**Thought**

Thought White Paper

## THOUGHT WHITE PAPER CONTENTS

# Hybrid data agents that work as applications.

The global, decentralized, ground-up artificial intelligence network. Unlocking the knowledge potential of the world's information with a Universal Turing Machine.

"

Great things are done by a series of
small things brought together.

— Vincent van Gogh

Nature uses only the longest threads to weave her
patterns, so that each small piece of her fabric reveals
the organization of the entire tapestry.

— Richard P. Feynman

The things that really change the world,
according to Chaos theory, are the tiny things.

— Neil Gaiman

**Thought**

# **1** **Introduction**

**THOUGHT IS A COMPLETELY NEW PARADIGM IN COMPUTING.**

This platform combines data analytics and artificial intelligence to change the way the world creates, processes, interprets, and disposes of the near-limitless amounts of information being created. This white paper will discuss how the system works from bottom up, beginning with the hybrid data and application known as a Nuance and how it interacts with the Thought Fabric.

The paper includes discussion on the sentience of the data itself; its emergent qualities, its lifecycle, and token economy. The paper also discusses several examples of how Thought can be used in various scenarios. First, we discuss the landscape of how Thought originated and the problems it solves for consumers, industry, and the world.

**THE EMERGENT INTERSECTION OF DATA ANALYTICS AND ARTIFICIAL INTELLIGENCE**

For millennia humans have been creating and analyzing data. Whether it was the patterns in the stars, the categorization of different types of animals, or weather patterns throughout the year, recognizing and making use of patterns in data was both a pastime and often key to survival.

Increasingly, the modern world has shifted to creating and interpreting data through machines instead of humans; and every year there is more data to sift through. It will only increase with time.

In our current state, the need for strong and effective data analytics is growing and pattern recognition in this data is even increasingly more crucial today as the world's computers and sensors create unimaginable amounts of data. The challenge today as it was then is to create efficient and effective layers of interpretation for the data. The modern world is no different; in our current state we simply have more data to sift through.

This explosion in data, combined with advances in artificial intelligence, has dramatically increased the ability of computers to process this explosion of global data, but even so, in 2025, according to the IDC, less than 1/5th of all data will be "analyzed."

Also, according to the International Data Corporation, by the year 2025:

- The global datasphere will grow to 163 zettabytes (that is a trillion gigabytes). That's ten times the 16.1 ZB of data generated in 2016.

- Nearly 20% of the data in the global datasphere will be critical to our daily lives and nearly 10% of that will be hypercritical.

- Two-thirds of global financial firms will integrate cognitive data from third parties to improve the customer experience through targeted product and service offerings and fraud protection. Applications for these cognitive systems touch a large surface of our business and personal lives.

- Almost 90% of all data created in the global datasphere will require some level of security, but less than half will be secured.

Until now we have managed all of this data as though it were a static stream of matter, indistinct and undistinguished. All data is created equal until it is sifted and categorized by insight engines; and that data is growing exponentially. Pattern recognition industries, including "big data analytics and "artificial intelligence," are already worth hundreds of billions in market size but still suffer major challenges.

The reason is simple: data is inherently inanimate, and only becomes valuable within the context of an application. As humans and machines continue to produce more data, these associated applications have scaled into massive, cumbersome systems with entire enterprises fashioned around them. At the same time, the web, social media and innovations such as Internet of Things (IoT) rapidly increasing numbers of devices continue to drive exponential data growth. This adds up to a landscape littered with applications, too much data and insufficient intelligence to handle it. Going forward, the current paradigm will no longer be scalable, extensible, adaptable or smart enough to cope with the massive influx of information being generated.

Even with the progress these industries have made, AI and Data Analytics face significant ongoing challenges and inefficiencies. The major issues currently facing AI technology include training, the black box problem, and the issue of transparency and privacy.

### Training

Though AI is heralded as a solution to the massive interpretation of data, traditionally AI neural networks and models still require training on how to sift and interpret massive data sets. Finding the right data and the time it takes to train algorithms with that data is a large encumbrance to truly responsive and ultimately effective AI.

### Black box problem

According to PwC, a central AI theme in 2018 involves solving for the "black box problem" which is faced by sufficiently complex AI networks. In some cases, the designers of the neural networks admit that the networks are too complex to understand how the outputs were derived.

### Transparency/privacy

Powerful AI technology is currently in the hands of a few. Large monopolistic platforms have access to massive amounts of user data used to train their algorithms and they don't (or can't due to the complexity) publish their algorithms. The common user has no insight into how these algorithms are operating and what is being done with their outputs.

**Thought has re-imagined this paradigm by removing the application layer and implementing the data itself as a hybrid application and self-organizing AI organism leveraging a blockchain framework to provide transactional structure and cybersecurity.**

**Thought**

**THE SOLUTION LIES IN A COMPLETELY NEW APPROACH TO ANALYZING DATA.**

In 2018, even the most innovative deep learning techniques still struggle to sift through the world's information. The newest approaches involve mimicking the human brain and creating neural networks.

Thought has gone beyond these solutions and has re-imagined this paradigm by removing the application layer and implementing the data itself as a hybrid application and self-organizing AI organism leveraging a blockchain framework to provide transactional structure and cybersecurity.

At its core, Thought disintermediates the application and embeds smart logic directly into every bit of data. Now, data is no longer inanimate; rather, it becomes agile, able to act on its own, directly at the source of creation, distribution, or action. Ultimately, the result is a new class of information that exists within an intelligent blockchain-enabled Fabric. Thought extends distributed, artificial intelligence to massive data stores and existing systems. It eliminates traditional applications and their related costs, complexities, and scale issues. Because of its ability to embed intelligence directly into the data layer, Thought has the potential to revolutionize both AI and the process of teaching artificial intelligence.

**In the Thought paradigm there is no difference between the data and the application layer; they are one and the same. In Thought, the data is smart and takes action as soon as it is created.**

A few immediate benefits of this new paradigm include reduced process latency and the ability to eliminate a gigantic layer of cost and complexity from an organization of any type and any industry. No longer is it necessary to collect data, examine it, and instruct an application to take action on that data. Data can take action immediately and continuously; AI training is simplified since the training data and AI application are combined. The AI black box problem is solved by hybrid data that "knows" how it traveled through a neural network.

**In the Thought network and paradigm, AI training is simplified since the training of data and AI application are combined. The AI black box problem is solved by hybrid data that knows how it traveled through a neural network.**

## THE IMPLICATIONS OF THIS TECHNOLOGY ARE IMMENSE

As a foundational layer, Thought has many possibilities. For example, you might imagine a completely autonomous power grid. If the average power grid were running on Thought, it could be would be possible to create a self-healing and fully autonomous power grid. Even if an outside party were able could to bring down a part of the grid, the network would self-heal and come back online on its own.

It would also be possible to create an autonomous, unhackable traffic network acting with swarm intelligence; or a model of how bees instinctively solve the Traveling salesman problem; or better understand the complexities of neural networks.

In a future state, Thought will significantly reduce the cost, complexity, and latency in data processing from enterprises because the network eliminates the need for the traditional application stack. The data and the application merge.

Application stacks are redundant in the Thought paradigm as data travels through networks without need for such processing. This provides a streamlined data flow.



The New Way: No Applications Needed

Nuance

Data

The Old Way
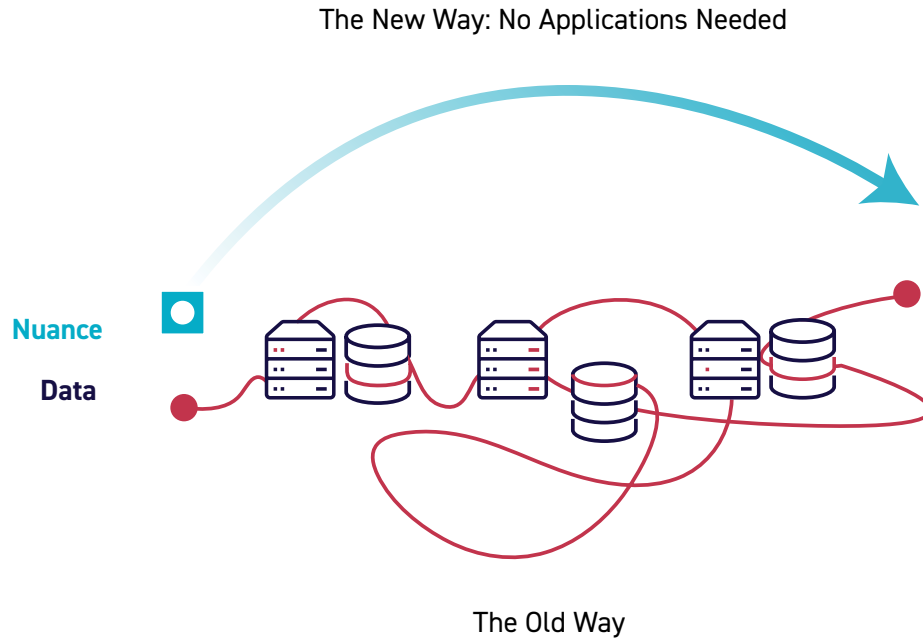
Thought is patented and partnering with a variety of industries including electrical connector and IoT manufacturers, healthcare companies, and engaged in creating a global developer network. It's not just an idea, Thought's premier offering is a licensable standard of development that can be applied to anywhere data is captured, processed and stored. Since it is a standard it allows vast collaboration across otherwise incompatible systems and data types.

For example, Thought is working to embed its AI Fabric directly into logic control boards. This allows application-layer-like intelligence to coexist directly on the board eliminating the need for third party data analysis and shortening the time to action. Now, the logic board that controls a hospital room thermostat can gather information, immediately communicate with the health system's EHR and facilities management software and then raise, lower or alter room humidity. All without an adjacent application, latency, cost or additional licensing.

### Problem 1: Training

Traditionally AI neural networks and models require training with massive data sets. Finding the right data and the time it takes to train with that data is a large encumbrance to responsive AI.

### Solution: Hybrid data/algorithm (The Nuance)

Training is simplified since data and algorithm are one and the same. Data trains itself.

### Problem 2: Black box problem

Sufficiently complex AI networks suffer from the so called Black box problem where determining the validity of the outputs of AI networks can be problematic since the networks become too complex to understand how the outputs were derived.

### Solution: Nuance tracers

Nuances can record their path through a neural network.

### Problem 3: Transparency/privacy

Powerful AI technology is currently in the hands of a few. Large monopolistic platforms have access to massive amounts of user data used to train their algorithms and they don't (or can't due to the complexity) publish their algorithms. The common user has no insight into how these algorithms are operating and what is being done with their outputs.

### Solution: Thought's decentralized AI

algorithms are transparently designed and improved by the community.

# Thought is a completely new paradigm in computing.

**This platform combines data analytics and artificial intelligence to change the way the world creates, processes, interprets, and disposes of the near-limitless amounts of information being created.**

If the average power grid were running on Thought, it could be self-healing and fully autonomous. Even if an outside party were able to bring down a part of the grid, the network would self-heal and come back online on its own.

In a future state, Thought will significantly reduce the cost, complexity, latency in data processing from enterprises because the network eliminates the need for the application stack. The data and the application merge.

**Thought**

# **2** **Glossary: Terms to Know**

**Agent** - A software program that acts on behalf of its user.

**Agent Life Cycle** - The lifecycle of an agent from data creation, "birth", Beginning of Life (BOL) to data deletion, "death", or End of Life (EOL).

**Blockchain** - A distributed and agreed upon database of transactions.

**Blockchain Node** - Software of Thought that maintains the blockchain.

**Concept** - A template developed by a community of developers to generate Thought agents.

**dDNA** - Digital DNA, which can exist within a Nuance.

**DLT** – Distributed Ledger Technology. Replicated, shared, and synchronized digital data concensus (with no central administrator) that exists across multiple sites, countries, or institutions.

**Fabric** - An abstraction layer built on all of the computer resources in Thought that allows Nuances to tell computers what to do.

**Fabric Node** - Software of Thought that ties individual devices into the fabric.

**Finite State Machine (FSM)** or simply a state machine, is a mathematical model of computation. It is an abstract machine that can be in exactly one of a finite number of states at any given time.

**Goal Set** - A defined set of outcomes.

**Nuance** - A hybrid data and application agent and the basis of all activity on Thought.

**Nuance Virtual Machine (NVM)** - Software of Thought that processes Nuances.

**Proof of Capability** - Similar to Proof of Evolution, Proof of Capability will verify a compute platform's capabilities as its useful "puzzle."

**Proof of Evolution** - Proof algorithm similar to Proof of Work used in cryptocurrencies such as Bitcoin, where instead of working through a difficult non-useful "puzzle", Proof of Evolution will process data agents as it's useful "puzzle".

**Thought** - A global decentralized network where hybrid data agents tell computers what to do.

**Utility Token** - The Thought token is required to purchase products and services from Thought.

# 3 Thought: Platform Overview

The Thought platform can be applied to any industry or data problem. It is based entirely around one of the smallest units in computing: the individual piece of data.

In the current paradigm, that piece of data is being routed through servers, stored in data centers, and compiled and sifted through big data analytics and artificial intelligence algorithms.

> The Thought Fabric consists of a group of computers registered with the Fabric consensus.
>
> These computers, called Nodes, advertise their capabilities and allow Nuances to move from Node to Node seeking the one that can complete the next requirement in their action logic. Nuances travel to Nodes on the Fabric to use their capabilities.

Thought is the groundbreaking introduction of a new paradigm; a platform that facilitates a holistic ecosystem of innovative developers and their applications relying on strong data governance to guarantee privacy, ownership, and provenance.

Thought has built a foundational, information transformation network with **data** as the commodity. The ecosystem hosts data-hungry applications for researchers in artificial intelligence and cognitive computing, in diverse industries such as healthcare, transportation, government, media, utilities, and finance. The ecosystem is fueled by monetization of sensor based-data and analytically rich data sets. Collection, transformation, and delivery of this data happens on and within the Fabric platform and ecosystem; and a blockchain-based token system coordinates activity on the Fabric.

