers of DAO products

- ☐ Member_US_DAO_DS_1 is data or analytics provider (could be data scientist or not) part of the DAO working for US FOODS.

- ☐ Member_TY_DAO_DS_2 is data or analytics provider (could be data scientist or not) part of the DAO working for TYSON.
- ☐ Member_SM_DAO_DS_3 is a data or analytics provider (could be data scientist or not) part of the DAO working for Smithfield.
- ☐ Member_TIEX_1 is a member of the general DAO working for TIEX.
  Note: In the future this member can be anyone.

External party not a DAO member

- ☐ NDAO_Member_DS_1 is a datascientist not part of the DAO.
- ☐ NDAO_Member_DS_2 is a datascientist not part of the DAO.

Use Cases

1. Determine which restaurant operators may not buy chicken again
2. Determine which restaurant operator may buy more chicken
3. Determine which restaurant operator should buy chicken but doesn't
4. Determine which sales person should call on a specific restaurant operator about ham
5. Determine which marketing message about chicken should be delivered to a specific restaurant operator
6. Determine which marketing content about chicken should be delivered to a specific restaurant operator
7. Determine which marketing promotion for chicken should be delivered to a specific restaurant operator
8. Determine which restaurant operators may not buy ham again
9. Determine which restaurant operator may buy more ham
10. Determine which restaurant operator should buy ham but doesn't
11. Determine which sales person should call on a specific restaurant operator about ham
12. Determine which marketing message about ham should be delivered to a specific restaurant operator
13. Determine which marketing content about ham should be delivered to a specific restaurant operator
14. Determine which marketing promotion for ham should be delivered to a specific restaurant operator
15. Determine a sales person's effectiveness for selling chicken across all restaurant operators who should buy more chicken
16. Determine which sales person would be bet to sell chicken to a restaurant likely to buy less chicken
17. Determine a marketing person's effectiveness for selling chicken across all restaurant operators who should buy more chicken
18. Determine which marketing person would be best to sell chicken to a restaurant likely to buy less chicken
19. Determine a sales person's effectiveness for selling ham across all restaurant operators who should buy more ham

20. Determine which sales person would be best to sell ham to a restaurant likely to buy less ham
21. Determine a marketing person's effectiveness for selling ham across all restaurant operators who should buy more ham
22. Determine which marketing person would be best to sell ham to a restaurant likely to buy less ham

Data sources involved

- ☐ TYSON_Salesforces_API : Sales endpoints only available for Member_TY_DAO_DS_1
- ☐ US_FOODS_Salesforces_API : Sales endpoints only available for Member_US_DAO_DS_2
- ☐ SMITHFIELD_Salesforces_API : Sales endpoints only available for Member_SM_DAO_DS_3
- ☐ TYSON_Marketing_API : Sales endpoints only available for Member_TY_DAO_DS_1
- ☐ US_FOODS_Marketing_API : Sales endpoints only available for Member_US_DAO_DS_2
- ☐ SMITHFIELD_Marketing_API : Sales endpoints only available for Member_SM_DAO_DS_3
- ☐ DAO_Operator_API : DAO endpoints only available for DAO members
- ☐ DAO_LLM_API: DAO endpoint only available for DAO members
- ☐ DAO_API: Restaurant operator identity endpoint available for all DAO members, consisting of:
    - ➢ Google Eats Attributes
    - ➢ Yelp Eats Attributes
    - ➢ cross-DAO product Attributes
    - ➢ Twitter reviews

There is no integration done for any of the APIs in question.

Data involved

- ☐ Historical sales data
- ☐ Historical sales activities data
- ☐ Historical marketing activities data
- ☐ Operator level forecasts
- ☐ Operator level attributes and identity
- ☐ 3rd party data on location characteristics and dynamics
- ☐ Other non-tyson sales activities at the operator level
- ☐ Other non-US Foods sales activities at the operator level

- ☐ Other non-tyson marketing activities at the operator level
- ☐ Other non-US Foods marketing activities at the operator level

Services involved

- ☐ Data transformation
- ☐ Data security
- ☐ Data encryption
- ☐ Algorithm execution

Models involved

- ☐ DS_1_Model_Operator_Predictor_Ham:
  A ***product specific*** sales prediction model that receives a sales normalized information and returns Predictions for a specific period of time for Ham
- ☐ DS_1_Model_Operator_Predictor_Gen:
  A ***generalized sales*** prediction model that receives a sales normalized information and returns Predictions for a specific period of time for product and can accept Ham as a training parameter
- ☐ DS_2_Model_Operator_Predictor_Chicken:
  A ***basic sales*** prediction model that receives a sales normalized information and returns Predictions for a specific period of time for Chicken
- ☐ DS_1_Model_Operator_Predicto_Eggs:
  A ***basic sales*** prediction model that receives a sales normalized information and returns Predictions for a specific period of time for eggs
- ☐ DS_2_Model_Operator_Predictor_Cheese:
  A ***basic sales*** prediction model that receives a sales normalized information and returns Predictions for a specific period of time for cream cheese
- ☐ DS_1_Model_Operator_Health:
  A ***basic prediction*** model that receives a 3rd party data and predicts operator financial wellbeing and health

These models are DAO generated models and federated across all analytics.

- ☐ TY_Model_Sales_Predict_1:
  Uses basic sales prediction model that receives a sales normalized information and returns Predictions for a specific period of time.
- ☐ US_Model_Sales_Predict_1:
  Uses basic sales prediction model that receives a sales normalized information and returns Predictions for a specific period of time.
- ☐ SMITH_Model_Sales_Predict_1:
  Uses basic sales prediction model that receives a sales normalized information and returns Predictions for a specific period of time.

