# *Introduction to Data Economics* Chapter 5:

## Introducing the Lydion DEOS and its Functions

Authors:
Arka Ray, BS
Jennifer Hinkel, MSc
Sirtaj Singh Brar, BS

Editor:
Ben Bagamery

## Table Of Contents



 

## Introduction to Chapter 5: Introducing the Lydion DEOS and its Functions

While we are paying for things using digital data all the time within "everyday" Data Economic Frames as described in Chapter 4 of *Introduction to Data Economics* above, *none of the data being used to create the digital Data Assets we pay with within these Data Economies actually belongs to us, even when we generate it* (see the example of Anonymized Location Data in **Section 2.2 of Chapter 4** and in **Section 4.4 of Chapter 3: Types and Examples of Data Economies.**

This legacy structure of "everyday" Data Economies presents an opportunity for envisioning a different type of Data Economy and corresponding methodology, and that is where the Lydion Data Economic Methodology for Data Assetization and the Lydion Data Economic Operating System (or Lydion DEOS) comes in.

**The primary goal of the Lydion DEOS is to allow the design and implementation of Data Assets (and the Data Economic Frames they live within) *using any combination of datasets and data streams owned by the Participants in the Data Economy.***

**(Coming Soon) Diagram 1.1: "Lydion Methodology - Lydion DEOS Manufactures different Data Asset Types from 'Completed Tasks'"**

**In other words, Lydion Data Assets are created using datasets that we—companies and people—generate and own.**

**The Lydion DEOS implements the Lydion Assetization Methodology:** *a generalized framework for turning any combination of digital datasets and data streams into one or more Types or Classes of Data Assets with the properties of Identity, Context, Utility, and Value within a Data Economic Frame of Reference.*

Chapter 5 of *Introduction to Data Economics* picks up where Chapter 4: Deeper Dive into Fundamental Data Economic Concepts left off. It explores the basic design and functionality of the Lydion DEOS (Data Economic Operating System) and how the Lydion DEOS enables

implementation of the Lydion Data Economic Methodology to create Lydion Data Assets from any combination of datasets and datastreams.

## 1. Introduction to the Lydion DEOS and Lydion Methodology of Manufacturing Data Assets From Digital Datasets

### 1.1 The Lydion Methodology for Data Assetization—How the Lydion DEOS Turns Raw Digital Datasets / Data streams into Data Assets with Utility and Value

**In Sections 3.5 and 3.6 of Chapter 4, we examined a General Methodology for creating Data Assets, and concluded that to create a Data Asset from a given dataset,** the following are needed:

1. A raw Data Quantum (a package of data formed—or quantized—from the underlying dataset)

2. Context-Function (or Generation-Function)

3. Utility-Function

4. Evaluation-Function

The Utility and Evaluation Functions are themselves dependent on the Context-Function, and the Context Function takes as input the Raw Data Quantum (or Generation Context).

Therefore, to convert a given dataset into a type of Data Asset, the Lydion DEOS needs a general framework and related methodology to:

1. **Group (or quantize) the dataset into Raw Data Quanta,** each of which can form a Data Asset.

2. **Define the Context-Function—or Generation-Function**—that operates on the Raw Data Quanta thus created to turn them into Data Assets.

This Context-Function/Generation-Function does the heavy lifting of quantizing (or packaging) the underlying dataset per the requirements of the rules encoded in the Generation-Function on how to package up the underlying dataset into Raw Data Quanta that the function can then operate on to form Data assets.

3. **Define Utility and Evaluation Functions** following from the Context-Function/Generation-Function.

**To put it more simply, the Lydion DEOS needs to prescribe a general method to define the Context (or Context-Function)** that expresses the meaning of raw data quantum from a raw dataset and turns it to a Data Asset. This defined Context should then enable the expression of Utility and Value for the Data Asset (and the underlying Raw Data Quantum that forms its Generation Context).