The platform consists of three layers, which are described in detail in this white paper:

1. **Information Layer** - Nuances and Concepts that control the
   compute layer through the Fabric

2. **Fabric layer** - Thought software abstraction layer

3. **Compute layer** - All computer resources that are joined to the network



Information Layer

Nuances

Fabric Layer

Fabric nodes

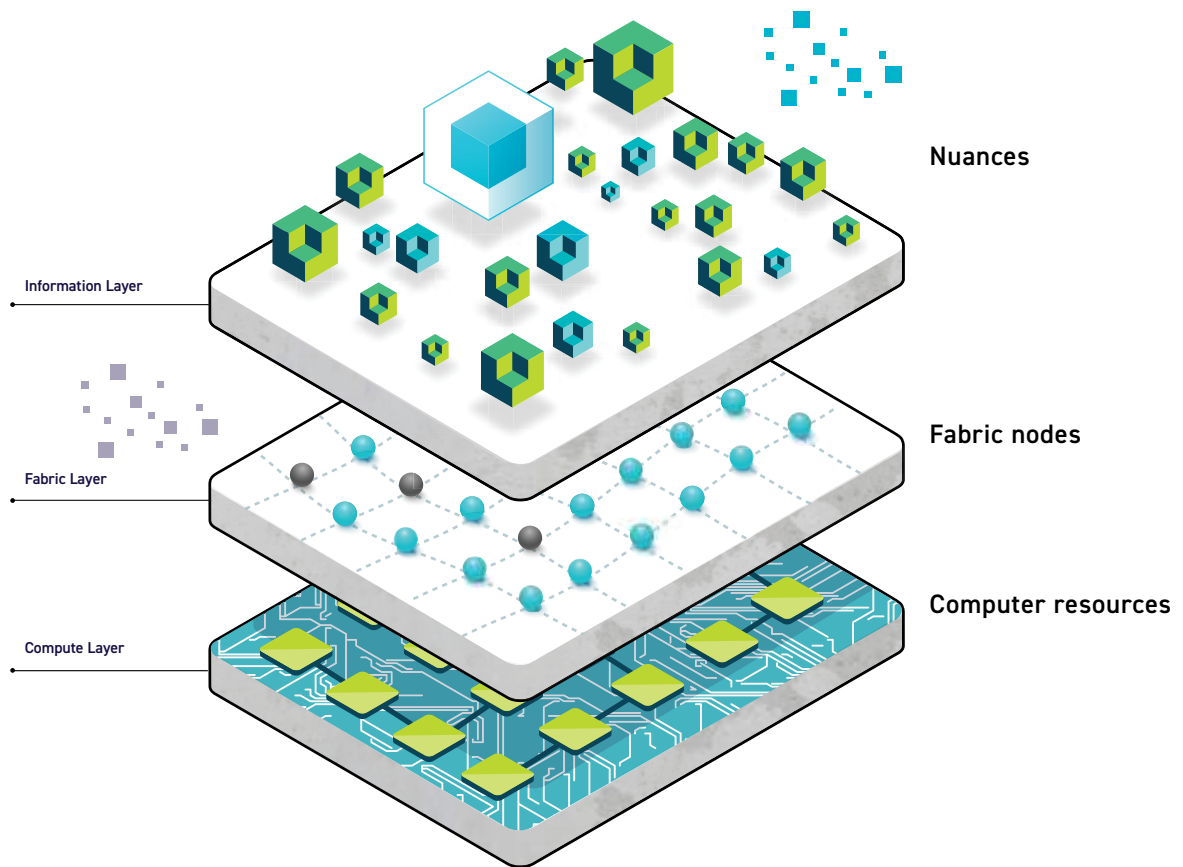Compute Layer

Computer resources

FIGURE 2: The Thought network is comprised of 3 distinct layers which serve specific functions.

# 4   Information Layer

This section describes what Concepts and Nuances are, as well as their function in the network.

## 4.1 Concepts (Templates)

In the most simple terms, Concepts are the templates that are used to create Nuances. They contain the logic which will be combined with data to create a Nuance. Concepts can contain logic used to form neural networks to goal sets to Markov Models. Any system of logic which can describe a Nuance's desired behavior may become the basis for a Concept.

### HOW CONCEPTS ARE CREATED

Concepts are submitted into the network by developers; and the developer of a Concept sets the price for the instantiation of a Nuance from that Concept. In order for a Nuance to be created, the person or entity who will be the "owner" of the Nuance pays the developer in Thought tokens. Developers are incentivized to provide a variety of useful Concepts. Concepts can be modified by developers; they are subject to splitting, merging, and evolution. Modification of a Concept is defined as evolution. A Concept can be split to satisfy design principles such as separation of concerns. A Concept can be merged to create Nuances with multiple sets of behavior. A developer can also generate a new Concept using two existing Concepts.

Goal sets can be chained where the next goal set is a child of the previous. A goal set could specify that another goal set be created once it is satisfied. This Concept is referred to as "spawning" a goal set. For more information on Goal Sets, see section 5.7 on page 40.

## 4.2 Nuances

Nuances are the smallest and arguably the most critical part of the network. They are comprised of multiple parts including data slices and application code; see Figure 3, below, and Figure 4 on the following page.

FIGURE 3: NUANCES
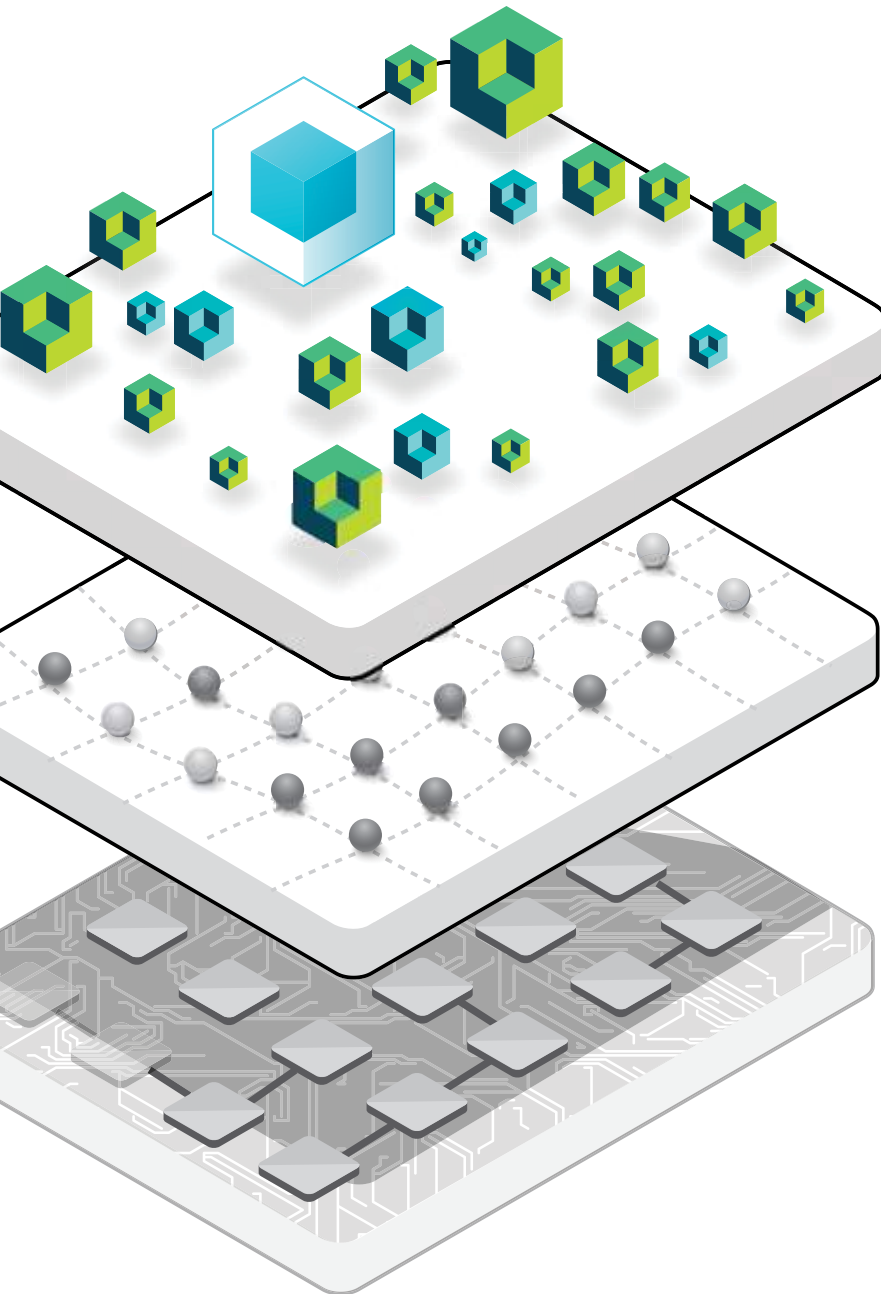
**Application code + Data slices**

**Anatomy
of a Nuance**

Brain

Data

Component

FIGURE 4: ANATOMY OF A NUANCE

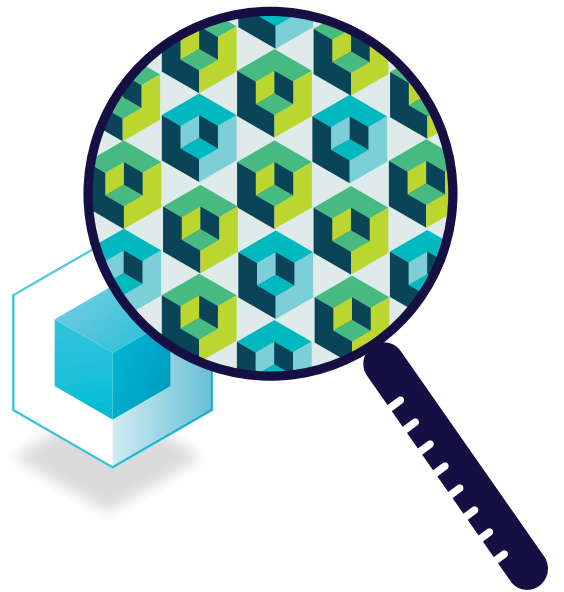## WHAT NUANCES DO

Nuances are the vehicles for transporting, organizing, sending, analyzing, and tracking data across the network and ultimately the globe. Nuances are intelligent; that is, they are encoded with AI capability that allows them to "understand" how to behave. Based on this capability combined with their encoded logic, they execute intended behavior, typically in groups that have the capability to intelligently gather, swarm, or flow through the network in the most efficient way possible. See Section 4.2.4 and associated Figure 5.

For example, a set of Nuances may be used to correlate two input streams of data and find the irregular patterns between the two streams. Consequently the AI will understand that this path is the correct one for its lifecycle, and route through the network automatically without needing to pass through gateway machines.

Nuance is encrypted at the outer layer and requires decryption to be accessed by the Fabric.

> Unlike the Concepts based in the blockchain that are executed on all Nodes, Nuances are processed only on the Nodes needed to perform their function. This is the core of what makes the system so efficient; Nuances meld data, logic, and security to enable pure widely distributable processing of information outside the blockchain.

**Thought**

## 4.2.2 How Nuances Work: Technical Detail

To accomplish their ultimate goal, Nuances flow through the Fabric, which consists of nodes with unique access to sensor, processor, and actuator capabilities of underlying compute networks. This is the core of what makes the system efficient; Nuances meld data, logic, and security to enable widely distributable processing of information leveraging blockchain for immutable structure and validation of processes. Nuances can contain raw sensor data, annotations, logs, or workflow histories. Nuances provide data provenance by embedding data genesis, ownership, quality relative to standards, and subsequent modification as it is processed in the Thought Network.

Unlike Concepts, which can be executed on all nodes, Nuances are processed only on the Nodes needed to perform their function.

All activity on the Thought network is derived from data-driven instructions that are executed on the Fabric nodes which create a universal insight computation Fabric. See Figure 8.

Nuances:

• Are configurable to customer-specific requirements

• Are applicable to any industry seeking to find valuable insights in large data sets while reducing the need for long-term or large-scale data storage

• Add context, security and ownership to any type of data

• Carry logic for routing and actions

• Reduce the cost of analysis and the need long-term data storage

• Provide a substrate for powerful emergent artificial intelligence and agent-based analytics

• Enforce data ownership agreements

• Track origins of data

• Enforce data boundaries (e.g. entity, jurisdiction)

• Find valuable insights in large data sets

## 4.2.3 Inter-Nuance Interaction

Nuances carry small bits of data on an individual basis but often act in groups when performing tasks or completing goal sets. Thus, in addition to Nuance hierarchy and Nuances being able to contain other Nuances, each Nuance has the ability to affect other Nuances:

1.  Single or multiple Nuances can affect single or multiple target Nuances

2.  Nuances can copy, exchange or swap characteristics
    or information from other Nuances

3.  A Control Nuance can create or terminate ("end of lifecycle,"
    or EOL) other Nuances based on rule sets

## 4.2.4 Nuance Hierarchy

Fabric nodes and Nuances have swarm functionality and nested hierarchy building within Concepts, similar to swarm and hierarchy research by Chen Hanning (see Ref). Examples of this are represented in Figure 5 below:

**FIGURE 5: NUANCE SWARM FUNCTIONALITY AND INTERACTION**
Within Fabric nodes, swarms of Nuances may interact to accomplish goal sets.