- ☐ SOM_Model_BNOMAIv2_2:

Is the information normalizer, taking raw information and normalization parameters as an input, responding a normalized set of data.
This model already exist and it's ready to be used by the DAO. It was made by a member not part of the DAO based on the default TIEX_Model_BNOMAI_1 made by TIEX.

- ☐ sDAO_Model_Product_Predict_1:
  Is a model that receives Sales normalized data from TIEX_Model_Sales_Predict_1 and returns how much chicken needs to be be produced over time to fullfill sales of the model.
  This model does not exist.

Local Model execution job

- ☐ sDAO_Model_Product_Predict_Chicken_1_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of chickens production needs for the period of time asked.
- ☐ sDAO_Model_Product_Predict_Chicken_1_Job_2 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.
- ☐ sDAO_Model_Product_Sales_Chicken_1_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.
- ☐ sDAO_Model_Product_Marketing_Chicken_1_Job_2 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.
- ☐ sDAO_Model_Product_Predict_Ham_1_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of chickens production needs for the period of time asked.
- ☐ sDAO_Model_Product_Predict_Ham_1_Job_2 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.
- ☐ sDAO_Model_Product_Sales_Ham_1_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.
- ☐ sDAO_Model_Product_Marketing_Ham_1_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and

deliver the information expected. It will run all parts and deliver the report of ham production needs for the period of time asked.

This execution model job does not exist. It will use all existing inputs of data and models.

Utility Services

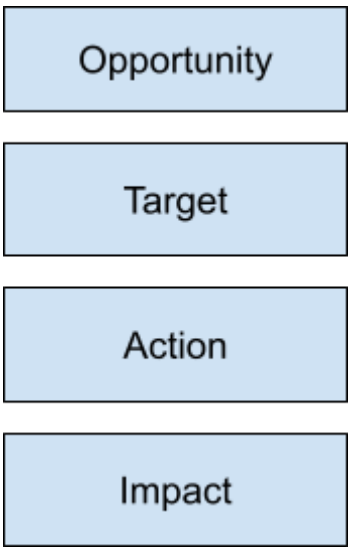
- ☐ sDAO_Model_Federation_Job_1
- ☐ sDAO_Performance_Monitoring_Job_1
- ☐ sDAO_Alert_Job_1

Output

- ☐ Best operator
- ☐ Best message
- ☐ Best promotion
- ☐ Best product

NOTE

There is a hierarchy of model federation:



Security management:

There are three levels of security for managing federation:

1. Do I want to allow an algorithm to be executed against my data?
2. Do I want to allow my algorithm to be combined with other algorithms that I do not control?

3. Do I want to allow my algorithm output to be shared?
4. Do I want to allow my algorithm output to be federated?
5. Do I want the federation to be shared with someone?
6. How can I monetize this output?
7. How can I make this output sticky so as to preserve customer relationships?
8. Do I want to create a subset of individuals that can see my output?
9. Do I want to create a subset of individuals that can aggregate my output with theirs to create net new?

Federated Model execution job

- ☐ sDAO_Model_Operator__Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected. It will run all parts and deliver the report of chickens production needs for the period of time asked and federates:
  - ➔ sDAO_Model_Product_Predict_Chicken_1_Job_1
  - ➔ sDAO_Model_Product_Predict_Chicken_1_Job_2
  - ➔ sDAO_Model_Product_Sales_Chicken_1_Job_1
  - ➔ sDAO_Model_Product_Marketing_Chicken_1_Job_2 :
  - ➔ sDAO_Model_Product_Predict_Ham_1_Job_1
  - ➔ sDAO_Model_Product_Predict_Ham_1_Job_2
  - ➔ sDAO_Model_Product_Sales_Ham_1_Job_1
  - ➔ sDAO_Model_Product_Marketing_Ham_1_Job_1

  This execution model job does not exist. It will use all existing inputs of data and models.

- ☐ sDAO_Model_Customer_Segmentation_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected for US Foods, Tyson and Smithfield.
- ☐ DAO_Model_Market_Optimization_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected or US Foods, Tyson and Smithfield.
- ☐ DAO_Model_Market_Impact_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected or US Foods, Tyson and Smithfield.
- ☐ DAO_Model_Sales_Impact_Job_1 : Once all models and data inputs integrations are ready, a model execution job needs to be created to retrieve and deliver the information expected or US Foods, Tyson and Smithfield.

Oracles involved

At first sight many Oracles are needed to incorporate external information, one for each API Integration.

