PURIFYING MARKETS: ALIGNMENT EXPERTISE AND THE COLLABORATIVE PRODUCTION OF GLOBAL MARKET INDICES

Alex Preda

Lingnan University, Hong Kong

ABSTRACT

I suggest here the notion of alignment expertise as group-based sets of operations and skills that simultaneously align (1) apparently incompatible worlds, (2) different domains of knowledge, and (3) heterogeneous layers of objects of expertise. The case on which I build the argument is that of market indices. These play a key role in global markets, being illustrative for the data world we live in. Indices are regulatory, epistemic, commercial, and automated objects. These features have to be simultaneously produced and maintained. Based on ethnographic observations in a data analytics firm, I argue that market indices align two very different worlds: the mini-world of econometric formulae and the messy world of computing machines. I identify two operations, grounded in collaborations between software engineers and financial analysts, that perform this alignment: first, a purification operation sorts out price and volume data into "source of truth" and "strange," respectively. Then, a second purification removes excessive, yet "truthful" data. I argue that these operations are anchored in coordinated expert skills of reading and writing in multiple scripts at once.

Keywords: Alignment; objects; coding; data; markets

Expertise In and Around Organizations: The Changing Constitution and Ecology of Expertise
Research in the Sociology of Organizations, Volume 97, 117–139

ISSN: 0733-558X/doi:10.1108/S0733-558X20250000097006

INTRODUCTION

I examine here alignment expertise, understood as group-based, collaborative operations through which (1) apparently incompatible worlds of formulae on paper and machines, (2) different domains of knowledge, and (3) heterogeneous layers of objects of expertise are aligned with each other. I ground the argument in an empirical examination of global market indices: used by fund managers, pension funds, investment banks, or arbitrageurs in their strategies, pointed at by politicians as signs of successful (or failed) economic policies, indices such as the Dow Jones, S&P500, FTSE, or the Nikkei have a global presence and an impact well beyond financial markets. Market indices have multiple, heterogeneous layers that depend on each other and that need to be kept aligned. This cannot be achieved without aligning two very different worlds: that of formulae on paper and that of computing machines, respectively. In its turn, this requires aligning different domains of knowledge: finance, software engineering, and data science.

Consider the following elements: (1) market indices are regulatory objects, subject to the specifications of market authorities, but also regulating markets. They enable the construction of particular instruments (e.g., ETFs, or exchange-traded funds cannot be built without indices), are intrinsic to the strategies of market actors (e.g., hedge funds, market makers), and are used to evaluate financial performance. (2) They are commercial objects, being sold by data providers (since at least the 1920s) on a subscription or ad hoc basis, for a profit. (3) They are epistemic objects, being the subject of academic and commercial research, with evolving methodologies and data. (4) They are automated objects, being produced, processed, and disseminated within global networks of data centers, computer servers, and transmission lines.

None of these laminations[1] is possible without the others: the existence of indices as regulatory objects depends on them being viable commercial, epistemic, and automated ones at the same time. Conversely, indices being commercial, epistemic, or automated objects depend on them simultaneously being regulatory objects and continuing being so. As regulatory objects, indices are well-defined instructions, with institutional force, to represent an ideal market.[2] As commercial objects, indices are data sold and displayed to customers within set parameters. As epistemic objects, they are econometric formulae subject to experimental modifications. As automated objects, they are data that have to be regularly generated and circulated by machines.

The puzzle is then, what operations and activities make it possible to produce objects that are all this at the same time?

In the following, I examine this question based on an ethnography of how market indices are produced and maintained. The case at hand is that of licensed crypto market indices produced and disseminated globally by a UK data analytics firm. As in traditional finance (TradFi), markets in crypto and their actors (both institutional and retail) make extensive use of indices licensed by national regulators such as the Financial Conduct Authority (FCA, UK) or the Commodities and Futures Trading Commission (CFTC, USA), indices produced and sold by specialized data providers. They are relevant not only because of their economic consequentiality on a global scale but also because they are emblematic of the data world we live in, where expertise is heavily involved in producing consequential data.

Building, maintaining, upgrading, circulating, and selling indices require a permanent collaboration of multiple domains of knowledge, not least those of software engineers, finance professionals, and data analysts. I ground the analysis in ethnographic observations of the operations deployed collaboratively in producing market indices, combined with interviews conducted with those involved in producing and using indices.