**The general method that the Lydion DEOS uses to define the Context (meaning or reason for existence) of datasets to convert them into Assets is quite simple, and is based on the following insight:**

If we examine the Context of all the real-world Data Assets we have discussed so far, we see that there is a pattern: ***The Data Asset is always itself the Outcome or result of Work (or Tasks) requiring a provable, non-trivial amount of energy being spent in the real world.***

Put another way—***each Data Asset contains data expressing Outcomes or "proof of work" performed—or energy expended—in the real world as its Context (more specifically its Generation Context)—or as its reason for existing.***

**(Coming Soon) Diagram 1.2: "Show Lydion Data Asset getting "minted" due to "proof of work" in the real world"**

**This leads us to the insight that is at the heart of the Lydion methodology: *The very existence of a Lydion Data Asset is the result of work performed (or more specifically tasks completed) in the real world.***

1. This work generated valuable results (or Outcomes) that can be captured and expressed through digital data, and which are used to produce (or manufacture) an instance of a Data Asset

2. Conversely, if such work had not been completed, that instance of the Data Asset would not exist.

As a result, **the Lydion Data Asset can *represent a proxy* for these real-world Outcomes** and the Completed Tasks that led to its creation, including:

1. The expenditure of energy required for completing these tasks and generating the Outcome, and hence

2. The scarcity of Outcomes that these tasks generated.

**(Coming Soon) Diagram 1.3: "Components of the Lydion DEOS"**

**The Lydion DEOS generalizes this concept into a framework in which digital datasets and data streams get packaged up into "logical blocks" (or "data quanta") that represent proof (measurement) of results—or "Outcomes"—created by some kind of work performed.**

1. Each of these logical blocks is an instance of a type (or class) of Lydion Data Asset.

2. We have seen that for any given dataset (and the desired specifications of the Data Assets that need to be created from that dataset) the Lydion DEOS needs to:
    a. … **define one or more Context-Functions (or Generation-Functions) that express the meaning of the underlying dataset** by packaging it up into discrete and uniquely identifiable data quanta in a way that can then …

   b. … lead to definitions of Utility-Functions and Evaluation-Functions expressing the Utility and related Value of the underlying dataset.

3. The general method that the Lydion DEOS therefore uses to design such a Context-Function **is to define the rules for quantizing the underlying dataset into a set of blocks "or data quanta" where each Raw Data Quantum packages up data representing the Outcomes of a set of Completed Tasks.**

**(Coming Soon) Diagram 1.4: " Data Asset -> Data Quantum with layers / functions - Context via Completed Tasks + Outcomes -> Leading to Utility + Value - turning it into a Data Asset via DEOS"**

4. **As a result of the Context-Function, the data contained within a Lydion Data Asset—its Generation Context—represents the "Outcomes" of a set of "Completed Tasks,"** where each "Task" required the expenditure of a provable amount of energy to "Complete".
   a. The data within the Data Asset therefore contains "Proof of Work" that has been completed in the real world and that has required some amount of energy, modeled as a set of "Completed Task Transactions".
   b. This is called "Transactional Proof of Work".

5. **The Utility and Value of such a Data Asset—the new Outcomes it can create—can then be based on the Completed Tasks and their Outcomes that are contained within the Data Asset** and represent its Context (and the Proof of Work that led to the Asset's existence).

**As a result, Utility-Functions and Evaluation-Functions for the Data Asset can be created based on the Context-Function and the potential utility and value of the data representing Outcomes of Completed Tasks** that Context-Function packages into the Data Asset.

6. **As discussed in Section 3.5 of Chapter 4, the Context-Function injects metadata into the Raw Data Quantum that includes a unique identifier for the Data Asset being**

**created** as well as other states such as "Ownership".

**(Coming Soon) Diagram 1.5: "Data Asset Utilization via State Changes + Additions to Dynamic Context"**

7. **The Utilization of a Data Asset is represented by:**
    a. **Changes in the metadata or state injected into it by the Context Function.** For example, utilization of an Asset by transferring it to a new owner can be represented by the change of "Ownership" state from the previous owner to the new owner.
    b. **The record of each such Utilization is itself a Completed Task.** Each such Utilization Completed Task is added to the Data Asset's Context—or the data quantum that it contains.
    c. **As a result, a Data Asset contains the entire history of its utilization** (Utilization or Dynamic Context) in addition to the history of the reason for its creation (Generation Context).

See **Chapter 2: Fundamental Concepts** for another high-level explanation on the Lydion Methodology for generating Data Assets, in particular the concept of the "canonical copy" of a Data Asset and its ability to represent (or proxy) real-world outcomes.

See **Chapter 3: Types and Examples of Data Economies** for a deeper exploration of the "Atomic Assumptions of the Lydion Data Economic Methodology" that explores these concepts within a formal framework.