- ☐ OR_TYSON_Salesforces
- ☐ OR_US_FOODS_Salesforces

- [ ] OR_SMITHFIELD_Salesforces
- [ ] OR_US_FOODS_Marketing
- [ ] OR_TYSON_Marketing
- [ ] OR_Smithfield_Marketing
- [ ] OR_TYSON_Sales_Model_output
- [ ] OR_TYSON_Marketing_Model_output
- [ ] OR_US_FOODS_Sales_Model_output
- [ ] OR_US_FOODS_Marketing_Model_output
- [ ] OR_Smithfield_Sales_Model_output
- [ ] OR_Smithfield_Marketing_Model_output
- [ ] OR_Operator_Purchase_Model_output

None of these Oracles exists. They are made by using Chainlink over Avalanche.
https://docs.avax.network/community/tutorials-contest/2021/red-dev-avalanche-chainlink-tutorial

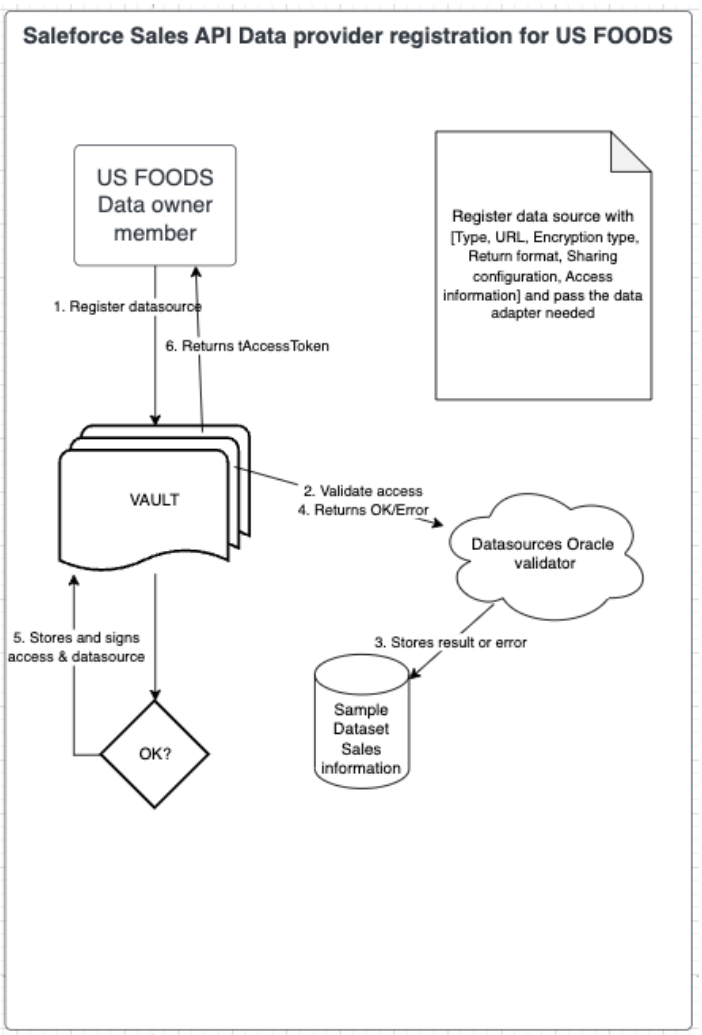
- [ ] OR_Sales_Model_Gen_output

Creation Process

As there are some elements needed the entire process cannot be done righ away. Basically a couple of tasks are needed first.

Salesforces Integration

- Saleforces API Data provider registration
  - Each data provider needs to be registered and federated by the data owner using a proper encryption algorithm. Such registration also considers the need of Data Adapters. In this particular case each of the data owner will register the APIs and their Access tokens by signing with their private keys.
    1. Source: API URL
    2. Parameters: Organization, Domain, Date start, Date end
    3. Response format: JSON
  - This will save the Data provider tied to the the requester address and private key on the VAULT Contract. This key will also be used for encrypting and decripting results returned by the salesfoces integration engine.
  - Only the owner can see plain data from the response obtained through the Datasources validator Oracle



- Integration engine for Saleforces integration
  - Using the Integration Engine template it will be needed to redefine
    1. Encription method (homomorfic, watermark, diferential privacy)
       - Homomorfic to preserve data structure
    2. Encription key (private, public, combined)
       - Member_US_DAO_DS_1 private key for US FOODS
       - Member_TY_DAO_DS_2 private key for TYSON
       - Member_SM_DAO_DS_3 private key for Smithfield
  - The user will be able to stamp the information source or ask stamping to a third party. In this example we cover the auto stamping by providing a proof document that state that the API Key used belongs to real customer info for an epoch. Ideally each time this integration is used this stamp should be renew. Non-existance of this stamp will lower the trust of the model(s) using it.

- Oracles creation (OR_TYSON_Salesforces, OR_US_FOODS_Salesforces, OR_SMITHFIELD_Salesforces, OR_US_FOODS_Marketing OR_TYSON_Marketing, OR_Smithfield_Marketing, OR_TYSON_Sales_Model_output, OR_TYSON_Marketing_Model_output, OR_US_FOODS_Sales_Model_output,

OR_US_FOODS_Marketing_Model_output,
OR_Smithfield_Sales_Model_output,
OR_Smithfield_Marketing_Model_output,
OR_Operator_Purchase_Model_output)

- These Oracles are going to be created to incorporate the information usage from the APIs and be used. In other words, it doesn't bring the information itself but it's the only way to to access the API.
    1. For OR_TYSON_Salesforces it:
        - Ask access to TYSON_Salesforces_API, which will check on the FOOD DAO Contract if requester address has allowance
        - Will retrieve the **temporary access token** from the VAULT
        - Will write access request by requester address.
        - Will deduct funds from the requester address (while signing)
    2. For OR_US_FOODS and the following oracles is the same process following the respective security and API specific accesses.
- Made in coordination between Member_TIEX_1 for building Chailink Smart contracts and members, Member_US_DAO_DS_1 and Member_TY_DAO_DS_2 and Member_SM_DAO_DS_3 respectively depending on the source of information needed.