In a first step, I examine the literature on alignment expertise, objects, operations, and collaborations, with a focus on market indices. The second step presents the data and methods. The third step analyzes the ethnographic findings with regard to expert operations and the skills supporting them. I argue that market indices are objects that align two worlds: the mini-world of econometric models and the messy, real-life world of distributed computing machines. Two operations play a key role in this alignment. I call them here Purification I and II. Purifications sort and reduce data so that they can fit both worlds concomitantly. The conclusion discusses the relevance of studying forms of expertise that produce authoritative data and reiterates the notion of alignment expertise.

ALIGNMENT EXPERTISE AND MARKET OBJECTS

In relationship to expertise, issues of alignment have been conceptualized primarily as (1) alignment of expertise or (2) alignment expertise. Alignment of expertise means ranking different expertise domains and (experts) based on claims of knowledge absorption and authority (Bunderson, 2003). Especially when it comes to the relationship between software engineering and knowledge domains such as finance, alignment of expertise has been represented as software expertise absorbing knowledge domains (Ribes et al., 2019), as adapting to the demands of domains (van der Voort et al., 2021, p. 3), or as domain knowledge adapting to pressures from software expertise (Hansen & Souleles, 2023, p. 436). Alignment expertise though is understood as task-solving operations, grounded in skilled activities, that link together (Li, 2007, p. 265) apparently disparate entities or worlds, as well as knowledge domains, in order to produce objects of expertise, with specific properties.

In the context of financial markets, objects of expertise have been conceptualised alongside at least three dimensions: boundary objects coordinate the perspectives and interests of heretogeneous participants in collaborative activities (Leigh Star & Griesemer, 1989); epistemic objects generate uncertainties and focus curiosity (Knorr Cetina, 1999); infrastructural objects support distributed activities and relationships across networks of participants (Pardo-Guerra, 2019, p. 137). What objects of expertise would do then, generally speaking, is to coordinate participants, focus their attention, and support their distributed activities. Financial markets rely on combinations of boundary, epistemic, and infrastructural objects. These combinations, together with the social relationships emerging in their operation, are regarded as socio-technical assemblages (Borch, 2022; Callon, 1998; MacKenzie, 2009, 2021; Pardo-Guerra, 2019). Assemblages create affordances for financial action while accelerating the circulation of financial capital. While objects of expertise can be interconnected (hence the notion of socio-technical assemblage), they are usually regarded as distinct and not as laminations of one and the same artifact.

For instance, analyses of high-frequency trading, of rapidly evolving algorithms and machine learning (ML) processes in financial trading (Hansen & Borch, 2022, 2021) have rested on a distinction between infrastructural and epistemic objects. While the first designate interconnected datacenters, servers, and transmission lines supporting the operations of algorithms and ML tools, these latter are rather seen as epistemic objects that generate uncertainties, focus the curiosity, and have limited explainability.

An ML-based trading algorithm is an object of expertise with epistemic features: it might generate new uncertainties, attracts the curiosity of builders and users (having only partial explainability), and generates multiple narratives concerning its effects (Borch, 2022; Borch & Min, 2023). However, it is also an economic object that has necessitated investments, has operating costs, and is expected to produce over time a return greater than its development and operating costs. If it were to cease being profitable, a trading algorithm would be abandoned. Therefore, it would cease being an epistemic object too. We know indeed that trading algorithms have a limited lifespan and cease being used once their stop being profitable, or when they are detected by competitors.

Not only that: by definition, a trading algorithm is a set of formulae (Lepinay, 2011), but also an automated object: its working and the results it delivers are inseparable from the machine(s) it is embedded in and communicates with. While computer screens (on which its results are displayed) are treated some times as secondary objects (Nicolini et al., 2012, p. 625), in financial contexts, they are treated as primary objects that both incorporate and display "the market" (Knorr Cetina, 2003; Knorr Cetina & Bruegger, 2002). Similarly, computing machines are primary objects, as trading algorithms will be written to run on machines with specific properties. If these properties are modified, so are the outputs of the algorithm and its functioning. Donald MacKenzie (2021, p. 167) highlights that the language in which algorithms are written is chosen in such a way as to be close to the machine, in terms of physical implementation of the algorithm. When it is modified or collapses, the algorithm will cease to be a commercial object and an epistemic one. Conversely, if such an implementation becomes too expensive, the commercial character will be endangered. This points to the fact that epistemic objects cannot be entirely separated from infrastructural ones. Nor can they always be separated from commercial objects as, in financial markets at least, they are evaluated on their costs and the profits they generate. Overall, my argument is that we should consider objects of expertise not only under the angle of their connections but also as having laminated features that depend on each other.