■ Nuance (Smart Data)

◌ Swarm

⟷ Interaction between Nuances

⟷ Interaction between Swarms

▢ Fabric Node
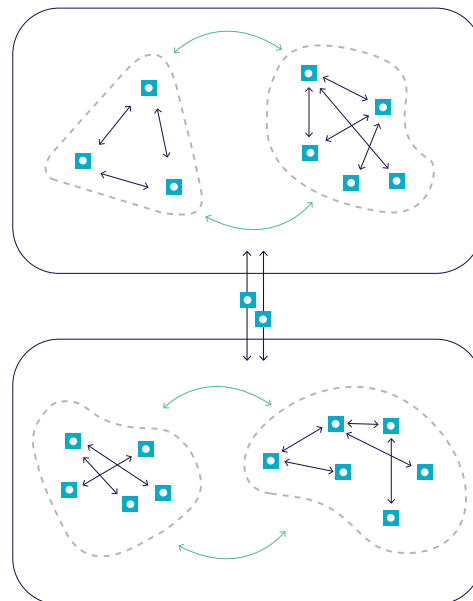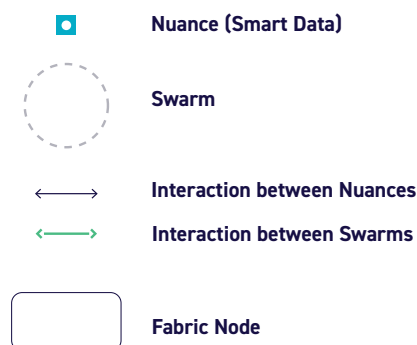
## "Digital DNA" Capability

Nuances can copy, exchange, or swap characteristics
or information from other Nuances.

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| A Data Landscape | Nuances spawned from Concepts with AI looking for broad patterns | Nuances spawned from Concepts with AI to accumulate records according to strict similarities | The two layers of Nuances communicate and spontaneously link to discover insights |
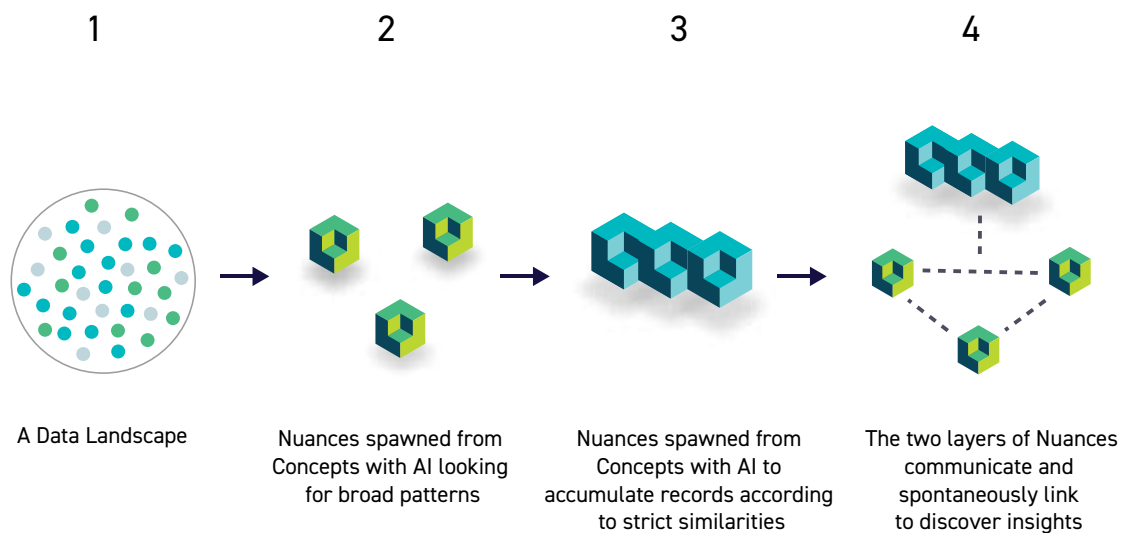
## 4.2.5 Nuance Lifecycle
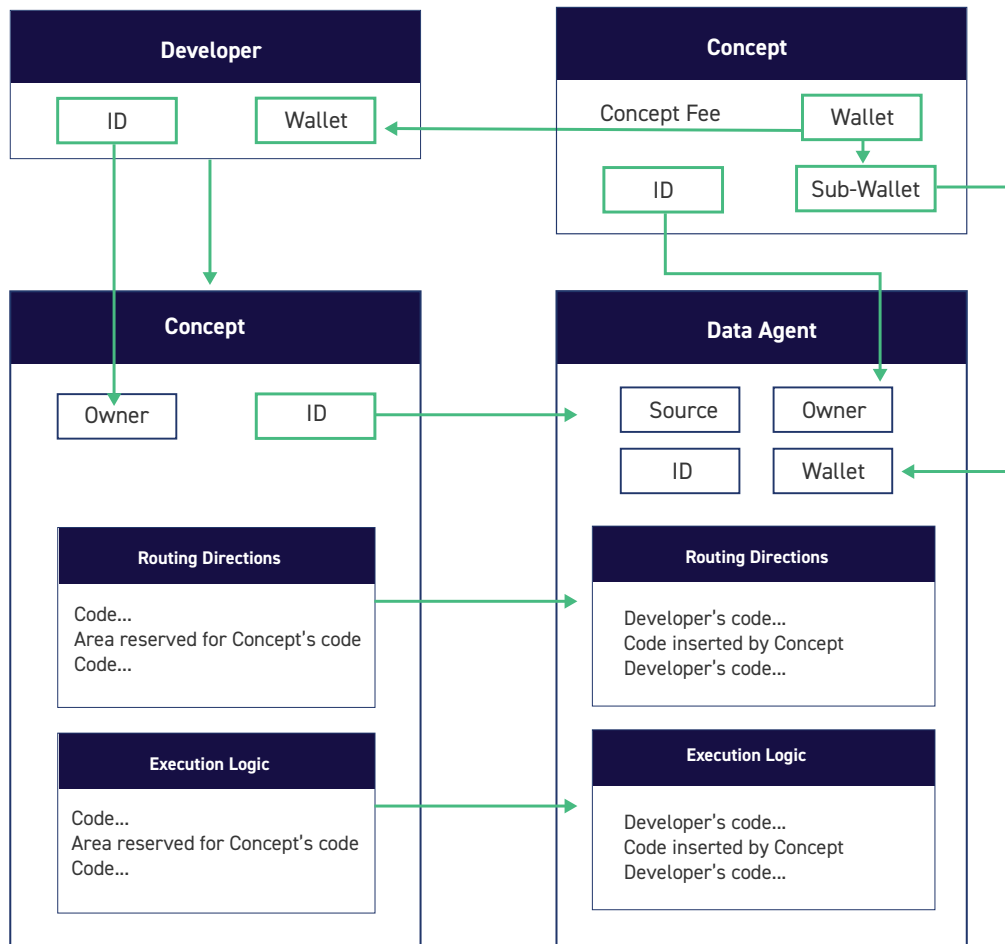
## Nuances: intelligent data vehicles

Nuances collect information about the Fabric
to select their next destination.

A set of Nuances spawned from an appropriate Concept will
distribute themselves through the Fabric as a swarm, actively
seeking the best performance/price for the processing they need.

This happens in the same way that birds flock
to find an optimal place to roost.

Nuances are created or "born" by joining data with a Concept. The owner of the Nuance pays the developer of the Concept in tokens when this occurs, as described in Section 4.1. Nuances continue to exist once created until they or their immediate parent publishes an end-of-lifecycle (EOL) notice into the consensus. Nuances with a published EOL cease to exist and are not processed or stored by any Fabric Node. In the case of a Nuance created directly by a person, the person is the parent who can publish the EOL notice. Nuances can also create (spawn) other Nuances in the same way that a person can. The Nuance has a wallet and must transfer tokens to the developer of the source Concept, thus becoming the parent of the new Nuance and gaining the right to publish its EOL.

The Id of the parent (person or Nuance) is incorporated into the Id
of any newly created Nuance. This makes it possible to:

- verify lineage of any Nuance back to the chain originator.

- verify that a request to publish a death notice (to kill a
  Nuance) is coming from the Nuance parent.

Nuances collect information about the Fabric to select their next destination:

A set of Nuances spawned from an appropriate Concept will distribute themselves through the Fabric as a swarm, actively seeking the best performance/price for the processing they need. This happens in the same way that birds flock to find an optimal place to roost.

Nuances can communicate and cooperate to achieve a combined function:

- A set of cooperating Nuances go to the same Fabric Node and
  reside there, providing their combined function as a colony.

- If one of these Nuances should fail for any reason, another one
  created from the same Concept will take its place.

- Nuances can reside on a Fabric node to provide an additional layer of interface
  acting as the avatar for the device. In this model, visiting Nuances "talk" to the avatar
  Nuance rather than using the Fabric node's components directly. Such an avatar
  Nuance can be easily re-spawned from the same Concept should it ever develop a
  fault, and can also be replaced by a new avatar Nuance with improved functionality.

- Nuances coexist and cooperate, so systems may be constructed of
  multiple avatar Nuances, each providing a different set of services
  wrapping the underlying functionality of the Fabric node.

Similar to device avatars, a Nuance may be dedicated to wrapping additional services
around a data stream. Such a Nuance would reside on the Fabric node closest to
the source of the stream and could, for example, provide stream-based analytics.

### 4.2.6 COMPLEX INTERACTIONS: SPAWNING, SPLITTING, MERGING, EVOLUTION, AND DIGITAL DNA

The Fabric is an evolving ecosystem where data from Nuances is transformed by the Fabric to
gain insight and knowledge about the meaning of that data and what it can do.

Nuances can perform the following functions to enhance value of information within Thought:

1. **Spawning** - Creating an offspring

2. **Splitting** - Dividing into multiple parts

3. **Merging** - Combining multiple parts

4. **Evolving** - Changing along a rule set

5. **Digital DNA** - Swapping/exchanging data

## NUANCE SPAWNING

Nuances spawn other Nuances by retrieving a Concept from the chain and spawning Nuances using the Concept. This forms an explicit hierarchy of Nuances.

### Nuance Spawning Example

If a Nuance with a Goal-Driven AI needed to do a large parallel search through some data set it might retrieve an intermediate level Concept for implementing such a search. If the full data set was not resident on a single Fabric node, then the intermediate level search Nuance might spawn a swarm of low-level collector Nuances to spread through the Fabric to collect the relevant data points.

### Nuance Splitting and Merging

Nuances split by spawning identical copies of themselves, in which each new split or merge contains a subset of the original Nuance data. The Nuances can then be processed in parallel on multiple Fabric member nodes.

## NUANCE EVOLUTION

Since a Nuance is a blend of data and application logic, any change to the data represents evolution of the Nuance. If the Concept (from which the Nuance was spawned) supports this, then the changing data may cause a change in the Nuance's behavior.

### Nuance Evolution Example

An example might be a simple "monitor Nuance" monitoring a sensor, for example a temperature sensor. As it monitored the sensor, it would repeatedly evolve its own data to contain the latest temperature reading. At some point, this collection of data might fit a pattern the monitor Nuance was checking for, perhaps a sustained high temperature above 100deg Celsius. At this point the Nuance would change its behavior and seek to find your phone so that it could inform you that the water was ready to be poured for tea. A Nuance created from a more complex Concept might go through many such behavioral transitions.

As an extension of the behaviors above, a Concept might provide the logic to manage and translate Digital DNA (dDNA). A Nuance spawned from such a Concept would hold its dDNA as its data and determine its behavior based on the content of that dDNA. When two such Nuances met, they could interact to trade that dDNA or spawn a new Nuance with some dDNA from each "parent." The newly evolved or newly-created Nuances would then proceed with new behavior as described in their new set of dDNA. Taken together, these behavior strategies provide a way for data to spontaneously reorganize in intelligent ways.

### Digital DNA (dDNA) Example

Consider the library of all Hubble telescope images: this large data set could be reorganized by time, by region of the sky, or any of a number of ways based purely on the metadata. In another way, each of the images is a large data set in need of processing to find features of interest.

A set of Concepts could be written to allow this collection to find patterns of interest. A master Nuance representing the whole collection could spawn one Nuance per image, each with the goal of having the image processed for features of potential interest. Each of these might spawn multiple swarms of Nuances: one swarm looking for binary star systems, another set of swarms looking for each of the typical classifications of stars, and so forth. These swarms of feature-tagging Nuances would spread through a global Fabric of home computers and cell phones, all capable of doing the relatively simple recognition tasks.

Once these feature-tagging Nuances started to return, the master Nuance could then spawn another set of Nuances, which would now look to verify these features over time by collecting the data from all the feature-tagging Nuances representing the same celestial object in the sky over the course of all observations. A cyclic variation in the brightness of any of these features might trigger a notification to astronomers that this celestial object is worthy of greater study as potentially harboring an exoplanet. Simultaneously, a set of Nuances using a dDNA pattern might start with dDNA encoding properties of the various celestial objects.

By swapping dDNA, tracking the swaps, and selectively sorting themselves to various Nodes according to selective similarity, they might discover patterns of commonality among celestial bodies that no human had thought to look for, thus uncovering entire new avenues for research.

## Nuance Protocol with Action Logic

```javascript
var originator = '%ORIGINATOR%';
var destination = '%DESTINATION%';
var statusKey = 'status';

function getCapabilityAndDefinition(action) {
  var foundCapability = null;
  var foundDefinition = null;
  var i, j, capability, invocationDefinitions, definition;

  capability_loop:
  for (i = 0; i <= capabilities.size(); i+=1) {
    capability = capabilities.get(i);
    invocationDefinitions = capability.getInvocationDefinitions();
    for (j = 0; j <= invocationDefinitions.size(); j+=1) {
      definition = invocationDefinitions.get(j);
      if (definition.getAction().toLowerCase() === action.toLowerCase()) {
        foundCapability = capability;
        foundDefinition = definition;
        break capability_loop;
      }
    }
  }

  return {
    capability : foundCapability,
    definition : foundDefinition
  };
}

function toggleRemoteStatus() {
  var map = getCapabilityAndDefinition("toggle");
  var capability = map.capability;
  var definition = map.definition;

  capability.invoke(self, definition);
  logger.info('toggling light');
}

function travel() {
  if (originator === localDeviceUuid) {
    // We're back home
    return;
  }

  var visited = data['visited'];
  visited.add(localDeviceUuid);
  var iterator = devices.getAll().iterator();
  while (iterator.hasNext()) {
    var device = iterator.next();
```

```
      if (!visited.contains(device.getUuid().toString()) && ! device.getU-
uid().toString().equals(originator)) {
          // We found another device we haven't visited yet
          device.sendNuance(self);
          return;
        }
      }

      // No where else to go -- return home
      var originatorDevice = devices.get(originator);

      if (null === originatorDevice) {
        logger.error('Originator device not found!\\n' + devices);
        return;
      }

      originatorDevice.sendNuance(self);
}

function go() {
  logger.info('originator ' + originator);
  logger.info('destination ' + destination);

  toggleRemoteStatus();
  travel();
}
```

# 5  Fabric Layer

The Fabric layer represents the protocol that facilitates the activity of the network. Nodes, Nuances, Concepts, and the native blockchain all employ the power of the Fabric to accomplish their respective goals. This section will discuss the various activities that occur within the Fabric layer, as well as the components performing those activities; and how they interact with one another.