- Each Saleforces API Integration engine is composed by a Smart Contract and a Job processor
    - Each of these integrations will expect a **temporary access token** that will be used to ask the VAULT for the **access token** to access the Salesforces API by the Job processor
    - Made by the Member_TIEX_1. While he has been granted access by the DAO it will be able for him to try it out. Also any of the following member of the DAO will be able to run and see their sourced models execution:
        1. Member_US_DAO_DS_1
        2. Member_SM_DAO_DS_3
        3. Member_TY_DAO_DS_2

Models needed creation

- DS_1_Model_Operator_Predictor_Ham model will be done by Member_SM_DAO_DS_3 to use when running the job.
    - Use the Model NFT Contract template
    - Use the script creation for models
    - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
      Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.
- DS_1_Model_Operator_Predictor_Gen

        - Already exists as part of the platform
        - Member_SM_DAO_DS_3 will use Ham sales normalized data to input the existing model
        - Uses the existing Oracle.
    - DS_2_Model_Operator_Predictor_Chicken model will be done by Member_TY_DAO_DS_2
        - Uses the Model NFT Contract template inheriting from the Basic_Model_Operator_Predictor model made by TIEX
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
          Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.
    - DS_1_Model_Operator_Predictor_Eggs will be done by Member_US_DAO_DS_1
        - Use the Model NFT Contract template
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
          Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.
    - DS_2_Model_Operator_Predictor_Cheese
        - Use the Model NFT Contract template
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
          Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.

    - DS_1_Model_Operator_Health
        - Use the Model NFT Contract template
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
          Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.

    - TY_Model_Sales_Predict_1
        - Use the Model NFT Contract template
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
          Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.

    - US_Model_Sales_Predict_1
        - Use the Model NFT Contract template
        - Use the script creation for models
        - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.

Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.

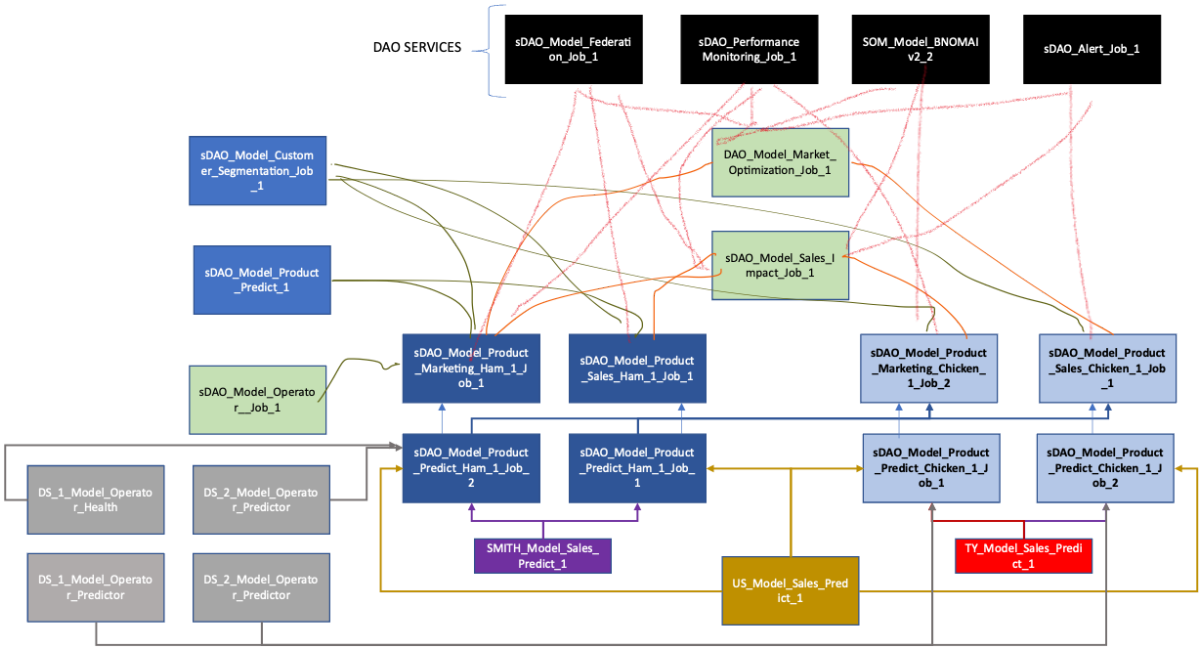
- SMITH_Model_Sales_Predict_1
  - Use the Model NFT Contract template
  - Use the script creation for models
  - Connect and implement the respective Oracle usage. In this case we will refere the OR_Smithfield_Sales_Model_output Oracle.
    Note: The NFT doesn't run or gather information for the Oracle, it states that it uses the Oracle though.

Model execution JOBs

- DS_1_Model_Operator_Predictor_Ham will be done by Member_TIEX_1 it will make use of sDAO_Model_Product_Predict_1, that make use of SOM_Model_BNOMAIv2_2, TIEX_Model_Sales_Predict_1 as well as the OR_TYSON_Salesforces and OR_US_FOODS_Salesforces to access both TYSON_Salesforces_API and US_FOODS_Salesforces_API
- sDAO_Model_Product_Predict_Chicken_1_Job_2
- sDAO_Model_Product_Sales_Chicken_1_Job_1
- sDAO_Model_Product_Marketing_Chicken_1_Job_2
- sDAO_Model_Product_Predict_Ham_1_Job_1
- sDAO_Model_Product_Predict_Ham_1_Job_2
- sDAO_Model_Product_Sales_Ham_1_Job_1
- sDAO_Model_Product_Marketing_Ham_1_Job_1

- Each execution needs to consider performance updates for the monitor performance Job to update.
- Each execution needs to consider alert updates for the alert Job to update.