A trading algorithm (or, for that matter, a market index) would require aligning not only the interests and perspectives of participating experts but also two very different sets of entities: abstract formulae, on the one hand, and machines that enable the production of trades (or index numbers), on the other hand. Similarly, in order for a simulation model to produce forecast numbers, it would need to be aligned with computing machines. The formula of an algorithm cannot produce any trades without such an alignment, which is neither given, nor natural, nor unproblematic. This alignment is crucial for an algorithm (or an index) being an epistemic, commercial, and regulatory object at the same time. Aligning paper formulae with machines in such a way that trading outputs, or index numbers are regularly and continuously generated necessitates operations, understood as coordinated, iterable actions grounded in group-specific skills.

Such operations require, more often that not, collaboration across different, group-based expert activities (e.g., Collins & Evans, 2007; Collins 2018). Collaborations have been treated in terms of the operations that define the participants' expertise (e.g., Pakarinen, 2024, p. 22) or in terms of their impact across domains. This impact can include semantic and syntactic transfers (e.g., Bechky, 2003), expert languages that extend across or creolize domains, the dominance of one expertise over the other, or the substitution of one through the other (Collins et al., 2010, p. 9). Expertise as a set of group activities is defined by (non)discursive skills and practices (Collins et al., 2017, p. 777), the character and understanding of which depend on the relationships within which they are deployed (Pakarinen & Huising, 2023). The implication here is not only (and not primarily) that expertise varies as it is displayed to different audiences (Anteby & Holm, 2021; Eyal, 2013), but that expert skills (such as coding, or computing an index formula) can evolve depending not only on how expert participants relate to each other but also on how they relate to the objects of their expertise.

For the context investigated here, that would mean that the expert activities of producing and maintaining a market index are deployed within a set of interactions and relationships within groups of experts. As the index needs to align different layers (e.g., commercial, regulatory, machinic, etc.), we should expect the operations that produce indices (and the participants' understanding of their expert skills and collaborations) to be geared toward this alignment. In the following, I will examine closer the expert operations that align indices, together with the skills that support them. Before that, however, I will provide a short overview of market indices as objects of expertise.

INDICES AS OBJECTS OF EXPERTISE

Market indices have been seen as crucial in financial economics at least since the development of the capital asset pricing model in the early 1960s, as they play a significant role in estimating the risk of financial assets (Perold, 2004, p. 21). Risk profiles of individual financial assets are calculated against a market benchmark (usually an index); so are excess returns on stocks against fixed income instruments (Perold, 2004, p. 4). While indices such as the Dow Jones were already introduced in the late 1880s, and have their origins in financial journalism, starting in the 1920s, they became an object of econometric methodologies concerning the measurement of returns or of market fluctuations (Hautcoeur, 2006).

From the beginning, at least in the USA, market indices were produced and sold by private firms. Starting in the 1930s with the work of the Cowles Commission, indices became objects of expert collaborations: financial economists collaborated with statisticians in producing them (Hautcoeur, 2006, p. 14; Porter, 1995). Indices have remained objects of academic expertise, but also of professional expertise within financial firms: global marketplaces such as the London Stock Exchange or the Chicago Mercantile Exchange (CME) produce and sell a manifold of indices using their own trading data. Additionally, data providers (e.g., Bloomberg) produce and sell their own indices, using data feeds from different venues. We have thus indices produced on a venue's own data and indices computed using data across different venues (the kind investigated in this paper). Market indices can be the object of a private contract between a client and the provider, or they can be sold to several clients. In this latter case, regulators require that index providers disclose their methodology to the public, so that indices can be replicated, audited, and are representative of underlying market realities. Anyone following methodological specifications should be able to apply the formulae to data and obtain the same results; the way in which results have been obtained should be verifiable. Indeed, providers such as Bloomberg, CME, or the analytics firm discussed here have made their methodologies public.