These pieces are described below:

- The Fabric and associated Node types, blockchain nodes and Fabric nodes

- The Nuance Virtual Machine

- Fabric Core Functionality including Cybersecurity, Impact and Safety, and Useful Work Consensus

- Proof of Evolution

- The Blockchain Protocol

- Goal Sets

- The Native Token Economy; including how it functions and its token economics

## 5.1 Fabric

FIGURE 8
FABRIC LAYER



Information Layer

Fabric Layer

Compute Layer

Thought
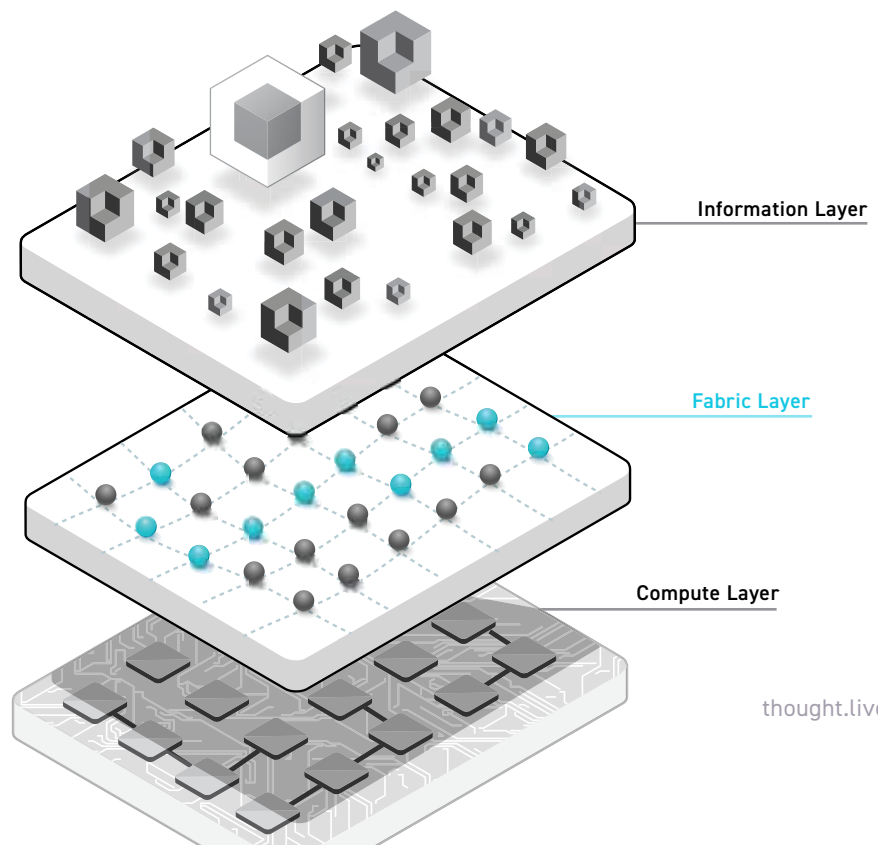
The Fabric enables peer-to-peer decentralized or centralized, transparent transactions of secure data: consisting of a group of computers (maybe as large as servers and maybe as small as wearable devices) that all register with the Fabric consensus. These computers advertise their capabilities and allow Nuances to move from member to member, seeking the one that can help them to complete their lifecycle and action logic. Nuances travel to members of the Fabric to use their capabilities.

The Fabric with its Nuances and blockchains is an evolving ecosystem with multiple actors that perform various functions:

- Processes Nuances with artificial intelligence capabilities

- Enforce Concept-created rules

- Maintain control and structural blockchains and token economics

- Transport Nuances to a destination based on its goal set

Nuances are transformed by members of the Fabric to gain insight and knowledge. Nuances themselves can spawn offspring which are further transformed. The chains connected to the Fabric store Concepts, identities, and other information within the consensus. If a chain were compromised, it would be possible to merge Concepts and other relevant information in the chain into the consensus of another chain; alternatively, a chain can be subdivided into sub chains in which two new chains are created and the information within the previous chain is migrated into the newly-spawned chains. The sub chains are said to be children of the original chain.

> The Fabric processes mimic the way physical commodities are produced, shipped, transformed, distributed, marketed, and sold across vast distances, based on trusted currency to ease the exchange between the various parties involved.

**Thought**

## 5.2 Node Types

The Thought architecture consists of globally distributed ledgers maintained by Blockchain Nodes that handle global-level value transactions and consensus along with validation of identities of Nuance processors called Fabric Nodes. The highest level Fabric Nodes create a scaffolding of linked sub-chains that are autonomously generated to account for entity, government, jurisdictional, and institutional boundaries as required.

### 5.2.1 BLOCKCHAIN NODES

Nodes participate in customized blockchains to facilitate activities that require consensus. Among these are the various developer, owner, and device credentials known to the Fabric, the library of Nuance Concepts, and the uniqueness and provenance of individual Nuances. The set of Fabric nodes processing a given blockchain can be considered to be localized to that blockchain; participation in the Fabric is bound by participation in the associated blockchain. Once created, Nuances flow directly from node to node in the Fabric and are not constrained by the processing rate of the blockchain. However, the nodes in the Fabric use the blockchain to verify the validity, security, and trustworthiness of all Nuances they receive. Nuances are encrypted in transit, with the keys distributed through the blockchain. Thus, Nuances can be considered to be naturally private to the blockchain from which they originated.

### 5.2.2 FABRIC NODES

Fabric nodes may participate in more than one blockchain, and can process Nuances from a variety of originators. This provides an additional solution to the limitations of individual blockchains. Should the functionality of a Fabric be constrained by either the processing rate of a particular blockchain or by some market event, a new blockchain can be initiated and populated with the credentials, Concepts, and contracts needed for the specific goal of the owner of the Fabric.

## 5.3 The Nuance Virtual Machine

The Nuance Virtual Machine (NVM) is the environment in which Nuances are executed, and Concepts are written. For now, JavaScript Nuances are processed by the NVM and then routed or executed by the Fabric Node. The JS is stripped of "eval" and "Function()" calls for safety, along with additional pre-

processing to ensure syntax and semantics are validated by the NVM. In the future, Thought will implement domain-specific language for the NVM.

The NVM allows for Concepts to interact with different blockchain back ends, as well as access methods available for routing and common action logic.

Fabric Nodes directly access raw data streams and instantiate Nuances based on Concepts accepted by the Thought network at the relevant level chain (global versus sub-level). There will be additional Thought network sub-functions required to ensure stability and operating parameters which include a multiple hierarchical goal system, an impact evaluation system, cybersecurity system, and malfeasance management system.

FIGURE 9: FABRIC OVERVIEW
Sample process flow of a Nuance flowing through the Fabric layer to accomplish a task. In this case, the Nuance visits a series of sensors before taking action based on the information it gathers from those sensors; intelligently producing action from a series of inputs.
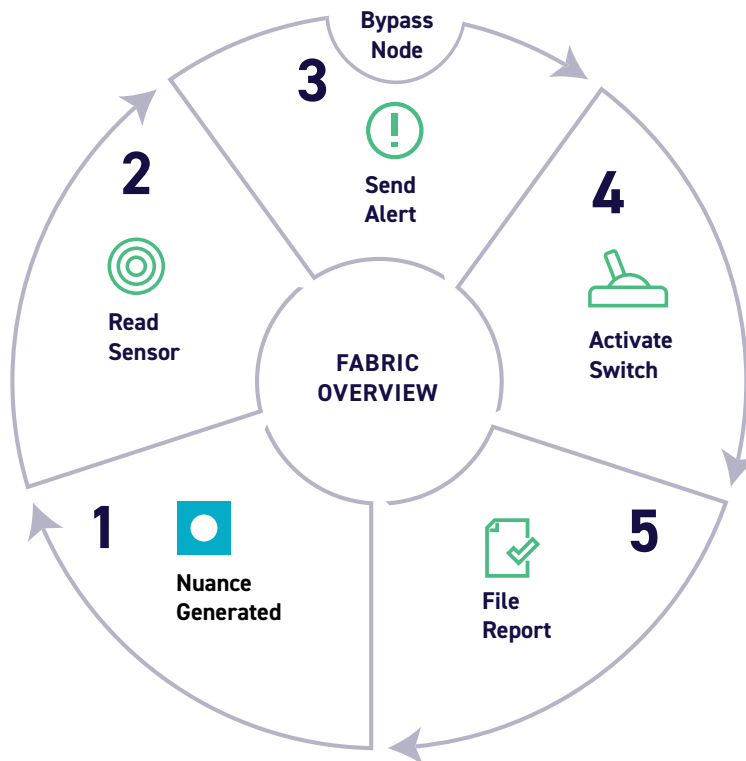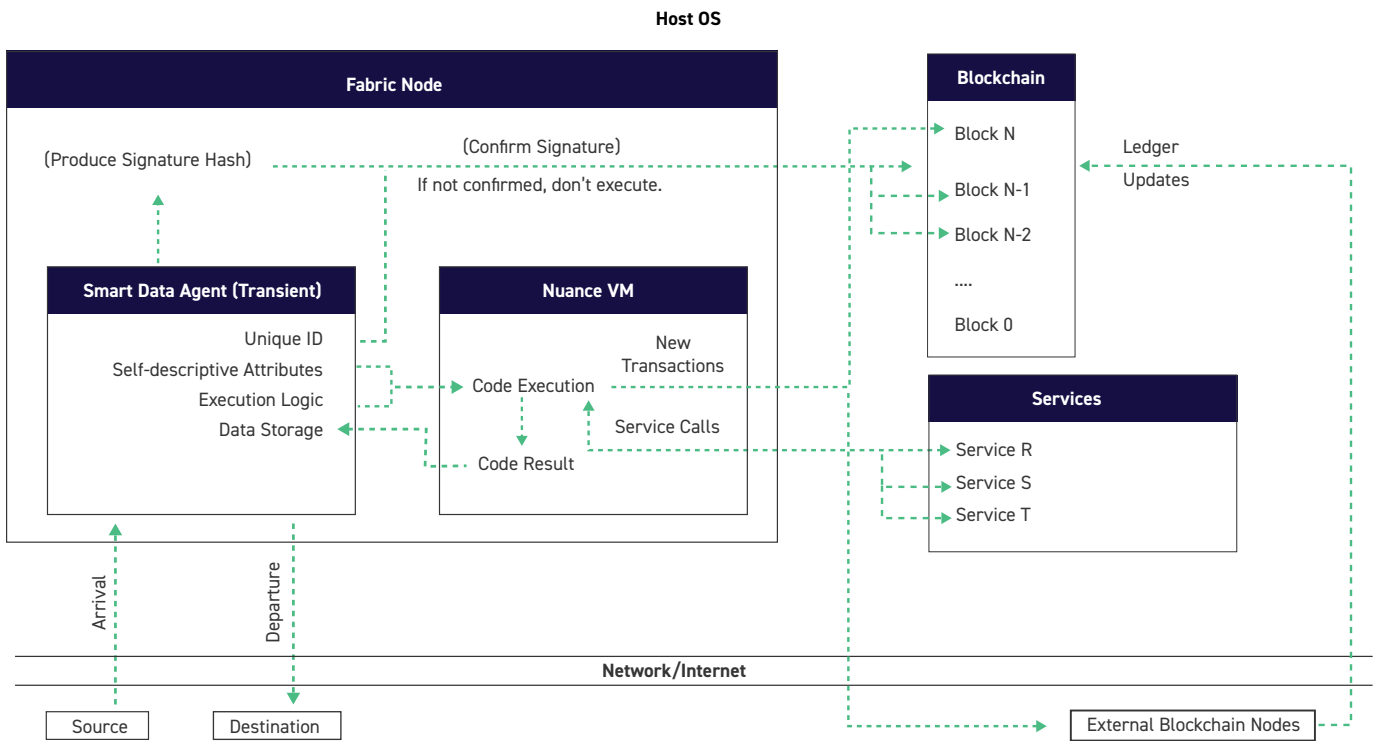
FIGURE 10: NUANCE BLOCKCHAIN INTERACTION

The Thought network and protocol create a decentralized data-processing Fabric based on a mature and secure blockchain and a native token. Clients spend tokens to have data converted to Nuances, processed on the network, and in return, gain insights from the network. The Fabric handles data processing requests via universal data exchange markets: Clients and miners set the prices for the services requested and offered and submit their orders to the markets.

The markets are operated by the Nuance automata, which employs Proof of Evolution to guarantee that miners have correctly processed the Nuance they committed to process and a Proof of Capability to guarantee that Fabric nodes processing Nuances have the compute and resource capabilities claimed.

## 5.4 Fabric Core Functionality

### 5.4.1 CYBERSECURITY

The system incorporates multiple levels of data-level cybersecurity with adjustable, multi-layered and dynamic cryptographic strengths. Each Nuance is fully encrypted on the outer layer and requires decryption at an Fabric node for access. Within the Nuance there can be multiple additional levels of encryption at varying strengths. Nuances will also contain authorization and access control logic to determine if a particular Fabric node can access various internal encryption layers.