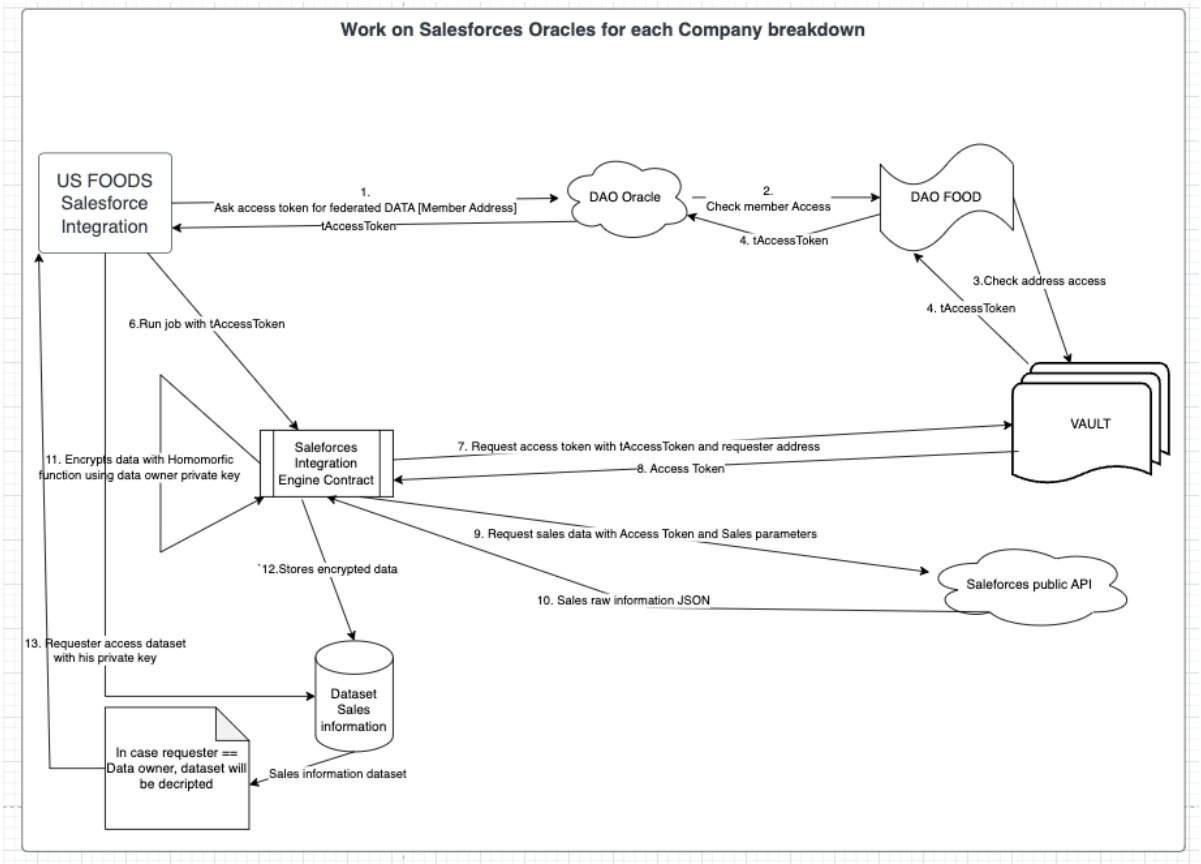
Model Relationships and flows:



Model Execution process

Only allowed members are able to execute this job DAO_Model_Product_Predict_1_Job_1. Such validation is done by the requesting member to the exchange platform service.

1) Member will call the getService passing the Model he wants , in this case “DAO_Model_Product_Predict_1_Job_1” and signing with his wallet address private key paying for TX gas needed for such TX.
2) This will start and run a Job by the TIEX platform that will
    a) Ask the corresponding DAO (FOOD DAO) if such address has access
    b) Run the Job by using the models and elements created above.



Vault and security access method

For accessing any model or data from a DAO , dao members needs to allow access criptographically ensuring the entire process. For such feature we created the VAULT concept. The vault doesn't contain data, only access to data in a decentralized way. For this example we will show how the DATA access to the APIs where established for DAO FOOD intelligence creation process. We already know that both US FOODS, SMIThFIELD and TYSON decided to create such intelligence for themselves but benefiting from each other, so they will:

1) One member of the DAO (data owner) will start a VAULT request, providing the signing access type (API with bearer access token, user& password , other) and the list of addresses (members) that can access to the data, will also provide the corresponding acccess token (in our case the access token for US FOODS API or TYSON depending the case. This will store the access token on the vault , encrypted in a way that only allowed members wallets address signing will unlock the access token when called through the corresponding oracle using :

    f_unlock(encrypted_access_token, requester_signature, requested_data)-> access_token

    requires(Contract.Caller==f_get_oracle_for_this_data(requested_data))

2) VAULT request can be optionally associated to a time frame
3) VAULT request can be disallow or deleted at any time unless there is a time frame option in place.
4) This VAULT lives on IPFS or similar
5) A recovery system needs to be placed?