**(Coming Soon) Diagram 1.6: "Data turning into Location Data Asset via DEOS - Addition of Context, Identity, Utility - Show Node etc."**

As a simple example of a Lydion Data Asset, consider a type of Data Asset that packages up a user's anonymous location data as described in one of our earlier examples.

1. **The Context of such an Anonymized Location Data Asset** contains the data representing the results of the user's Completed Tasks, which in this case involved the user traveling from one location to another. The Outcomes, or results, of these "Tasks" are the GPS coordinates (and other location-related data) that the user generates by traveling.

2. **The Utility of such a Location Data Asset** can then be based on the Proof of Work (the user traveling around) represented by its Context, for example, earning the user credit for letting a Telecom company use the data.

3. **The Value of such a Location Data Asset** varies based on its utility and the form that the value is expressed in.

For example, 1 GPS data point may be needed to produce a Location Data Asset which can then be used to gain 1 hour of cell phone service by the user. The Value of such a Data Asset can be expressed in at least 2 ways (and possibly many more): 1) 1 Hour of Cell Phone Service or 2) $X—the Cost of Buying 1 Hour of Cell Phone Service.

**(Coming Soon) Diagram 1.7: "Anonymized Location Data Asset Table"**

| **Data Asset Type** | **Participants in Data Economic Frame** | **Context -** *The Story of Identity, Scarcity, and History*<br><br>**("I express that…"** | **Utilization Process - (or** Utilization-Function)<br><br>**("When put through this process…")** | **Utility and Value -** *The Story of Utility and Story of Value*<br><br>**("Hence I can be used to…"**<br>**("Hence my value is…"** |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| **Anonymized Location Data Asset** | Cell Phone User, Telecom Company, Location Data Licensor / Buyer | I mean that the manufacturer of this Data Asset has traveled to Location X at time T | When ownership transferred, permanently or temporarily from "Cell Phone User Participant" to "Telecom Participant" | Hence can be utilized by the "Cell Phone User Participant" to<br>1. Gain access to an amount of money X<br>OR<br>2. Gain access to Y hours of cell phone service<br><br>*New Outcomes = Access to Money or Cell Phone Service*<br>*(due to Utilization Process mentioned)*<br><br>*Value = $X (amount of Money as Credit) or*<br>*Value = $Y (Price of Y hours of Cell Phone Service)*<br>*(calculated by an Evaluation Function: Cell Phone Service -> $)* |

### 1.1.1 Understanding Data Assets Through the Analogy of a Rechargeable Battery

To gain further intuitive understanding of a Lydion Data Asset, it can be thought of as **a digital rechargeable battery.**

- A battery is charged using electrical energy, which it stores as chemical energy.
- Similarly, a Data Asset is charged by (or made up of) valuable results / outcomes / performance, stored as data representing these outcomes (and the work / effort required to achieve these results.)
- Just as the chemical energy stored in a battery can be used to do new work, the value stored in a Data Asset in the form of data can be used—or utilized—to create interesting new outcomes and incentives.

In this analogy, the Lydion DEOS can be thought of as the infrastructure enabling the creation of an electrical grid connecting all participants in the Data Economic Frame.
Within this grid, Participants can plug in the "Data Asset Battery" to charge it (with data from Outcomes of relevant Completed Tasks) and utilize it to generate new Outcomes.

Such a grid connecting Participants of a Data Economic Frame is called a "**Data Economic Network,**" which will be explored in more detail in **Section 2 of this chapter.**

### 1.2 The Primary Features of a Lydion Data Asset

**(Coming Soon) Diagram 1.8: "Cross Section of a Data Asset Showing Generation and Dynamic Context"**

**A Lydion Data Asset, created by the Lydion DEOS is a package of data—or Data Quantum—comprising two types of data:**

1. **Context, which is the data representing the "Outcomes" of "Completed Tasks"** that were required to generate the Asset. This is further divided into Generation and Utility Context.
    a. **Generation Context:** Completed Tasks that led to the creation of the Data Asset (that is, the Asset instance's "Transactional Proof of Work".)
    b. **Dynamic (or Utility) Context:** Completed Tasks in which this Data Asset has been utilized for in the past.

2. **Rules (as Programs)—that are expressed as modular digital programs (each called a Machine, together a Data Asset Factory)** for generating, utilizing, and evaluating this type (or class) of Data Asset.