### 5.4.2 IMPACT AND SAFETY

The network will employ a system to gauge impact of certain logic and rule sets and appropriately define levels of energy spent on validating the logic and creating safety and redundancy features. For example, a single temperature reading from an IoT sensor may have a low impact and be assigned a low value $I_c = Life[5]+Rednc[0]+Safety[10]=.0001$, where an alert Nuance affecting operating conditions at a nuclear power plant may have a very high impact $I_z = Life[1M] + Rednc[10]+Safety[1000]=12,256$. The system will use the Impact rating to affect network processing such that it might require more computing participants to validate the data processing transaction and perform goal conflict resolution.

For capabilities that are actuators (in other words they make a change in the physical world), Thought may require that the Nuance present a special token. (See more about the Token in Section 6.) This token is mediated by the creating Concept and so can have specific rules built into it. In the example of something that would be turning of one electrical grid of a major city, it might require that a token be presented and valid which was only valid if a series of fingerprints had been entered and verified.

Nuances

Information Layer

Fabric Layer

Fabric Nodes
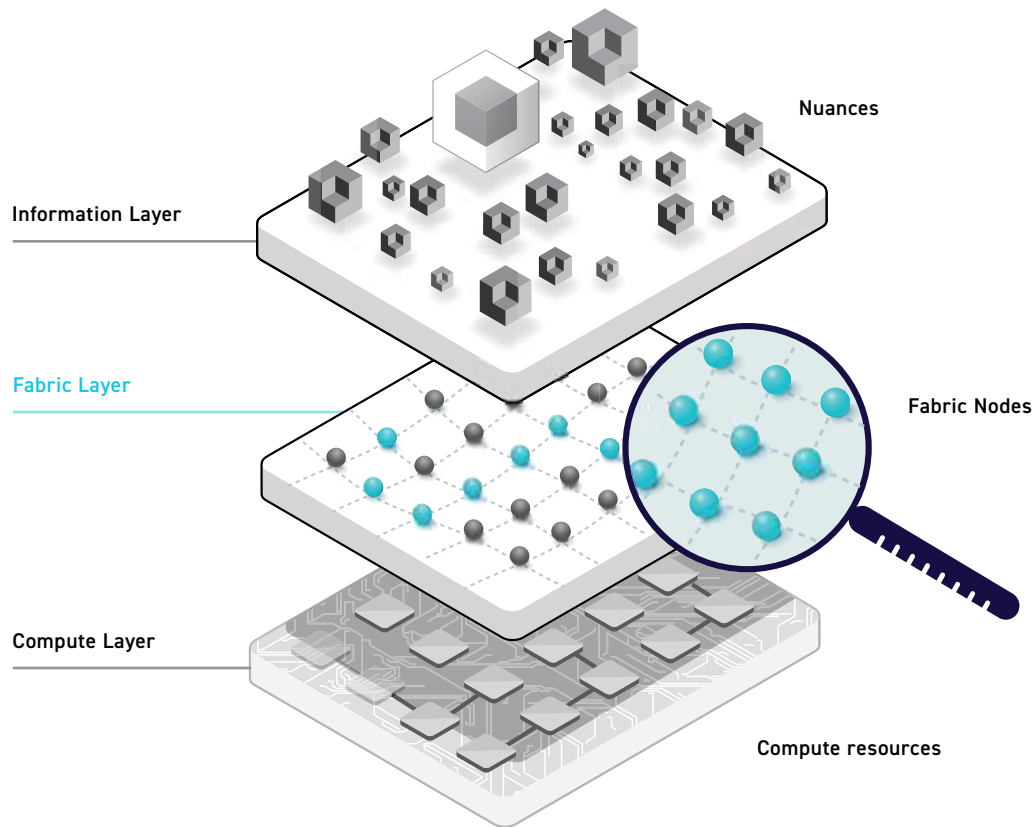
Compute Layer

Compute resources

FIGURE 11: FABRIC NODES

### 5.4.3 USEFUL WORK CONSENSUS

Various Thought network functions can be implemented on any consensus mechanism that achieves Thought network protocol validation. One of the development goals for the Thought network's usage of consensus protocols will be to minimize wasteful work and instead focus on computations that benefit the network by chain maintenance, goal set conflict resolution, and the processing of Concepts and Nuances.

## 5.5 Proof of evolution, useful work

Proof of evolution (PoE) is a model of verifying that a node within the Fabric processed a Nuance as requested. The verification work serves as a useful way to provide security to the Fabric.

> Businesses that already have their own blockchain running will be able to seamlessly integrate into the wider blockchain network, while still using their own private blockchains and consensus algorithms.

Nodes within the Fabric are not fully replicated state machines. Each node can have its own set of capabilities to perform work on behalf of a Nuance. Thought has devised a method to determine if a node has indeed performed the processing of Nuance on behalf of another node, i.e. PoE.

**Technical detail**

Using a fully homomorphic and quantum computing proof encryption scheme, where $E$ is the encryption function and $D$ is the decryption function, for a given Nuance with data $d$, the Nuance is defined as a sequential set of functions $F$ that each are a capability on a foreign node or behavior within the Nuance whose input is $d$ or the output of the previous function in the set. It can then be said that if $F$ is the arbitrary set of functions { $A, B, C$ } and $B$ is a capability on a foreign node within the Fabric, when processed by a foreign node, the Nuance with data $d$, will be processed in the following manner:

$$A(d) = O_1 \rightarrow B(O_1) = O_2 \rightarrow C(O_2) = T$$

where $T$ is the transformed data of the Nuance, using a fully homomorphic encryption scheme, it can be said that if d is encrypted, the data of the Nuance is $E(d)$ with transformed data $T$.

To verify that the foreign node is honest about its processing, it will be asked to hand back a mathematical proof $P$ such that $P \equiv F$. The original sending node and all other Fabric members can use this proof to check that $P(d) \equiv D(T)$. The proof is distributed amongst nodes in a random order fashion with each node processing an encrypted portion of the proof and no node is aware of which node owns which piece of the proof. In addition, if the hashes of $d$ and $D(T)$ are the same, then it can be shown that no transformations were applied to the Nuances' data.

> Individual Nodes run software that lets them work on/for the Nuances. When Nodes start up, they connect to the chain(s) that they need to operate and they talk to each other. The set of all the nodes connected to a particular chain is the Fabric for that chain. A Node may be participating in more than one chain and thus be part of more than one Fabric. Such Nodes can act as bridges between Fabrics.

**Thought**

## 5.6 Blockchain Protocol

On the top-level chain, the blockchain network will employ a modified consensus algorithm with a Proof of Evolution component. For the sub chains on the network, users are free to use whichever consensus algorithm they wish for their private chains. This feature will primarily allow businesses that already have their own blockchain running to be able to seamlessly integrate into the wider blockchain network thus allowing them access to the Fabric. The ledger can be split into multiple sub-ledgers. This is not to be confused with a hard fork or soft fork. A ledger that has split is merely a set of ledgers with the same transactions, but a subset of Concepts in the original ledger.

A ledger can also be merged with another ledger. When this occurs, a new ledger is generated containing the set of Concepts in the original ledgers or a subset of those Concepts.

A ledger can be spawned from another ledger. This ledger is linked to the original ledger. This new ledger can inherit data and Concepts from its parent ledger. A new token can be specified for the child ledger also.

A ledger evolves over time as new data and Concepts are added to it. Data and Concept are valued within the network, therefore the potential value of ledger increases with time. In this manner, ledger networks can compete with one another to produce valuable data and insight.

### MULTI-LEVEL DYNAMIC CONSENSUS

On the top-level chain the blockchain network will employ a modified consensus algorithm with a Proof of Evolution component. For the subchains on the network, users are free to use whichever consensus algorithm they wish for their private chains. This feature will primarily allow businesses that already have their own blockchain running to be able to seamlessly integrate into the wider blockchain network, thus allowing them access to the Fabric.

Blockchains within the Fabric are depicted here as global, corporate, and private chains. These chains may exchange certain types of information while retaining on-chain privacy of other types of information. Nuances facilitate the data flow between various chains.
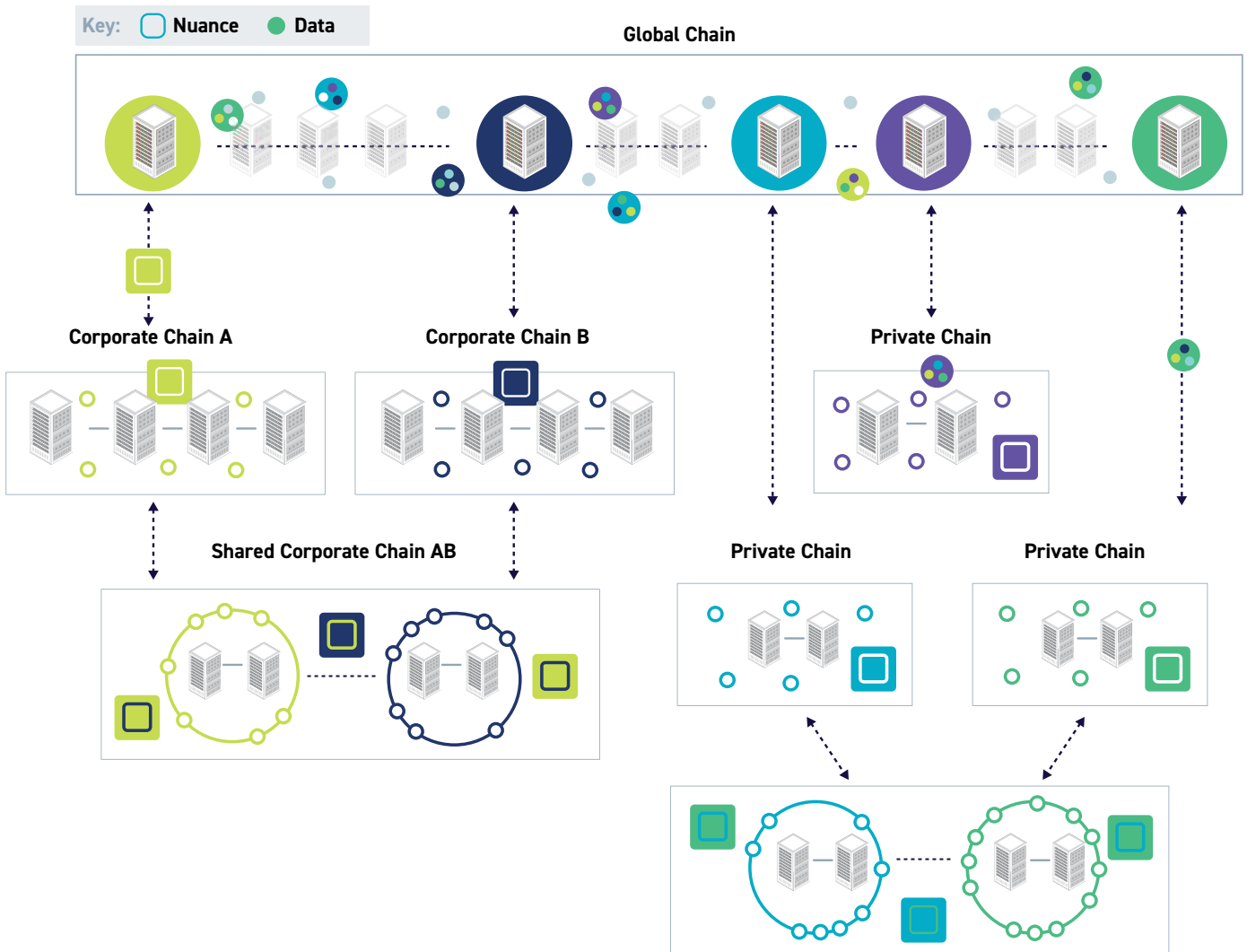
## 5.7 Goal Sets

The Thought network is a structure for defining and enforcing goal sets and managing goal set conflicts. To that end, one of the most important guiding principles in the Thought network will be to have the ability to define and follow goal sets. The Thought network is a structure for defining and enforcing goal sets and managing goal set conflicts. Goal sets will be defined in Concepts at different levels of the Architecture hierarchy.

Goal sets, as their name indicates, consist of a set of goals defined by a) the network, b) a blockchain, c) a Concept. A high-level goal set for the network, for example, might be "Don't destroy the earth." For a blockchain, a goal set might be "Don't lose any data." A community-defined Concept could be "optimize traffic on route 81 corridor from mile marker 24 to 67."

Goal sets can be global, high level and far-reaching such as "optimize energy usage in City X" or low level such as "ensure redundancy of output data from analytics model Y."

Thought network implements goal sets as follows:

Nuances with goal sets defined by Concepts will be similar to autonomous AI bots in video games similar to Goal Oriented Action Planning architecture using Finite State Machines (FSM) in game development. (Source: Brent Owens, https://gamedevelopment.tutsplus.com/tutorials/goal-oriented-action-planning-for-a-smarter-ai--cms-20793).

 "Goal oriented action planning is an artificial intelligence system for agents that allows them to plan a sequence of actions to satisfy a particular goal. The particular sequence of actions depends not only on the goal but also on the current state of the world and the agent. This means that if the same goal is supplied for different agents or world states, you can get a completely different sequence of actions, which makes the AI more dynamic and realistic."

The GOAP model includes actions with preconditions, effects, costs and FSM states. An example from the sourced research would be an agent whose goal is to make firewood. Examples of preconditions are "needs an ax to chop firewood."
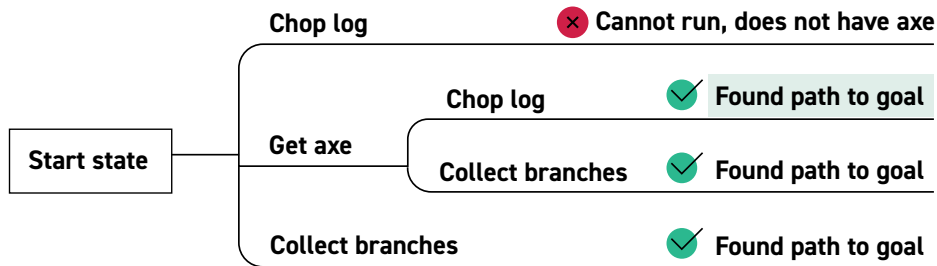
Consensus is reached using the consensus algorithm of the underlying blockchain. All conflicts in the consensus are resolved in this manner.