Job Board

As many items where not available, it would make sense to establish a job board on the TIEX platform for communication to all community the need for some work to be done. In this example such jobs would be:

- ➔ sDAO_Model_Product_Predict_1_Job_1
- ➔ OR_TYSON_Salesforces
- ➔ OR_US_FOODS_Salesforces
- ➔ Saleforces API Integration for both TYSON and US FOODS

Rewards engine

As defined in upper sections there is a reward schema that needs to be executed, for each sDAO_Model_Product_Predict_1_Job_1 execution it should be a fee for:

- ➔ sDAO_Model_Product_Predict_1_Job_1 maker that will be splitted with
  - ◆ sDAO_Model_Product_Predict_1 maker that will be splitted with
    - SOM_Model_BNOMAIv2_2 maker that will be splitted with
      - TIEX_Model_BNOMAI_1 maker (TIEX) that will be spllited with
        - ◆ TIEX Platform
        - ◆ Model NFT investors
    - sDAO_Model_Product_Predict_1 maker that will be spllited with
      - TIEX Platform
      - Model NFT Investors
    - OR_US_FOODS_Salesforces maker that will be spllited with
      - TIEX Platform
    - OR_US_FOODS_Salesforces maker that will be spllited with
      - TIEX Platform
  - ◆ TIEX Platform

ERICSNOTE:

- Data provenance and trust

- A new type of protocol or approach is required that tracks on blockchain provenance and trust
- Other technologies help including:
    1. Decentralized Identifiers (DIDs)
    2. Verifiable Credentials (VCs):
    3. Data integrity and traceability using blockchain to provide a transparent, tamper-resistant record of data provenance.
- For example

In this scenario, let's consider a supply chain management system for tracking the origin, processing, and distribution of coffee beans. We have a data originator, DA_01, who is the coffee bean farmer and several data transformers (DT_01, DT_02, etc.) representing various stages of the coffee bean supply chain.

1. DA_01 (coffee bean farmer) records the harvest data, including the date, location, and quantity of beans harvested, and uploads it to the supply chain management system.
2. DT_01 (bean processing facility) receives the beans and records data on the processing method, storage conditions, and processing date. They then transform the raw harvest data from DA_01 by appending the new processing information.
3. DT_02 (transportation company) transports the processed beans to the roasting facility. They record data on transportation conditions, such as temperature and humidity, and transport duration. They further transform the data by appending transportation details to the existing dataset.
4. DT_03 (roasting facility) roasts the beans and records data on the roasting process, such as roasting temperature, duration, and batch size. They then transform the dataset again by appending the roasting information.
5. DT_04 (packaging facility) packages the roasted beans and records data on packaging type, weight, and labeling. They add this information to the existing dataset, transforming it further.
6. DT_05 (retail distributor) receives the packaged beans and records data on storage conditions, stock levels, and sales. They transform the dataset once more by appending their data.

At each stage, the dataset has been transformed, and data from various sources has been added to the original harvest data provided by DA_01. This creates a complex web of data provenance that can be difficult to track and verify. For example, it can be challenging to ensure that the transportation conditions data provided by DT_02 is

General notes for implementation:

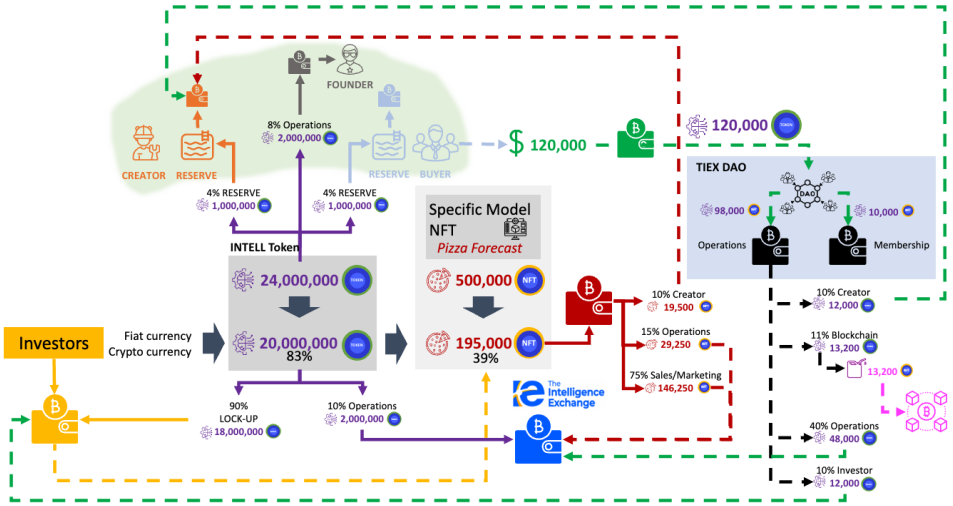
1. A Pay for TX model and or generalized accounts should be used to avoid

General notes about the DAO and tokens

1) TIEX DAO Token (ERC20 Utility Token)
    a) Whitelisting
        i) Founders

ii) Businesses
iii) Third parties
iv) Investors in the DAO
v) General audience

b) Tokens Flow and logic



Complete diagram:
https://shortesttrack-my.sharepoint.com/:p:/g/personal/erich_intelligenceexchange_ai/EZeA8IpYWuhHhc85SsCzvLwBurbnQi9QhcX959P5cuiOxg?e=B3ZChT&wdLOR=cA509AA75-1515-9E44-B987-BB594743B47D

c) Airdrop
i) If we open this to general community members it will help with promotion efforts, small tasks, and the virality of our messages.