The formulae for computing market indices are largely similar and consist of volume weighted averages of the last trades over a given time interval. As data coming from different venues have different latencies (i.e., arrive at different speeds at the venue computing the index), some price and volume data might come at a speed lower than the average and therefore not reflect the last trade anymore. Index formulae penalize lateness by factoring in lower weights for price data that arrive later (by a set factor) than the latest transactions observable on the exchange in question. Another penalty is assigned to price data deemed to be outliers when compared with the prices at which the same asset trades at the same time on other venues. For instance, if asset A trades on venue X at a price that is higher (or lower) by a set factor of *n* compared with the venues Y, Z, Q, etc., it will be assigned a lower weight in the formula. In other words, the (public) index formula is built in such a way as to weigh down price anomalies and data lateness, in order to maintain the representative character of the index. Thus, the public, licensed formula of the index aims to ensure that events happening on one exchange (high/low prices, late prices) do not impact the representation of "the market" as a reality broader than any single exchange.

DATA AND METHODS

The data analytics provider (henceforth: the firm) where I conducted ethnographic observations was based in the UK, with engineering teams in two other countries, across three different time zones. It produced and sold licensed market indices, bespoke indices, market order books, and market data. Among its clients were global investment banks, as well as a major, globally influential central bank. The index subscription was also sold to accounting and custodian firms, and to node validators on the Ethereum blockchain. As a commercial object, the index was continuously expanding.

At the time of the fieldwork, the firm was working on new products such as analytics of social media feeds. It was in negotiations with a major Asian exchange for becoming their data analytics provider on crypto assets. The indices were sold by subscription to institutional clients. Retail clients could buy index data on an ad hoc basis. Clients could order custom-made indices that were sold only to them: this is in line with the operations of established providers, who sell both licensed and bespoke indices.

The firm collected live data feeds on the 40 most traded assets from 250 crypto exchanges worldwide. The incoming feeds were transferred into the own databases and used to compute a licensed market index that was then disseminated to subscribers and regulators. The index had to be computed continuously, as crypto markets are 24/7.

The firm had marketing, sales, finance analytics, and engineering teams working together in an open plan office on a hot desking principle – meaning that desks were assigned on a first come, first served basis. Engineers and finance analysts mixed together while working and more often than not sat at adjacent desks.

After negotiating access for more than a year, over the course of two months during 2023, I spent a full workday every week in the open plan office, sitting at a desk next to the CTO's, in the zone where finance analysts and engineers were working. As part of granting access, only handwritten notes were taken; interviewing members of the firm was conducted only during breaks. All firm members in the UK office had agreed to my presence. I conducted a total of 39 hours of observation.

I sat next to the CTO, with direct visual access to the engineering and analysts' desks. I could observe directly the interactions between software engineers and finance analysts, between engineers, and between analysts. I could observe directly Slack meetings as well, and the work processes in which both sides, separately and together, were engaged. I took onsite handwritten notes from conversations and interactions, as well as from the observed work processes. I also took handwritten notes from the oral accounts of the CTO about what engineers were doing at any given time. During breaks, I conducted interviews with: the engineers working on the index and the social media feed; the heads of index operations (twice), of order book operations (twice), of market data operations (twice); the data scientist working on the index. I transcribed the interviews in handwriting. I also conducted offsite interviews (recorded and transcribed) with the CTO and the head of index operations. In addition to the interviews conducted on- and offsite with members of the firm, I conducted, together with collaborators: interviews with 31 software engineers and 33 financial analysts involved in crypto and blockchain projects, in order to elicit accounts of the expertise involved; interviews (recorded and transcribed) with 22 crypto exchange managers, 11 market makers, and 5 OTC traders, in order to understand how indices are used in crypto trading.

While in the field, I started observing the main operations required by building and running the index. I cross checked observations against interviews in the field. I elicited in interviews rationales for the necessity of these operations. To outside observers, these rationales are not always immediately visible in the participants' operations and have to be reconstructed based on interviews. I built categories of the work processes involved in generating market indices and assigned them to the operations performed in producing and maintaining the index. I then built categories of the types of collaborations between software engineers and finance analysts and assigned them to work processes pertaining to each operation. I built categories of work processes that required a collaboration, and work processes that did not, and of the discursive knowledge and skills involved in each (see Table 1). I cross checked the operations and skills I was observing in work processes on the ground against interviews with engineers and analysts from outside the firm. I cross checked onsite against offsite interviews and against interviews with financial engineers and analysts from outside the firm. I cross checked the self-description of the onsite participants against onsite third parties descriptions, and against self-descriptions from interviews, to see whether there were hard boundaries between these expertises, as understood by participants on the ground, and whether these boundaries could possibly shape collaborations and work processes.