Conflicting goal sets are simple. Goal sets desire certain information across multiple consensus to be true. If a node or group of nodes wishes for members of the Fabric to verify a certain goal set, then those verifying nodes expect to be compensated. It should be up to the node compensating the other nodes that what it is asking for can be verified, or else it will not be able to realize its goal set.

Goal sets themselves can be split across multiple consensuses. The aggregated verifications of these goal sets determine if the goal set is satisfied. In this manner, goal sets can be merged to form a higher-level goal set whose conditions are satisfied by its sub-goal sets.

Furthermore, if goal sets can spawn goal sets and goal sets can be satisfied by sub-goal sets then goal sets can be treated as continuous moving goals. In this manner, newly spawned goals are treated as sub-goal sets. Each goal creates a new goal while remaining a sub goal of the original goal set. This is referred to as evolution of goal sets within the network.

## 5.8 The Native Token Economy

The Thought Network has a rich token economy. Tokens are earned by the owners of Fabric Nodes in exchange for the evaluation and routing of Nuances. Nuances have wallets and exchange tokens for routing and evaluation as well as for access to the unique components and capabilities of the Fabric Node. This incentivizes people to run Fabric Node instances and join them to the Fabric. Tokens will also be awarded for hardware integration development, Concept development and access to data streams and repositories.

### 5.8.1 HOW THE TOKEN ECONOMY FUNCTIONS

Fabric Nodes publish offers of data and services into the consensus. Nuances then select and travel to the Fabric Node with the best offer for the data or service they need. This incentivizes Fabric Node owners to offer high value data and services, thus making the overall Fabric more economical. Fabric Node owners are also incentivized to offer data and services that are in high demand.

The system of publishing offers into the consensus and having them claimed by Nuances constitutes a market. When offers are for data, a data market emerges wherein the value of different types of data is determined by the supply/demand interplay between the Nuances and the Fabric Nodes. When offers are for analytics services, a market for insight is created.

Within the context of the Fabric and its underlying blockchain, tokens have many uses. Links to data and information can be purchased using tokens from the blockchain. A transaction that satisfies the conditions of sale, such as supplying the data owner with tokens, needs only be sent into the chain to retrieve the data and/or information.

Similarly, data and/or information can be requested within the chain. A pub/sub Nuance serves as an advertisement for a particular type of data and/or piece of information. Any node in the blockchain can send the requested data/information to the contract and be compensated by the contract owner with tokens.

Outside of data and information, Concepts can also be added to the chain and purchased by nodes. Concepts are stored within the appropriate chain. To purchase a Concept, a transaction is issued by a requesting node in the blockchain. The price of the Concept is set by the Concept owner.

If the transaction of the requesting node satisfies the asking price of the Concept owner, the chain dispenses the Concept to the requesting node.

The requesting node can have a corresponding node within the Fabric generate a Nuance from the Concept. The Fabric node can then begin processing the Nuance. However, if the Fabric node lacks the capabilities needed to process the Nuance, then the node can send the Nuance to another node in the Fabric that does. The another Fabric will begin processing the Nuance and will be compensated with token from the requested Fabric node.

Tokens can also be awarded through the distribution mechanism of the underlying blockchain. In addition, tokens can be given by satisfying goal sets within the blockchain, Fabric or Nuance.
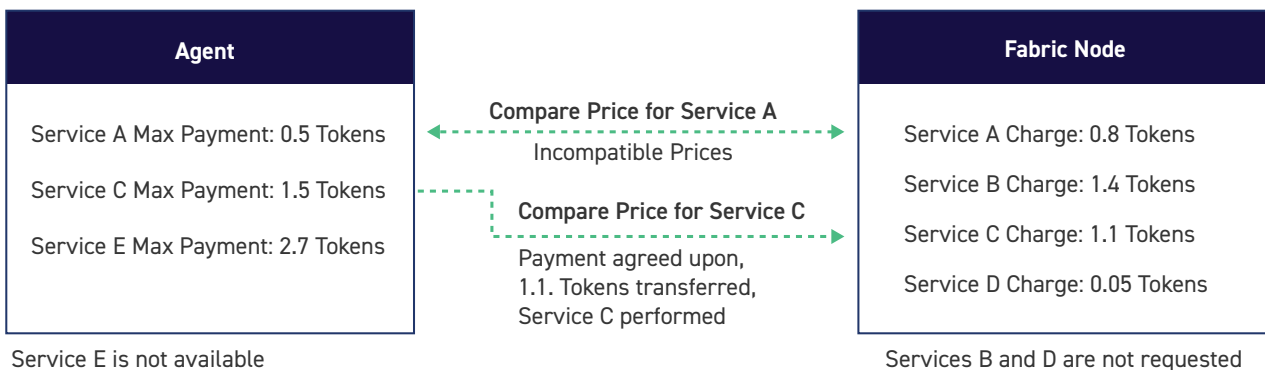
### 5.8.2 TOKEN ECONOMICS

Token economics function in the following ways:

- Tokens buy data as well as data insight (analytics, predictive, prescription machine learning)

- Tokens buy usage of Concepts (data models and applications on the network)

- Tokens buy data, which is transferred from entity to entity

- Tokens are earned by mining (processing Concepts and Nuances), creating Concepts, contributing data models, analytics models, artificial intelligence models, data sets to the network

- Tokens are consumed by the execution of tasks by Concepts and Nuances

Outside of data and information, Concepts can also be added to the chain and purchased by nodes. Concepts are stored within smart contracts. To purchase a Concept, a transaction is issued by a requesting node in the blockchain. The price of the Concept is set by the Concept owner/developer. If the transaction of the requesting node satisfies the asking price of the Concept owner, the smart chain dispenses the Concept to the requesting node.

The requesting node can have a corresponding node within the Fabric generate a Nuance from the Concept. The Fabric node can then begin processing the Nuance. However, if the Fabric node lacks the capabilities needed to process the Nuance, then the node can send the Nuance to another node in the Fabric that does. The another Fabric will begin processing the Nuance and will be compensated with tokens from the requested Fabric node. Token can also be awarded through the distribution mechanism of the underlying blockchain.

FIGURE 14: VALUE EXCHANGE

| Agent | | Fabric Node |
|---|---|---|
| Service A Max Payment: 0.5 Tokens | Compare Price for Service A<br>Incompatible Prices | Service A Charge: 0.8 Tokens |
| Service C Max Payment: 1.5 Tokens | Compare Price for Service C | Service B Charge: 1.4 Tokens |
| Service E Max Payment: 2.7 Tokens | Payment agreed upon,<br>1.1. Tokens transferred,<br>Service C performed | Service C Charge: 1.1 Tokens |
| | | Service D Charge: 0.05 Tokens |

Service E is not available

Services B and D are not requested

# 6 Compute Layer

The Compute layer consists of all computer resources available to Thought Network including CPU, GPU, memory, storage, and computer components such as sensors.

## In Brief

Fabric nodes are responsible for creating and managing component architectures to interface with Computer layer resources; the Compute layer consists of all node-local resources available to the Thought Network. Examples include CPU, GPU, local storage, physical resources such as sensors and actuators, and specialized processing capacities which might exist in hardware or software. In some cases, a node may act as a gateway for services not otherwise accessible to the Thought network. These resources are still considered node-local because they are only available to Nuances while they are on this Fabric node.

Fabric nodes provide access to their node-local resources through a component architecture configured in collaboration with the node owner and managed by the node. Fabric nodes advertise their components, capabilities, and costs into the Fabric Layer.

FIGURE 15:
COMPUTE LAYER



Nuances

Information Layer

Fabric Nodes

Fabric Layer

Compute resources

Compute Layer
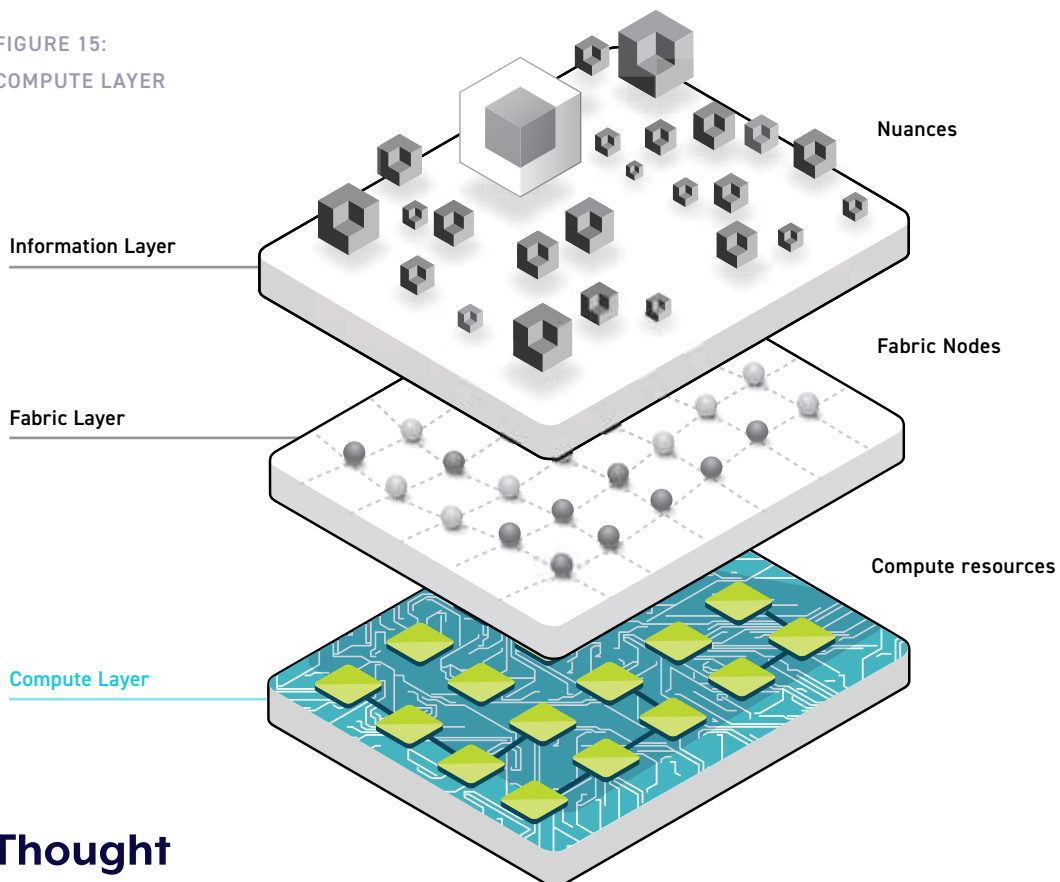
# 7    Use Cases

## 7.1 Healthcare

Thought provides a means to leverage the significant opportunity to engage with the individual patient more closely and import data from mobile health applications or connected devices. This interaction with the patient will result in the collection of more detailed clinical, environmental, and lifestyle information, such as heart frequency and body temperature, physical activity and nutrition habits, and sleep and stress management, which will prevent risk exposure and onset of disease. Personal monitoring over time should aid in early detection of deviations from a healthy state and trajectories should lead to actionable recommendations, making it possible for individuals to maintain health. However, many aspects that are specific to big data in health research need to be taken into account, such as data heterogeneity, institutional and legal fragmentation, and strong data protection standards.

The Thought network can provide data that is HIPAA compliant right from the data source using a multilayered encryption framework that can fully encrypt patient data at all times. Furthermore, privacy can be maintained in scenarios where researchers require access to derivative data (i.e. data that results from some analytic of the original data), that can be de-identified so that researchers have access to information about the data without accessing the data itself.

## 7.2 Weather Stations

A company that manufacturers weather stations wants to 1) be able to easily generate local and regional weather data for mapping and forecasts, 2) provide a means to offset the cost of ownership and maintenance, and 3) easily sell the maps and forecasts to the purchasers of the weather stations.

A micro-network is installed on each weather station to use wifi and connect to the internet; then all weather stations join together in an Fabric unified by a company consensus. In the consensus, the company hosts Concepts for various types of weather maps and reports. When they use Nuances to get these maps and reports, the owners of the weather stations earn tokens in exchange for the local weather data and processing power consumed by those Nuances.

The network provides consumer access by joining all the weather stations to a second consensus. Concepts are placed into this consensus for requesting various maps and reports from the company. Consumers with weather stations can spend tokens to create Nuances that fetch and then display various maps and reports. The company also provides a way to buy tokens.

Later, the company opens the consumer consensus to outside developers. These developers join additional devices to the consensus and publish new Concepts that allow weather stations to interact with automated blinds and other solar heat management systems to boost home heating and cooling efficiency by taking predictive action based on information available through the surrounding weather stations.

## 7.3 Distributed Algorithms

Smart Data not only enables data to protect itself from unauthorized access and to define how it should be routed and processed, but allows data to take advantage of the distributed Fabric to enable massively parallel processing and distributed analytics. This leverages the network of devices, nodes, routers, and other networks that are joined to the Fabric as a sort of global data-defined supercomputer under the direction of the smart data itself. Smart data Nuances can carry the algorithms that enable analysis or machine learning, and can use available nodes to process their data payload and find insights, relationships, and correlations without being bound to large centralized data stores and systems.

There are many forms of distributed analytics, from simple descriptive statistics such as aggregation and summarization to more predictive analytics such as regression analysis, to machine learning and artificial intelligence processes such as deep neural networks. These types of analytics are usually implemented in a clustered computing environment, for example using Apache Spark, Storm, or MapReduce. These are fixed, in-place solutions requiring permanent compute resources, installation, configuration, and management. In contrast, a data-defined solution will allow each smart data Nuance to find nodes in the Fabric that have the compute capability it requires and leverage each node as part of its analytic mission, then execute the analytic, gather the results, and deliver them as defined in the smart data Concept.