d) Proposals management
i) Proposals are offers between two or more companies to collaborate in the creation, federation and use of artificial intelligence.
ii) A DAO member can join a specific ecosystem with confirmation by the ecosystem creator and all other members of the ecosystem as confirmed by ownership of a token
iii) A member of the ecosystem can propose specific problems to solve, certain data or feature contributions, or organize outcomes.
iv) A proposal can contain a specific change to the rules, rewards shares definition for a newly trained model, an assembly of many, or improvements of existing models.
v) Any DAO member can issue a vote through the platform, votes will be always related to proposals. Voters will use the TIEX TOKENS.

e) Execution of proposals
i) Once a proposal reaches the minimum amount of votes it will be executed by the executive team within the terms established. Proofs of executions might be delivered to the DAO. ie. A new NFT is defined, and collaborators' wallets and their shares are defined. The

team will create a new smart contract and list the new NFT on the exchange platform to be bought and consumed by the investor(s). Proof of this action to be marked as completed is the contract source code and deployed address to be verified.

f) Exceptions

2) Exchange contract
   a) All NFTs transfers will be done through the exchange platform, which means:
      i) Buying an NFT
      ii) Selling an NFT
         (1) A DataScientist /Collaborator can sell his Models to other DataScientist or investors or Consumers, this will update the revenue path of further usages of such Models.
      iii) Burning an NFT
         (1) The owner can burn his Models anytime in a unilateral way.
         (2) The platform can burn any NFT while updating it with one or more other NFTs as an improvement. Will need the owner(s) signatures to be completed.
      iv) Atomic swaps
      v) Transfers executions

3) Whitepaper
   a) General description (Knowledge Grid has the right topics although descriptions are changing right now). As a note in the team section, there is a lack of blockchain people.
4) Website
   a) Presentation
   b) Roadmap
   c) Newsletter / Subscription
5) Auditing
   Once all the contracts are done we need to hire a well-known auditing company to ensure and do a quality stamp over the work done.
6) Community management
   a) Discord creation
      i) It will be the community place to go for collaboration, real-time discussions, and private access for token holders
   b) Forum (Discourse based)
      i) This is the place where the DAO is ruled. See https://forum.apecoin.com/ or https://port.oceanprotocol.com/
      ii) New proposals
      iii) Voting
      iv) Definitions
   c) https://snapshot.org/ or similar for proposing and voting proposals using your voting power (tokens).

Glossary

7) TIEX Exchange is the company that is building this platform and ecosystem. Owns many of the trained models mentioned in this document.
8) TIEX DAO is the DAO created and defined in this document.
9) TIEX NFT is one of the many NFTs that will represent ecosystems and trained models mentioned in this document.
10) INTELL COIN is the utility token used for working inside the TIEX Marketplace and DAO Platform.

DAO Factory Scenarios

Within the objective of creating multiple intelligence ecosystems TIEX DAO may find the need for some governance independence. This generates many possible scenarios that raise questions regarding management, legal structure, minimum requirements of the original DAO and what should be allowed and how to the sub-DAO.

First, we establish a DAO Factory concept where all members are part according to what was described before but adding a specific improvement for the ecosystems.

Initially two or more members of the DAO may create an intelligence ecosystem through the following process:

1) A new ecosystem proposal is issued according to the governance rules of the DAO describing the name of the ecosystem, its purpose and objective and where all members will need to be listed and signed the proposal. There will be a minimum amount of votes needed for this proposal to pass (TBD). All new ecosystems starts with an Ecosystem NFT that lives and works inside the DAO.
2) If the proposal gets enoughs votes it will be approved and planed for execution:
    a) Creation and deploy of the Ecosystem NFT by the TIEX DAO executive team
    b) Transfer of one NFT to each registered and proposal signer leaving enough free NFTs for future members

When an ecosystem grows it might find the need of having its own pace and some independence on specific topics from the TIEX DAO focusing on what the ecosystems needs.. ie. An entire supply chain for many pizza businesses is grouped within an ecosystem having high interest federating intelligence and doesn't need to apply for a vote among the other ecosystems.

In such case an ecosystem would need to be upgraded as sub-DAO following the same reasoning of Object Oriented inheritance and Interfaces. Some features of the DAO Factory

can be implemented or redefined and some of them need to be defined or overwritten. A sub-DAO can also take advantage of many of the existing features from the DAO Factory.

The DAO Factory covers, define and provide the following services (check use cases):

- Members identity verification
- Dispute mechanism
- Foundation to run off-chain tasks like payments, contracts, etc
- Service layer and code base
- Service provider access
- Governance framework
- Data and intelligence Federation
- Data provenance and veracity verification (Ownership, legal issues, trusted entity)
- Models
- Members and DAO management
- Ecosystem to sub-DAO upgrade/downgrade management
- Language models
- Identity graph

When an ecosystem gets upgraded it will need to create or redefine:

- Foundation to run off-chain tasks like payments, contracts, etc
- Governance framework
- Members and DAO management
- Liquidation process to downgrade

It also can redefine or adapt:

- Data and intelligence Federation
- Models
- Service layer

It won't be able to:

- Ecosystem to sub-DAO upgrade management

Creating a sub-DAO means to make a new instance of the DAO Contract from the DAO Factory Contract inputting to the constructor method the ecosystem NFT. This transaction will need foundation signatures as well as the NFT Ecosystem owners signatures. While creating the new DAO Contract it will burn the NFTs assigning the corresponding newly minted FTs to each owner member of the ecosystem.

Liquidating a sub-DAO means the exact opposite process, burning FTs and minting NFTs.