**(Coming Soon) Diagram 1.9: "Visualization of Data Asset (w/ Gen / Dyn Context) answering questions through the Stories"**

**The Context and Rules (Programs called Machines) within it allow the Lydion Data Asset to answer the following questions** to any Participant in the Data Economic Frame within which the Data Asset lives:

1. **Why were you created? Who are you (how can you be identified)?**
2. **What have you been utilized for in the past?**
3. **What can you be used for in the future?**
4. **How can you prove the above answers?**

The answers to these questions give the Data Economic Participant the information they need to make decisions around the value and utility of a specific type and/or instance of that Data Asset.

**Further, being able to answer these questions also enables the Lydion Data Asset to tell the following Stories** that we have seen in **Section 3.6 of Chapter 4** above as well as in **Section 3 of Chapter 2: Fundamental Concepts** that are required for any utilizable product and that form the basis of the Asset's meaning (context), utility, and value:

1. **Story of Generation / Identity (and Scarcity):** This story answers the questions
    - *"Who are you (how can you be identified?)"*
    - *"Why were you made?"*

Using the Completed Tasks in the Data Asset's Generation Context, the Data Asset's Context-Function can answer exactly what real-world Tasks had to be completed in order for it to be generated.
The Generation Context therefore also provides the answer to how scarce the Data Asset is—that is, how much work (energy expenditure) was required to generate it.

2. **Story of Utility:** This story answers the question

- *"What have you been utilized for in the past?"*

Using the Completed Tasks in the Data Asset's Dynamic (or Utility) Context, it can recall and communicate all the transactions in which this specific Data Asset instance has been utilized.

3. **Story of Value (or Future Utility):** This story answers the question
    - *"What can you be used for in the future?"*

Using the Rules (embedded in the Data Asset Factory programs), the Data Asset can communicate the types of transactions (also Completed Tasks) it can be used for in the future and the subsequent new outcomes that can be generated as a result of these transactions. Hence, the Data Asset relays why it should be considered valuable.

These Stories, however, do not address how the Data Asset can answer the fourth question above—***How can you (the Data Asset) provide proof for the other three answers?***

This question leads us to another adjacent question which needs to be answered first: **To whom do these answers need to be proven?**

**To answer both these questions, we need to introduce the concept of a *"Data Asset Market".***

**1.3 Introduction to Lydion Data Asset Markets, Lydion Data Economies and DENETS**

**Several types of Data Assets either get:**

a. **Generated by a combination of entities** contributing some combination of data (**Producers or Manufacturers of the Asset Type**).

b. **Utilized by a combination of entities** who may or may not be involved in generating the Data Asset (**Consumers of the Asset Type**).

Each Data Asset therefore has to not only carry context data and rules that can be used a) to construct units (or instances) of that Data Asset, and b) to utilize that instance of that class of Data Asset to create new outcomes, but each Data Asset also has to have a mechanism to communicate this data and rules to other entities who may want to utilize (or value) or even help generate new instances of the Asset type.

**These people and organizations who are involved in generating and/or utilizing a Class (or Type) of Data Asset are therefore Participants in a market that exists within their Data Economic Frame, called a *Data Asset Market.***

1. Each Data Asset Market determines and enforces the rules that enable the generation of Data Assets and their utilization among its **Participants** (entities: people, organizations, computers or other machines) as well as the management of every single instance of that type of Data Asset.

2. These Participants are called the Data Asset's "**Trust Domain**".

3. The rules governing a Data Asset Market are determined and agreed upon by a subset of its Participants, called the Market's "**Consensus Domain**" (a subset of the Market's Trust Domain).

**Such a Data Asset Market is responsible for managing (generating and controlling) every single instance of that Class (or type) of Asset.**

Further, each Data Asset Market is responsible for only one Class (or type) of Data Asset. For exploration on the reasoning for this and related choices is explored in more detail in **Chapter 3: Types and Examples of Data Economies**.

**(Coming Soon) Diagram 1.10: "Data Asset Market Component Diagram"**

**A Lydion Data Asset Market (DAM) comprises the following:**

1. **Trust Domain Participants:** Set of Participants who are responsible for generating and utilizing Data Assets from that Market. The Trust Domain contains both the Producers and Consumers of that Data Asset type.

2. **Consensus Domain Participants:** Set of Participants who are responsible for creating and enforcing the generation and utilization rules governing the Data Asset Market.

3. **Data Asset Factory:** A set of modular digital programs called Machines that encode and enforce the rules of the Data Asset Market, including the generation, storage and utilization of all instances of that class / type of Data Asset.