## A) DISTRIBUTED REGRESSION

Health care organizations gather patient data using a variety of connected electronic devices. To protect the privacy of patients, HIPAA regulations strictly govern what data can be shared among these organizations, which conflicts with the desire of these organizations to share data for research purposes. Smart data and distributed regression algorithms can alleviate this conflict. Within an organization, smart data enabled sensors collect patient data and compute local regression coefficients. These sensors share the result of the regression, without patient-identifying data attached, with all of the other sensors on the local Fabric, which is then combined to produce a mean regression coefficient for the data within a facility. This information, which is non-attributable to any given patient, can be shared with other facilities and further incorporated into a global regression result.

## B) DISTRIBUTED NEURAL NETWORK

Smart homes outfitted with IoT devices are becoming common - everything from thermostats to refrigerators that can order food and toasters that "remember" how you like your toast. What is not yet common is these devices working together in a meaningful fashion.

Neural networks will change that through smart data. Your home can build a model of your preferences, schedule, and habits, and configure itself to best support you. New devices added to the household can benefit from the learning of devices that already exist, reducing or eliminating the need for extensive configuration, while existing devices can use the new inputs from a new device to improve the overall network model.

# Machine Learning Algorithm Examples

## C) BAYESIAN NETWORK AUTOMATA: ALGORITHM MODEL

Concepts on the Thought Network can be used to create Nuances that act as nodes in an unbounded Bayesian Network Automata configuration similar to research by James Henderson; this can be located at http://cui.unige.ch/~hendersj/papers/henderson_iwpt11.pdf.

In this model Nuances can represent Inputs, operations, arguments, actions and states as well as outputs, as in Figure 16 below.

FIGURE 16:
BAYESIAN NETWORK
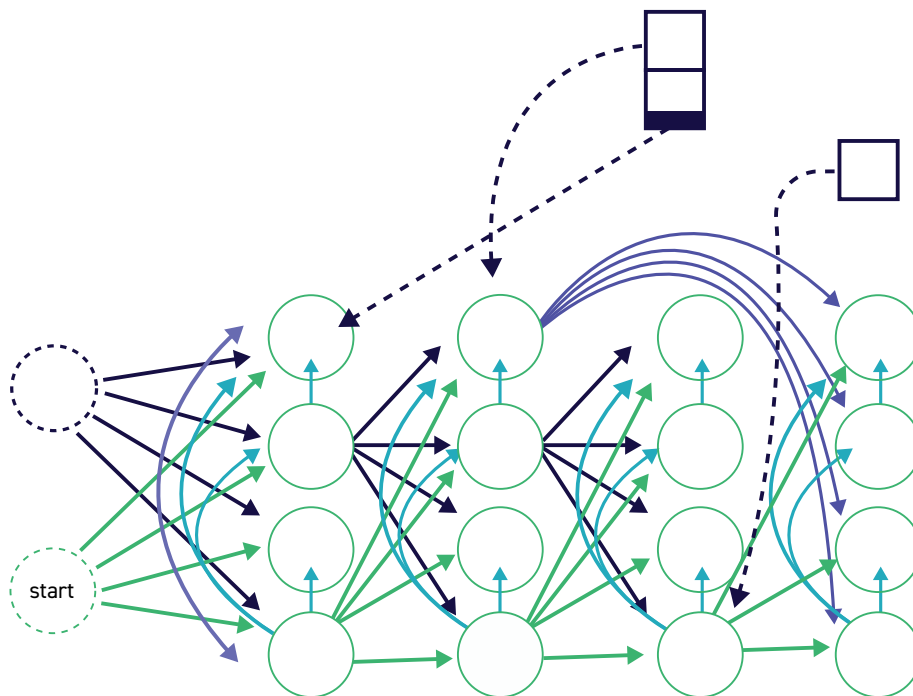
## D) NUANCES AS HEBBIAN LEARNING NODES: HEBBIAN MODEL

In a Hebbian learning system, the weight of connections between model neurons are tracked and adjusted based on how frequently they activate simultaneously in the same direction (positive or negative). Nuances can model artificial neurons and contain connection links to other Nuances as well as to data sources and Fabric node components. To utilize the Hebbian model, a Nuance would implement connection weighting logic potentially based on certain activities or frequency of a certain condition executing, or "learning." Additionally, Nuances can distribute or move themselves among many Fabric nodes to increase or decrease weight values and accomplish a similar learning goal. Additionally, weighting factors and triggering logic can be varied based on many complex conditions or dynamic logic, and even monetary conditions, improving upon the functionality of a basic artificial neuron. Nuances can model other biological functions as well.

Simplified Hebbian model applied to Nuance: $w(sub_i \ sub_j) = x_{sub_i}$ times $x_{sub_j}$ where the connection weight $w_{sub_i sub_j}$ between Nuance i and Nuance j equals $x_{sub_i}$ input to Nuance i.

## E) NUANCES AS MARCOV MODEL NODES

Nuances can simulate state machines in various Markov models. Each Nuance would contain data that held state information and logic that would simulate a Markov chain, for example. Nuance Markov nodes would connect to other nodes in the same algorithm model and contain adjustable state transition parameters and probabilities. Nuances would be designed to connect to inputs and outputs, according to the data models.
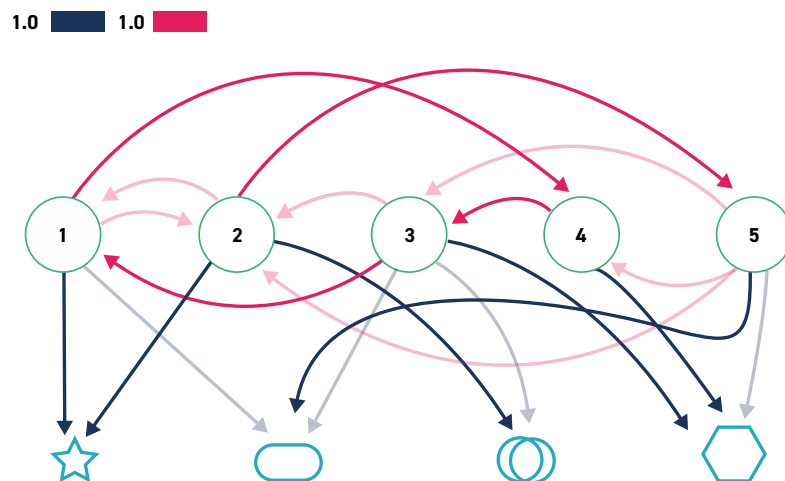


FIGURE 17:

MARKCOV CHAINS

## 7.4 Cybersecurity

Smart Data can be used to protect database data.

One example of this would be protected healthcare information, or PHI; and 100% of the PHI can be encrypted smart data, or some fraction such as (a seed) can be smart data. This data can contain logic that, if stolen, would alert a predefined person or entity that it was in fact stolen. (This functionality assumes sufficient NVM coverage and that the data passed through one of the NVMs [home wi-fi router] to recognize and process the Smart Data.)

For example:

PHI from a healthcare facility is stolen and transported to an overseas location.

Smart Data determines that it is not at its whitelisted location and spawns an alert Nuance that notifies the healthcare facility and the appropriate authority that the data identified by some serial number has been stolen.

The Smart Data spawns an alert Nuance, which notifies the healthcare facility and the appropriate authority that the data has been stolen.

## 7.5 Unified Data Exchange

In order to create a market for the exchange of information, Thought will assign a monetary value to data: a low monetary value to low value data, and a high monetary value to high value data using the below scale:

**Nuance value scale:**

• Low level data = $.01 per data element,

• Processed sorted data = $.05 per data element,

• Multi-data stream correlated data = $.07 per data element,

• Analytics results = $.26 per result

      Predictive insight = $2.24 per result

**Thought purchases Low level data sets:**

For example:

• Real time seismic sensor stream 10 locations abbreviated by (RTSS10),

• Sensor data from 4 weather sensors abbreviated by (RTW4),

• Historical peak seismic activity data pool abbreviated by (HPSAP)


**Thought then processes these data streams.**

For example:

• Sort and process real time streams, Sort(RTSS10) that output is $.05 per data element

• Sort and process real time streams, Sort(RTW4) that output is $.05 per data element

• Correlate seismic sensor streams with weather sensors
  Corr(Sort(RTSS10)&Sort(RTW4)) output is $.07 per data element

• Analytics of RT streams with historical data $.26 per element


• Predict earthquake with relative precision $2.24 per data element result

• Thought creates a buy and sell market for predictive Nuances:

  ° Demand for this information in Kansas predictive value drops
    to $.28 since there are rarely any earthquakes

  ° Demand for the predictive information in California rises to $125.23, after
    quake of 2.3 on Richter scale, predictive price goes up to $542.54

**NUANCE AS DATA TRANSLATION GATEWAY**

Nuances can convert data in one format to data in another format. One example would be converting two different Electronic medical record formats (below). Data field "Name" and "Surname" in EMR1 for example is converted by data agent "Name Converter" logic to "Given Name", "Middle Name", "Surname", and "Suffix" and vice versa.

FIGURE 18: NUANCE AS A GATEWAY
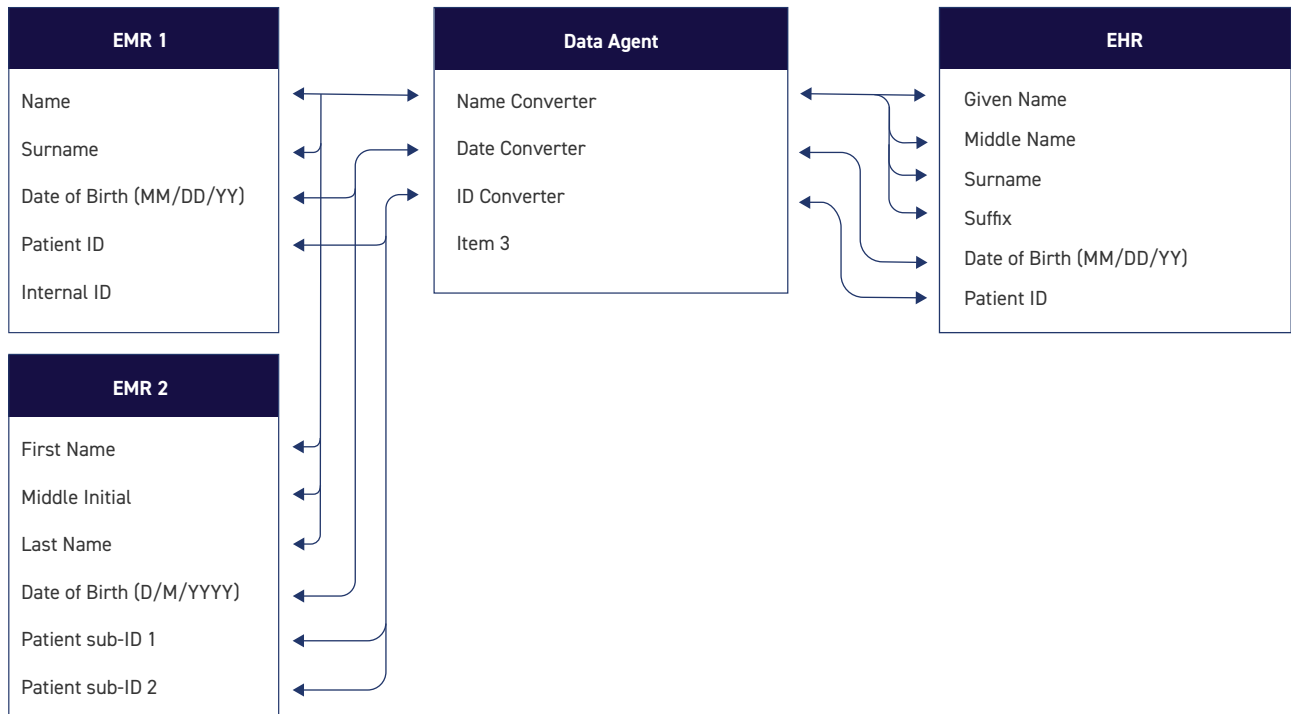
## 7.6 Smart City/IoT

The realization of Smart Cities, smart buildings, and computing at the edge with Internet of Things sensors and infrastructure requires a new level of integration of data and connectivity.

First and foremost, it takes the understanding that one organization alone will not be able to solve all the city's problems or build a smart city, making the city a better place requires an ecosystem. Therefore, a smart city will convene all the relevant stakeholders:

- The private sector, including corporations, small and medium-sized businesses, and startups;

- The public sector, including government agencies, NGOs and other civic partners (like neighborhood groups);

- The citizens themselves;

- Creating new ecosystems—bringing a whole network of companies and sectors together, inspired by a common initiative, to make the lives of the citizens easier by making a city truly "smart."
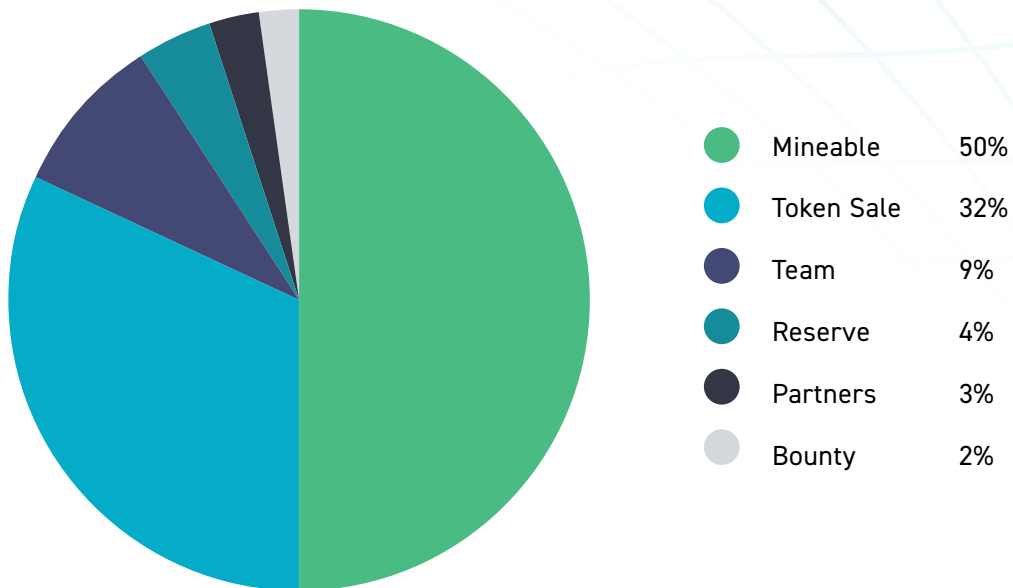
The Thought network can enable intra-industry communication and data exchange, as well as enable granular insight of large real-time data streams and historical data sets. Nuance data flows and processes can form an end-to end continuity of value. The Thought network's universal data exchange will enable new revenue streams, cost savings, and bring quality of life improvements to city residents, administrators, and governments.