MINI-WORLDS AND MESSY WORLDS

As a standard, regulators require indices to be replicable, auditable, representative of the market, robust, and stable. Any changes to the publicly available formula had to be approved and made public at least three months before they came into effect. A white paper (available in online open access) detailed the index formula, every single variable in it, the rationale for including these variables, and how every variable should be calculated. Transparency and stability of the formula were also seen as increasing the commercial success of the index and making it into a standard. The formula represented thus an officially sanctioned, expert-produced, efficient, transparent, stable mini-world (Yonay & Breslau, 2006) that included only a few variables (price, volume, time) while leaving out others (for instance, it didn't include the machines on which the index was computed and their properties).

While present at the back of their minds, the formula was rarely if ever present in collaborative discussions or other joint activities of engineers and analysts. It rarely was brought up, mostly when a client required changes and had to be rebuked by the CTO. The topics that dominated pre-announced discussions were mostly about blocks, validators, nodes, and the Ethereum chain. Spontaneous discussions centered almost exclusively on data, inputs, outputs, errors, and latencies; among themselves, the engineers talked about machines, queues, stalling, and more machines (see Table 1). Even when it came to actions and non-discursive skills, the formula didn't seem to play a role. Rather, it was seeing, reading, writing, correcting, removing, and making judgments on data that took the center stage – all actions and skills that had to be deployed in groups, not individually. Why was the formula quasi ignored?

Participants involved in producing, maintaining, and disseminating the index had to ensure that it had the above properties (replicability, auditability,

Table 1. Expert Skills and Knowledge in Producing a Market Index.

Collaboration Types	Participants	Discursive Knowledge About	Talk Topics	Non-discursive Skills	Actions
Online presentations to analysts and engineers, including those physically co-present	(All of the below, always) CTO Heads of: index, market data, order book operations Financial analysts Software engineers Data scientist	Crypto assets traded on centralized platforms and in decentralized finance The Ethereum blockchain and Ethereum accumulation at validator nodes Traditional financial assets (e.g., swaps, futures, derivatives)	Nodes Validators Rewards Sanctions Blocks Ethereum Periods Perpetual futures Asset pools	N/A	Making judgments on numbers (price, volume)
Spontaneous discussions/ unexpected summons between analysts and engineers while looking and pointing at screens	(Most times either head of index or head of market data or both were part of the group) Heads of: index, market data operations Financial analysts Software engineers Data scientist	N/A	Data (price and volume) Input data Output data Errors Latency Liquidity	Writing/reading code Reading in decimal and hexadecimal scripts at once Producing numbers in two scripts at once Seeing prices	Making judgments on numbers (price, volume) Removing/ modifying numbers Producing synthetic numbers
Engineers coding together on shared screens (including physical co-presence)	(Mostly CTO and other engineers) CTO Software engineers	Machines (number of cores, capacity) Machine tools	Messages Queues Machines (building) Machines (stalling and other problems) Endpoints Throttling	Writing/reading code Seeing errors in code Localizing code on screen Seeing code and machine performance at once and in relationship to each other Identifying/finding/using tools appropriate to the problem	Making judgments on machines Using tools to maintain/repair machines

representativeness, robustness, stability) at any given time, while handling uncertainties that arose in these processes. In theory, anyone could have replicated the index to check its validity. For an exact replication, however, the formula would have had to be coded on virtual machines identical with those used by providers; the machines would have had to be configured on the same server farms (keeping the same data feeds); the data specified in the methodology would have had to be obtained from trading venues and processed in the same way: the index feed would have had to be sent to clients with exactly the same latency. That is, in order to replicate the index, the efficient and transparent mini-world of the econometric formulae needed to be connected to a much messier, much less transparent world, made of virtual machines, cloud computing, data flows, and messages. This connection, or alignment had to be produced first: this was where engineers and analysts deployed their skills, and this is what they mostly talked about when they summoned each other.

The index had to deliver authoritative numbers regularly, with low latency, and not be impacted by market conditions, such as volatility. An unstable index would have oscillated too much when markets were volatile, and this was not supposed to happen. Additionally, crypto markets could be impacted by forking – that is, situations where an asset splits into two as a consequence of changes made to the blockchain protocol that generates the asset. Examples of new crypto assets created by forking are Bitcoin Gold, Bitcoin Cash, or Ethereum classic. When forking occurs, the index should incorporate the trading prices of the old asset, not of the new one, as the latter does not have enough trading history. Even worse could be situations when the index would mix prices from both assets in the calculation. The index also had to reflect a fair market – that is, a market without manipulation or errors. The formula had no variables for typing errors or for market manipulations. (It had an error term though, which could not be allowed to become too large.)