4. **Data Asset Vault:** Where the Data Assets generated and controlled by this Asset Market (including all associated metadata) are stored (as digital data, typically in a database).

5. **Data Sources:** Datasets, datastreams and other Data Asset Markets that are used to model Outcomes / Completed Tasks and then used to mint the Data Asset type managed by the Data Asset Market.

**(Coming Soon) Diagram 1.11: "Multiple DAMs - Forming a DE Frame"**

**A Lydion Data Economic Frame of Reference—or simply a Lydion Data Economy—is just a set of Data Asset Markets,** each managing a single type/class of Data Asset, that each have the same set of Participants in their Trust Domains. The set of Data Asset Markets in a Data Economy typically work together towards one or more common goals.

**A Lydion Data Economy (Data Economic Frame) therefore comprises:**

1. **A set of Lydion Data Asset Markets (DAMs)** that each manage a Data Asset type.

2. **Participants** in the Data Economic Frame who are part of each DAM's Trust Domain.

**The Lydion DEOS implements Lydion Data Economic Frames through DENETs—Data Economic Networks—that comprise Data Economic Applications (DE-APPs) running on Data Economic Nodes (DE-Nodes or DENs).**

Such DE Nodes, running instances of the same DE-APP, connect with each other to form a ***Data Economic Network (or DENET).***

**The DENET can be thought of as the real-world representation of a Data Economic Frame of Reference,** (much as Data Assets act as a proxy for the utility and value of the data they contain).

The following **Section 2** examines, at a high-level, the functions that the Lydion DEOS performs in implementing Data Economic Networks and the DE-APPs and DE-Nodes that comprise them.

## 2. Primary Functions of the Lydion DEOS

### 2.1 Summary of the Functions of the Lydion DEOS

**The Data Asset Markets within a Data Economy work together to enable a specific set of solutions for their Participants.**
As a result, a Data Economic Application (DE-APP) typically implements and manages not just individual DAMs, but the set of Data Asset Markets (and their corresponding Data Asset types) that together form a Data Economy or a Data Economic Frame of Reference.

**DE-APPs can thus be used to develop and implement Data Economic Solutions of various kinds using the Lydion DEOS, where each Solution comprises:**

- One or more Data Economic Frames of Reference, each with its own set of Data Asset Markets that can communicate among one another.

- DE-APPs running on Data Economic Nodes forming one or more Data Economic Networks among Participants.

DE-APPs enable different types of features and functionality for the Participants by enabling them to assetize and utilize the digital datasets of interest to the Participants towards a set of common goals for the Participants (related to commerce or any other activity of interest to them).

**(Coming Soon) Diagram 2.1: "DE-APP producing DE Solution producing DENET"**

**The Lydion DEOS provides the libraries, tools, and related API to implement Data Economic Solutions and specific implementations of Data Economic Solutions as (one or more) Data Economic Networks (or DENETs)**

1. The DE Solution takes the form of a Data Economic Application (DE-APP). A DE-APP is a distributed software application running on Data Economic Nodes (computers, where each Node represents a Participant in the Data Economy).

2. Each instance of a DE-APP runs on a Data Economic Node which represents a Participant in the Data Economic Frame being implemented by the DE-APP.

3. These Nodes together form a Data Economic Network (DENET), which manages all aspects of this specific Data Economy (or DE Frame of Reference).

4. Each of these Data Economic Nodes connects to and is able to communicate with every other Node in the Data Economy running the specific DE-APP.

5. **An implementation of a DE Solution is a specific DENET (or set of DENETs) among a specific set of Participants.** A Solution can have as many different implementations as desired among different sets of participants dealing with different datasets.

6. Each implementation of a Data Economic Solution creates a Data Economy (or Data Economic Frame of Reference) among specific Participants.

**The Lydion DEOS can be used to create two different but related types of DE-APPs:**

1) **Data Economic Solution Platforms:** Larger-scale DE-APPs that implement common sets of features and functionality required by specific types of DE Solutions in specific spheres of industry or other activity.

2) **Standalone Data Economic Application:** Specific implementations of a DE Solution for a set of Participants, typically built using an underlying DE Solution Platform.

The classes of problems / opportunities that Data Economic Solutions can effectively address, along with specific examples of real-world Data Economic Solutions will be laid out in a future Chapter.