### NUANCES SIMULATING BIOLOGY

Nuances can simulate biological functions such as cell activities, e.g. resource transport, resource creation, localized and systemic signaling, and immune system responses. Immune system Nuances could act as malfeasance detection functions by finding rogue or malfeasant Nuances based on detection logic and or algorithms and either alerting on presence or absorption or deletion of malfeasant Nuances.

# 8  Token Allocation

| | | |
|---|---|---|
| 🟢 | Mineable | 50% |
| 🔵 | Token Sale | 32% |
| 🟣 | Team | 9% |
| 🔷 | Reserve | 4% |
| ⚫ | Partners | 3% |
| ⚪ | Bounty | 2% |

We will create up to 809 million out of 1.618 billion Thought tokens using a custom Bitcoin version 0.15-based blockchain. The remaining 809M Thought tokens will be minable over 25 years.

- Total 1.618 Billion Thought tokens

- Allocation of 1.618 total possible tokens:

- 50% Minable over 25 years

- 32% Token Sale

- 9% Team (current and future) advisors and early contributors (10% immediately vested, remaining 90% vested over 1 year)

- 4% Reserve

- 3% Partnerships

- 2% Bounty

**Thought**

**Disclaimer:** Thought tokens that are sold before the Token Purchase Period are non-refundable, even in the event that the token sale does not raise the minimum amount.

The Thought tokens carry no ownership, revenue or governance rights: In particular, participant understands and accepts that Thought tokens do not represent or constitute any ownership right or stake, share or security or equivalent rights nor any right to receive future revenues, shares or any other form of participation or governance right in or relating to Nuance.

As noted elsewhere in this White Paper, the Thought tokens are not being structured or sold as securities or any other form of investment product. Accordingly, none of the information presented in this White Paper is intended to form the basis for any investment decision, and no specific recommendations are intended. Company expressly disclaims any and all responsibility for any direct or consequential loss or damage of any kind whatsoever arising directly or indirectly from: (i) reliance on any information contained in this White Paper, (ii) any error, omission or inaccuracy in any such information or (iii) any action resulting from such information.
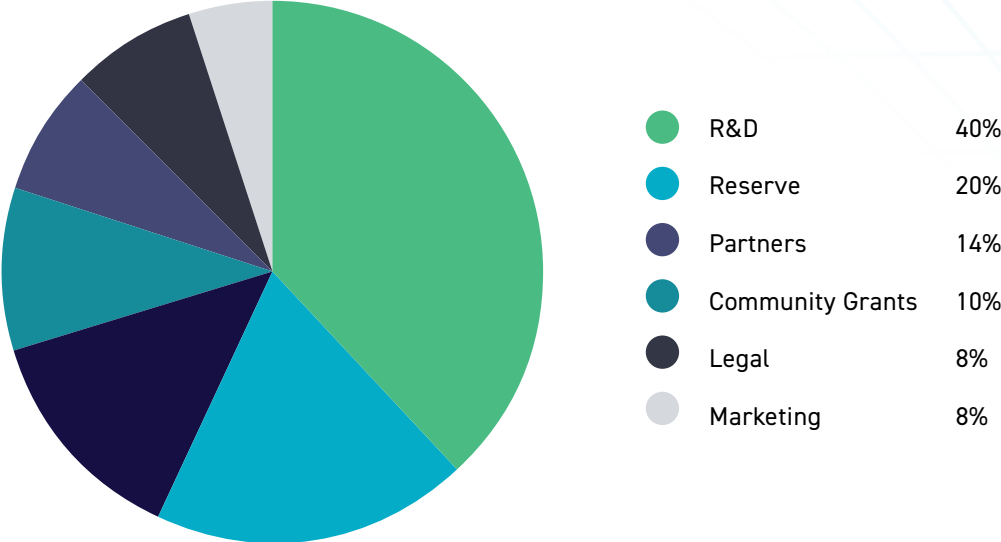
**By purchasing, holding and using THT Tokens, you expressly acknowledge and assume the following risks:**

A private key, or a combination of private keys, is necessary to control and dispose of THT Tokens stored in your digital wallet or vault. Accordingly, loss of requisite private key(s) associated with your digital wallet or vault storing THT Tokens will result in loss of such THT Tokens. Moreover, any third party that gains access to such private key(s), including by gaining access to login credentials of a digital wallet or vault service you use, may be able to misappropriate your THT Tokens. Any errors or malfunctions caused by or otherwise related to the digital wallet or vault you choose to receive and store THT Tokens, including your own failure to properly maintain or use such digital wallet or vault, may also result in the loss of your THT Tokens. Additionally, your failure to follow precisely the procedures set forth in this White Paper for buying and receiving THT Tokens, including, for instance, if you provide the wrong address for receiving THT Tokens, may result in the loss of your THT Tokens.

# 9  Utilization of Funds

FIGURE 20: UTILIZATION OF FUNDS



| | | |
|---|---|---|
| ● | R&D | 40% |
| ● | Reserve | 20% |
| ● | Partners | 14% |
| ● | Community Grants | 10% |
| ● | Legal | 8% |
| ● | Marketing | 8% |

# **10** **Corporate Governance Model**

**THOUGHT CONSISTS OF THREE ENTITIES:**

- Non-profit Global Foundation: Thought Network Ltd., BVI

- For-profit R&D Organization: Thought Network LLC, USA

- Thought Ecosystem - developer network, data scientist,
  artificial intelligence algorithm developers

## Governance

Governance mechanism will be based on data value and data insight value markets.

**GOVERNANCE RESPONSIBILITIES WILL INCLUDE:**

1. checks and balances through a board which provides project oversight
   with a view to token holders' and the Thought network's best interests;

2. structures that align developer and token holder incentives;

3. reasonable token holder contractual rights; and

4. clear and fair disclosure protocols.

5. token holder voting rights

6. limits on compensation

7. system of checks and balances through a board with oversight over management

8. anti-dilution protection: increase in the number of tokens
   outstanding or migration to another token

9.  binding contractual commitment to use best efforts to create the network for which the tokens are being created

10. contractual limits on how raised funds are to be expended

11. fair token allocation procedures

12. transparency with respect to large ownership concentrations and associated susceptibility to price manipulation

13. reporting or audit mechanism

## Disclaimer

The Thought network is a product, NOT a security or investment offering. THT is a token required for paying transaction fees or building or purchasing decentralized application services on the Thought network; it does not give you voting rights over anything, and no guarantees are made for its future value.
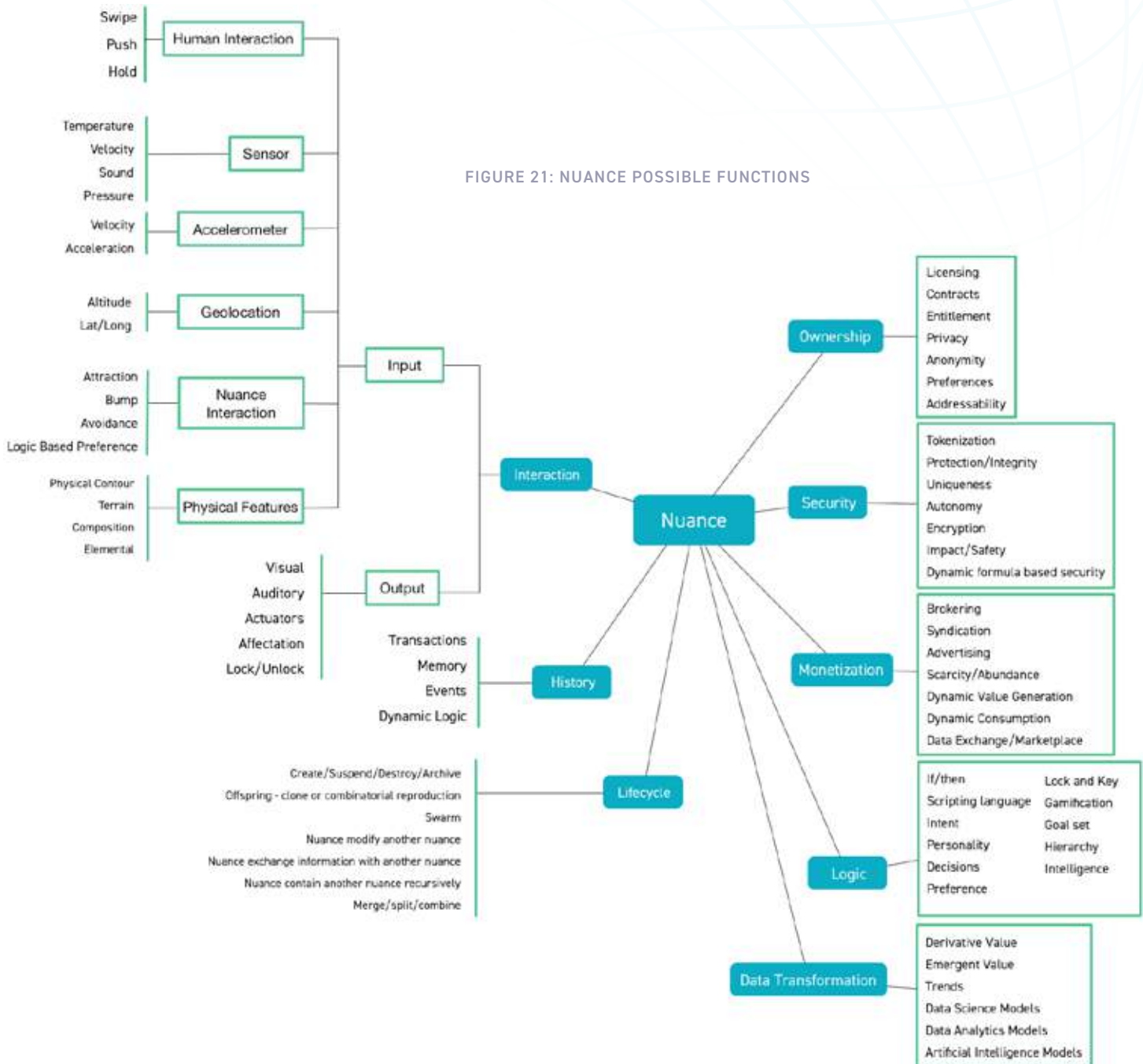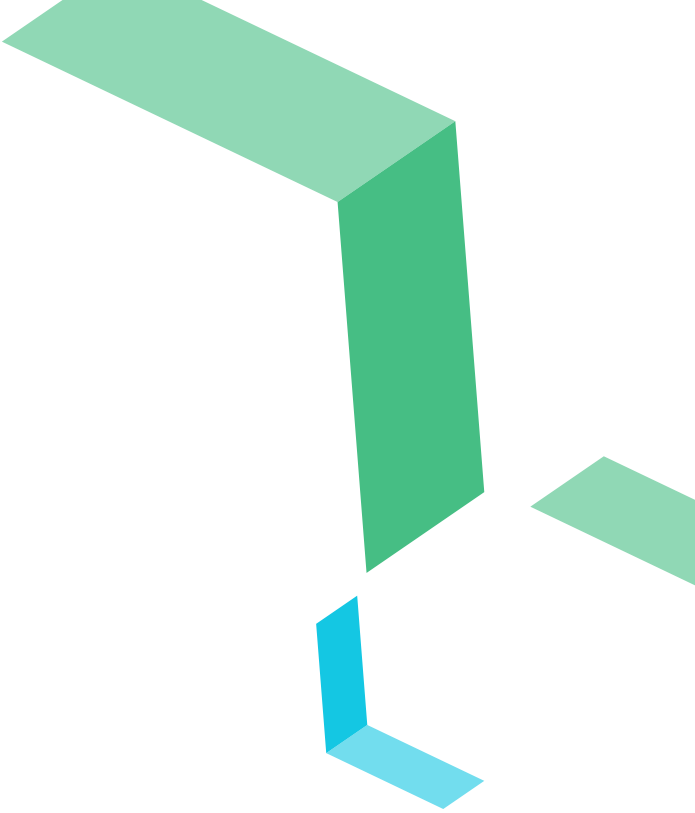
# 11 Sources

1. https://techcrunch.com/2017/11/19/the-battle-for-control-of-data-could-be-just-starting

2. http://usblogs.pwc.com/emerging-technology/top-10-ai-tech-trends-for-2018/

3. https://www.informationweek.com/big-data/big-data-analytics-market-to-hit-$203-billion-in-2020-/d/d-id/1327092

4. https://www.seagate.com/files/www-content/our-story/trends/files/Seagate-WP-DataAge2025-March-2017.pdf

5. https://www.researchgate.net/figure/44849231_fig10_Hierarchical-swarm-optimization-model-using-a-multiagent-system-in-nested-hierarchies

6. Brent Owens, https://gamedevelopment.tutsplus.com/tutorials/goal-oriented-action-planning-for-a-smarter-ai--cms-20793)

7. James Henderson http://cui.unige.ch/~hendersj/papers/henderson_iwpt11.pdf

8. Ray Kurzweil, How to Create A Mind

9. James Henderson http://cui.unige.ch/~hendersj/papers/henderson_iwpt11.pdf

# 12   Appendix

This graphic describes the various types of Nuances that might exist, their functions or uses and benefits.



FIGURE 21: NUANCE POSSIBLE FUNCTIONS

**Thought**