Thus, the index as continuously produced data was not required to be representative of a market with high volatility, where unexpected events such as forking occur, where traders commit errors and manipulate prices. It was required to be representative of a mini-world – a market with few anomalies, few errors, honest traders, stable, and where supply met demand quickly (deep liquidity). Periodically, regulators audited the index for these properties. Yet, this mini-world – simple, clean, efficient – was incapable of generating data on its own. In order to produce numbers, a far messier world was needed, one including an array of machines, together with engineers, analysts, and data scientists. This was not an easy task; it could not be automated. It required complex collaborative operations, supported by group activities and skills, and deployed within collaborations, to which I turn now.

PURIFICATION I: A LIBRARY OF STRANGE CASES

The firm collected price and volume data on the 40 most traded crypto assets from 250 exchanges worldwide via live feeds, with an average latency of 2 miliseconds. The data underwent a first set of operations before going into a database. From there, it went on a UK-based virtual machine that computed the index numbers. As it was going onto the machine, the data underwent a second set of operations. Once computed, the index numbers went to subscribers and were fed back into the exchanges (see Fig. 1).[3] Subscribers could see the index updating on their screens within a set period of time (e.g., every minute) specified in the contract. (A more expensive contract had a lower updating latency.) What subscribers and exchanges could see on their screens depended on: (1) the firm's visualization (the API, or application programmable interface); (2) data ordering tools (the data library); (3) what the firm decided to send back to them (index data, or index data and meta-data); (4) how the subscribers' machines communicated with the firm's.

At all times, the index had to be representative not of a real market, but of a pure mini-world. Consequently, engineers and analysts engaged in purifying data of numbers that indicated real-life happenings (errors, manipulations), as much as they could.

All the operations had to be done in real time, as the live data feed arrived from a messy world: exchanges distributed around the globe, on which some traders could make errors or try to manipulate prices. None of these would have fitted the mini-world of the econometric formulae. The index data also went back to a very messy world and was dependent on key variables that were to be found nowhere in the formula: data library, API, properties of virtual machines. As the live feed arrived, the analysts, data scientists, and software engineers did not know yet which data was error, manipulation, or a genuine transaction. Errors and manipulated trades had to be filtered out. The formula also required that price data concomitantly deviating by a preset order of magnitude from at least two other exchanges is weighted down in the calculation.

Before the data went into the database, all these operations had to happen: decide which data indicates a trading error, which data indicates a market manipulation, which data should be weighted down or not. As the data scientist (with a PhD in mathematics) explained to me, there was no reliable algorithm to

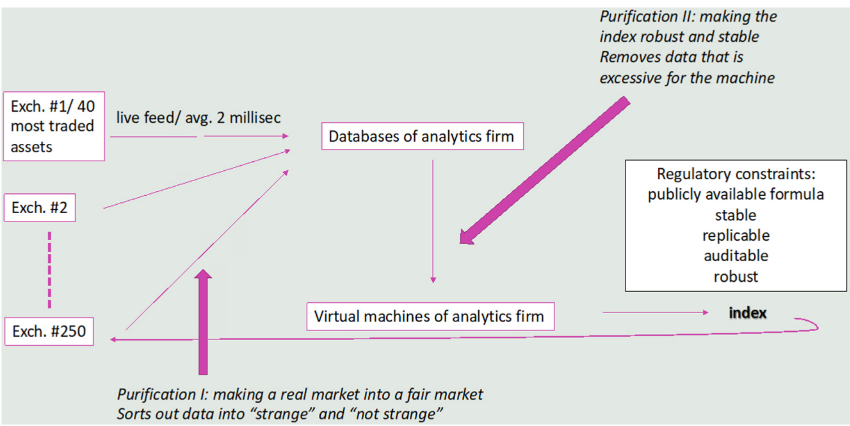

Fig. 1. How the Index Is Produced.

automatically and universally distinguish between an intentional order and an erroneous one; he could not use algorithms to automatically filter out all trading errors. Neither were all market manipulations detectable by algorithms. The detection of wash trading (one player buying and selling the same asset continuously) or of spiking the price (putting in intentionally large orders to attract other traders) requires human surveillance skills. Even if price spikes were weighted down by the formula, a manipulative one would still impact the index, bringing it further away from representing a fair market. According to the head of the order book, spoofing (placing large numbers of limit orders and then cancelling them, in order to move the price in the desired direction) could be detected only in level 3 data[4] coming from exchanges and required the attention of data analysts.