**(Coming Soon) Diagram 2.2: "Text with the functions of the DEOS"**

**The Lydion DEOS plays three primary roles in implementing Data Economic Frames of Reference via DENETs** (Data Economic Networks), formed by DE Nodes (Data Economic Nodes) running instances of the same DE-APP:

1. **The Process of "Data Assetization" to form different Data Asset Types via pre-designed *Data Asset Markets,*** implemented by a Data Economic Application (DE-APP) built on the Lydion DEOS.

2. **Formation of DENETs (or *Data Economic Networks)* to** Enable Communication Among Participants of Data Asset Markets within the same Lydion Data Economy.

3. **Communication Among Different Data Economic Frames via *Shared* Data Economic Networks.**

The following sections examine these functions in some detail.

### 2.2 The Process of "Data Assetization" via Data Asset Markets (DAM)

Lydion DEOS helps design Data Asset Markets (DAMs) that can generate different types of Data Assets from the same underlying data. The Data Asset Markets are implemented via (one or more) Data Economic Applications or DE-APPs, which are distributed software applications built using the Lydion DEOS.

**(Coming Soon) Diagram 2.3: "Manufacturing Diagram for a Data Asset from Raw Outcomes Data via DEOS"**

The Lydion DEOS implements a Data Asset Market (within a DE-APP) through a fairly standard process, regardless of the specific type or functionality of the specific Data Asset Market being implemented:

1. Each Data Asset Market takes a specific combination of data from underlying datasets and models them as Outcomes (or results) of Completed Tasks.

2. These Outcomes and associated Tasks make up a Data Quantum—a finite set of 1s and 0s that are packaged up in a "block" or "quantum" of data referred to as the "Raw Data Quantum" forming the Data Asset.

3. The Data Asset Market then associates digital programs (called Data Asset Market Machines or simply "Machines", together forming the "Data Asset Factory") with the Data Quantum that allows any Participant in the Data Market to understand what the underlying Completed Tasks and Outcomes mean, as well as the valid methods for utilizing these underlying Outcomes to create new Outcomes.

4. If Participants don't "natively" understand the value of the Data Asset, the Data Asset Market also provides methods, in the form of digital programs, to evaluate the Data Asset to a value that the Participant does understand (currency, goods/services, even

other Data Assets).

5. Finally, the Data Asset Market adds metadata to the Data Quantum that allows any Participant in the Data Asset Market (in the Market's Trust Domain) to uniquely identify the Data Quantum, thus turning it into a Data Asset.

6. Therefore, through the Data Asset Machines, the Lydion DEOS implements the key Functions required to convert the underlying Raw Data Quantum—representing the Outcomes of Completed Tasks—into a Data Asset, namely:
    1. **The Context-Function** (**or Generation-Function**) that injects identifiers and other metadata required to uniquely identify and transact the Data Asset,
    2. **The Utility-Function** that defines the Utilization Process that the Data Asset can be put through—by Participants in the DAM and the Data Economic Frame the DAM exists within—to generate New Outcomes, and
    3. **The Evaluation-Functions** that can be used by the Participants of the Data Economic Frame to measure the Data Asset's utility through a scalar or vector value.

As described in **Sections 1.3 and 2 of Chapter 4,** an underlying dataset can be used to produce or manufacture many different Types or "Classes" of Data Assets. The above process describes one Data Asset Market being created from the underlying dataset.

**(Coming Soon) Diagram 2.4: "Show Data Quantum getting Context + Utility to Turn into Multiple Data Asset types"**

**The same process can be extended to design and implement additional Data Asset Markets within the Data Economy, using the same underlying datasets:**

1. Each combination the following, when defined on an underlying dataset, forms a new Type (or Class) of Data Asset:
    a. Context (or Proof of Work) containing Outcomes of Completed Tasks.

   b. Associated Utility (and Value) due to the Context.

2. To form a new type of Data Asset from the underlying Data Asset, a new Data Market can therefore be defined, containing a fresh set comprising:
   a. A Context-Function (or Generation Function)
   b. A Utility-Function (one or more)
   c. One or more Evaluation-Functions

Each time a new set of Context-Function, Utility-Function, and Evaluation-Functions can be defined through Data Asset Machines, a new Data Market can be formed from the underlying dataset that generates and manages a new Data Asset Type.