Along the contracts we will create sub-topics within the forum, proposal and voting platforms for the new sub-DAO. In any case each sub-DAO can override this by choosing other platforms. It can also change the way governance work without breaking the rules of engagement.

When TIEX DAO raise a vote all members with or without ecosystem are allowed to vote. ALL ERC20 tokens are valid.

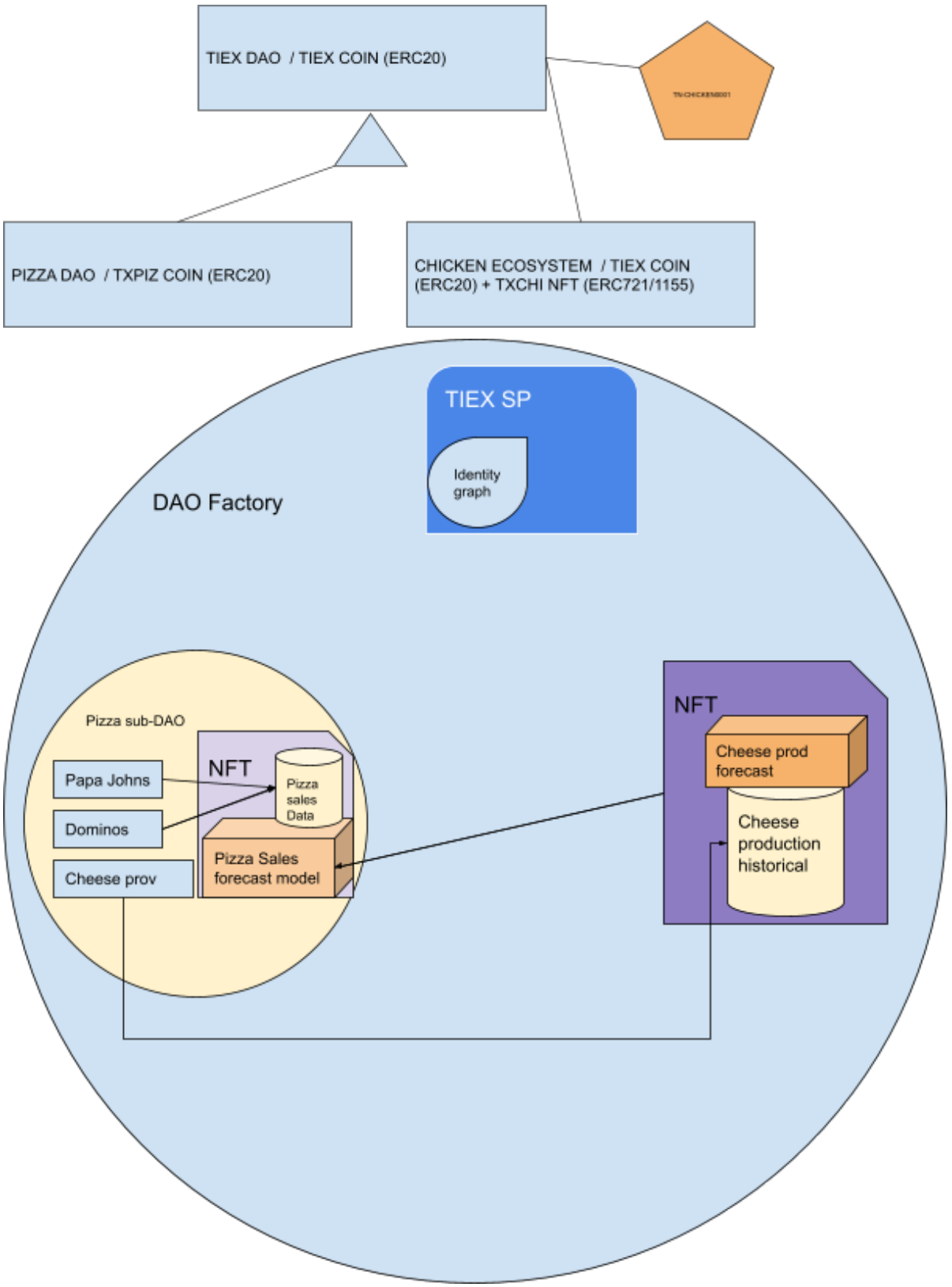

TIEX DAO / TIEX COIN (ERC20)
PIZZA DAO / TXPIZ COIN (ERC20)
CHICKEN ECOSYSTEM / TIEX COIN (ERC20) + TXCHI NFT (ERC721/1155)
TIEX SP
Identity graph
DAO Factory
Pizza sub-DAO
Papa Johns
Dominos
Cheese prov
NFT
Pizza sales Data
Pizza Sales forecast model
NFT
Cheese prod forecast
Cheese production historical

Illustration of DAO process

https://shortesttrack-my.sharepoint.com/:p:/g/personal/erich_intelligenceexchange_ai/EdDpDEiozclKluzL4gvsoJUBvXcz_AEntuesyoB_2aNxYg?e=mJZJBX

Outline of DAO services

https://shortesttrack-my.sharepoint.com/:w:/g/personal/erich_intelligenceexchange_ai/EayD5sZlcl9Ks8lVReUVqKUBeZJZT0qS9Z9hqel9siyx_g?e=2keqRS

Use cases:

https://shortesttrack-my.sharepoint.com/:x:/g/personal/erich_intelligenceexchange_ai/EbHuR59Ovn1Gh_ktNiMBAooBC2Su7YWgZ0pVkMh0-LYTmg?rtime=DIE4xMMj20g