One of the methods used by the data scientist was to go through past order books and build a catalog of, as he said, "strange cases" and then monitor the live data feed against this catalog that was augmented with new cases. Another approach was to generate synthetic errors, "spike" the data feed with them (before going into the database), and see if and how many an algorithm would detect. Both approaches required that the data scientist monitored the data feed (and removed undetected errors). Had the engineers and analysts let errors and manipulations slip into the database from which the index was calculated, the latter would have been distorted and ultimately the license would have been put in danger. They could not do this without seeing and judging first, in real time, what data is error: yesterday's errors are not necessarily today's ones, and even if they could write a program to filter out some expected errors, there were unexpected ones too. Neither could this filtering out be done on an individual, isolated basis, by a single engineer or analyst. In an interview previous to the fieldwork, the head of index operations complained that in the beginning engineers were putting onto the excel spreadsheet zeroes for errors, which distorted the index. She had to filter out the errors on the spreadsheet herself. This implied that seeing data as erroneous or not was not enough; one had to decide as well how to write data. "Seeing" and "writing" were key collaborative activities.

These operations, which taken together I call here purification I, are distinct from what has been termed purification of science, understood either as a demarcation operation (from other domains – see Elam, 1999; Latour, 1988) or as the introduction of explicit regulatory norms (Faulkner et al., 2006). It did not consist in classifying data into "good" and "bad" categories according to a general set of criteria (this could not be done). It was a collaborative, visual evaluation of how much individual data, as it appeared on the screen, was aligned (or not) with what has been seen before, coupled with decisions about how to record data in multiple scripts.

PURIFICATION I: OBTAINING TRUTH THROUGH SEEING AND WRITING

When a transaction in a crypto asset is executed, the asset will be moved between wallets by a smart contract (i.e., a piece of code that automates the transfer without a clearing house). If the smart contract has vulnerabilities discovered by hackers, the asset can be moved to a different wallet. This will make the transaction an invalid one – manipulations and frauds cannot be allowed into the database of the index. Engineers and analysts had to verify the authenticity of transactions – an operation for which they needed to look at the metadata associated with them. In one instance, the data scientist, together with engineers, huddled around the screen discussing how to establish the authenticiy of transactions:

> Data scientist [talking to engineers while all looking at the screen]: … transaction, do you have a key value? [… points at the screen] that's not going to tell you what is exactly happening …. So one simple block number, so the smart contract is good, transaction is authentic, so this Ethereum, right, just checking. So you need also the logs associated with this, the input will be the function that triggered the transaction, you need these logs associations … can you put also that hash in?

Verifying the authenticity of transactions required thus monitoring not just price and volume numbers, but also the logs showing transfers across wallets, that had to be visualized on the monitors shared by engineers and analysts.

Additionally, data arrived on the live feed with a time stamp from the original exchange and was time stamped again on arrival: as stamps did not show the same time difference for all data, engineers and analysts had to recalculate and synchronise the original time stamps on the data before it went into the index – which means that data had to be re-written too. This was a key operation, as the index was the volume weighted average of the last trades, and time stamps mattered.

On the live feed, the data was arriving in a decimal script, but was going from the database onto the machine in a hexadecimal script. For instance, one Bitcoin can be US$26,000 in decimal script, but in hexadecimal script, it is 6,590. Analysts and engineers had to make sure that the numbers going onto the virtual machines correspond to the numbers going in and out of the database – i.e., that they see the same price or volume in two totally different strings of numbers.

Members of the index team had two or three screens on their desks: on one screen the live feed of price and volume coming from exchanges was displayed on an excel spreadsheet. On the other screen, the live feed was displayed in code (with numbers in hexadecimal script). A third screen, where present, was used to monitor data latency and the operations of the virtual machine. The index formula (written in code) had to pick up the right data in hexadecimal script and compute the index – hence the second screen, on which the price and volume were displayed in hexadecimal script.