For example, to derive Data Asset Types A and B from underlying Dataset D:

- **Data Asset Type A** = Utility-Function-A (Context-Function-A(Raw Data Quantum from Underlying Dataset D)

- **Data Asset Type B** = Utility-Function-B (Context-Function-B(Raw Data Quantum from Underlying Dataset D)

3. Further, multiple new Types / Classes of Data Asset can also be generated using the same Context-Function. Technically, it is not strictly required to create a new Context-Function for a new type of Data Asset as long as a fresh Utility Function can be defined using the same Context-Function.

For example, to derive Data Asset Types A1 and A2 from underlying Dataset D:

- **Data Asset Type A1** = Utility-Function-A1 (Context-Function-A(Raw Data Quantum from Underlying Dataset D)

- **Data Asset Type A2** = Utility-Function-A2 (Context-Function-A(Raw Data Quantum from Underlying Dataset D)

As a result, this Data Asset Market enables a generation-utilization loop for each Data Asset type, and that enables the Market's Participants (Trust Domain) to create, evaluate, and utilize each Data Asset instance minted and managed by the DAM.

2.3 Formation of Data Economic Networks (DENETs) Among Participants of Data Asset Markets (DAMs) Through Data Economic Nodes (DENs)

**The Lydion DEOS also enables communication among participants in the Data Asset Market through a Data Economic Network accessed through a *Data Economic Node.***

**Each Participant in a Data Asset Market has a *Data Economic Node***—a computer running the Data Economic Application implementing the DAM—that represents their membership and Participation in the Market.

**The Data Economic Application** (**DE-APP**) implementing the Market is a distributed (or decentralized) application, and a copy (or instance) of the Application runs on each Data Economic Node representing a Participant in the Data Asset Market.

**(Coming Soon) Diagram 2.5: "Closeup of a DEN"**

**Each Data Economic Node, through the DE-APP, maintains a complete copy of the Data Asset Market,** which includes:

1. **The verified set of other Participants in the Data Asset Market** and their corresponding DE Nodes,

2. **A verified copy of the Data Asset Factory and its component "Machines"** (digital programs implementing the rules of the DA Market), and

3. **A copy of the Data Asset Vault, containing every Data Asset of the native Type generated** and managed by the Data Asset Market.

**Each Data Economic Node (DEN) also has direct access to the data sources**—datasets, data streams and other Data Asset Markets required to mint the Data Asset managed by the DAM.

The instances of the Data Economic Application (DE-APP), using their copy of the Data Asset Market, then enable these DE Nodes to:

1. **Communicate / receive Data Assets and translate / understand the Completed Tasks, Outcomes, Utilization and Value of the Data Assets** managed by that Market.

2. **Ensure that the copies of the DA Markets and their components (Factory and Vault) remain synchronized across the DE Nodes of all Participants of the DA Market.** This is how the DA Market ensures that all of its Participants agree, or are "in consensus" on:
    1. The rules governing the generation and utilization of that type of Data Asset,
    2. The specific instances of Data Assets that are part of the DAM, and
    3. All of the Participants who are allowed to join and participate in the DAM.

**(Coming Soon) Diagram 2.6: "Show Data Asset Markets connecting over a DENET"**

**These Data Economic Nodes together form a Data Economic Network (or DENET).**

**A DENET is a closed digital network among the Participants of the Data Economic Frame of Reference (who in turn are Participants in each DA Market within that Data Economy).**

As mentioned in Section 2.1, a DENET can be considered the physical manifestation and representation of the Data Economy (ie. Data Economic Frame of Reference).

**A Data Economic Network (or DENET) hence enables** interactions among:

1. **Participants of the Data Asset Markets** that are part of the Data Economy being implemented by the DENET.

2. **Data Asset Markets themselves,** enabling different types of Data Assets to interact and be exchanged or otherwise transacted for each other. Such interactions are usually key to enabling whatever Data Economic Solution is being implemented by the DENET.

3. **Different Data Economic Networks implementing different Data Economic Frames of Reference** to form Shared Data Economic Frames of Reference (or Shared DENETs.)

### 2.3.1 DENETs enable Ownership and Auditability of Data Assets

**(Coming Soon) Diagram 2.7: "Auditability of Data Assets"**

**Section 3.4 of Chapter 4** explored the properties of a Data Asset product created using datasets as raw materials: Context (including Identity), Utility, Value, Tracking and Auditability.

1. We have discussed in some detail the method that Lydion uses to inject Identity, Context, Utility, and Value rules into a data quantum to turn it into a Data Asset.

2. We have now seen how the remaining features—the external metadata and state that the Data Asset needs to track, such as **Ownership and Auditability**—are enabled when the Data Asset Market becomes part of a Data Economic Network. The features of the DENET described above enable these states to be independently verified by each member of the DENET / Data Economic Frame.

### 2.3.2 DENETs are Closed, Decentralized, Peer-to-Peer (P2P) Digital Networks

In most solutions where datasets are used for analysis by an algorithmic (for eg. "SaaS") service, especially if the datasets and data streams are under the control of different agents / owners, a traditional centralized system is typically set up, requiring data to be sent and analyzed by a centralized service.

- Once the data leaves the "Data Firewall" of a specific Agent, they have no way to track what happens to the data or what it is used for.

- The centralized system is typically authoritative and the single source of truth for all Agents.

This is a traditional centralized client-server based model for data analysis and usage.

**(Coming Soon) Diagram 2.8: "Theoretical Client-server DENET versus P2P / Distributed DENET"**

**A Data Economic Network (DENET) is implemented as a closed, decentralized, peer-to-peer digital network formed by DENs (Data Economic Nodes) connecting with each other,** as opposed to a client-server model where a central service acts as the source of truth and authority for the entire Data Economic Frame.

There are several reasons for DENETs being modeled and implemented as a decentralized peer-to-peer network:

1. Central authorities cannot possibly be scaled to understand / generate / manage every type of DA and their utilities across all participants.

2. Central authorities form central points of failure and attack vulnerability, making a centralized DENET more vulnerable, less antifragile, and less scalable.

3. Data streams / data sources in a peer-to-peer DENET can stay in their native locations / behind firewalls. DE Nodes, enabled with Lydion DEOS running the appropriate DE-APPs, can translate these disparate datasets into a common language for interaction with other nodes in the DE.

Data Economic Networks are also introduced and discussed at a high-level in **Section 4 of** **Chapter 2: Fundamental Concepts** which can augment the material in the current Section.

### 2.4 Enabling Communication Among Data Economic Frames of Reference through their Data Economic Networks (DENETs)

**The Lydion DEOS also provides the frameworks, technology, and tools for different Data Economic Frames—represented by different DENETs—to interact with each other and form Shared Data Economic Networks.**

**(Coming Soon) Diagram 2.9: "DENETs connecting with each other to form Shared DE"**

This can be implemented by allowing different closed DENETs with their own sets of Participants to:

1. **Direct DAM-DAM Interaction:** Interact with each others' Data Asset Markets and understand the meaning, utility, and value of the Data Asset types being generated and utilized by these Markets.

2. **Deeper Interactions among DAMs and their container DE Frames through Shared DENETs (and Shared DE Frames):** Form new, shared DENETs from their existing DENETs that enable the Participants from different Data Economies to come together to form new, shared Data Asset Markets generating new types of Data Assets. This results in further expansion of the features and functionality of their individual DENETs.

The methodologies for Data Asset Markets within different Data Economic Frames (Foreign Data Asset Markets) to communicate with each other are fundamental to the design of practical, real-world Data Economic Solutions and the DENETs that implement them.

### Conclusion to Chapter 5: *Introducing the Lydion DEOS and its Functions*

Following Chapter 5, readers of the *Introduction to Data Economics* series will be able to follow a few different tracks in parallel:

1. Chapters that will build on the philosophical and practical economic implications and impact of Data Economics and DEOS, including putting the science of Data Economics and Lydion DEOS within the context of micro and macro trends in economics, finance, and the expression and transaction of value using digital assets such as blockchain-backed cryptocurrencies that in turn power Utility Tokens, Non-Fungible Tokens (NFTs) and more. These chapters will build off the concepts introduced in Chapters 1, 2, and 4.

2. Chapters that will build on describing details of the architecture of the Lydion DEOS and the methodologies and best practices for building Data Economic Applications capable of powering DENETs that implement powerful Data Economic Solutions across sectors. These Chapters build off the concepts introduced in Chapters 4 and 5.

3. Chapters that will explore specific Data Economic Solutions that can be (and are being) built using the Lydion DEOS and the types and functionality of Data Economies they are implementing, building off the concepts introduced in Chapters 3, 4, and 5.

4. Chapters that will build formal mathematical and economic structures for foundational and advanced Data Economic principles, building off concepts introduced in Chapters 4 and 5.

In addition, the **Science page on *Lydion Insider*** features various materials, including slide decks with diagrams describing concepts from Chapters 4 and 5 as well as more details on the architecture of the Lydion DEOS.