On several occasions, the head of market data referred to the live feed (see Fig. 1) as the "source of truth" and insisted that the data they were producing had to be situated within a "tolerance range" with the "source of truth." The latter could be taken as the source from which truth has to be extracted, but not as truth in itself. The very same head of market data said though that only "genuine market movements" should be left in the "source of truth." This implied that the live feed was providing only the rough material that needed to be purified in such a way that nothing but true market movements and true transactions come to the fore as "the source of truth." These true movements were produced by the team – in the sense that they, together, decided which numbers will be the market and which will be not.

These decisions required that analysts and engineers engaged all the time in spontaneous, collaborative, and indexical seeing of "the source of truth": regularly, during my observations, an engineer or analyst would interrupt their work, point at a screen, and summon others with the words "what is this?" or "why is this here?" Then the summoned analysts and engineers will interrupt what they were doing and huddle together and discuss, alternatively pointing at the screens, sometimes for up to an hour, deliberating on the character of the data on their screens, until they reached an agreement on the data that is right for the index. The character and the content of the discussion were entirely indexical, as the participants kept pointing at the screens, comparing the data points with yesterday's ones, modifying codes, checking if transactions are authentic, and coordinating time stamps. Due to this indexical character, an outside observer could not very well comprehend what was going on without knowing in advance what analysts and engineers were concerned with. Interviews with the head engineer and the head of index operations revealed that they were concerned with filtering out errors, checking time stamps, checking authenticity, and ensuring that the volatility of the index remained low, as a condition of its stability.

These were spontaneous collaborations arising in the middle of work: nevertheless, invariably, analysts and engineers answered when they were summoned to come and look at the screens together and decide on the quality of their data. As they were searching for correct data in two scripts at once (decimal and hexadecimal), I would argue that, together, they developed the skill of seeing "genuine market" data, a skill that anchored their purification operations.

This could not have been achieved simultaneously, but separately, simply because this seeing had to take place in both scripts at once. I am not arguing here that engineers and analysts did not practice this seeing separately as well. However, every time they were in doubt about it, they summoned each other and responded promptly, spending considerable amounts of time discussing the data in front of them. One might argue that this skill of seeing data could have been brought in by team members who had previously worked on indices in TradFi (the emic term was non-natives). Conversely, one might argue that this skill needed to be developed exactly because many team members did not have work experience in traditional finance (i.e., they were crypto natives). If the first were true, then I should have seen non-natives teaching crypto natives how to see a "fair" index. I did not observe any such teaching. If the second were true, then I should have seen crypto natives summon non-natives to teach them how to see. I did not observe such summons either. Rather, this was an entirely interactional process, in which no reference was made by the parties to previous experiences of seeing data in TradFi.

What was Purification I then? To stay in line with the practices I have observed (that were largely inductive), I will attempt to infer a definition: Purification I was oriented toward the mini-world of the index formula. It sorted numbers that fitted the prescriptions of this mini-world, by extracting the "truth" from its source (the live feed) and confining "strange cases" to their own repository. These latter were not simply discarded, as they had their own use as a collectively accessible memory,[5] against which new cases could be evaluated, and as templates for generating sythentic errors. "True" numbers were re-written and made ready to go onto the machine. This combined sorting and re-writing operation was grounded in collective, indexical skills of seeing and evaluating numbers in multiple scripts at once.

PURIFICATION II: MAKING A MACHINE THE SOURCE OF TRUTH

Once Purification I extracted "genuine market movements" from the live feed – data that fit a normative model of pure markets – they were transferred from the database onto the virtual machine computing the index. This machine was situated in a much messier world than that of index formulae. The data, already purified once, had to be subjected to another set of operations so that it fits a messy world. This second set, which I call here Purification II, removed excessive, albeit pure data that could imperil the working of the virtual machine.

The index *had to be seen*, in its fairness and robustness, by subscribers on their screens. Without being visible to subscribers, it was worthless. If fairness of numbers was a matter of seeing together possible errors and manipulations, stability, robustness, and visibility depended, to a large extent, on another layer of the index: it was an automated object, located within and processed by machines. If the machines became unstable, the index became unstable too. What is more, if clients were to be able to read the index on their screens, then the numbers produced by the formula needed to fit the library (i.e., the collection of precompiled routines and scripts) and API (application programmable interface) set up by the firm, as well as the machines of the subscribers. Mismatches between numbers, library, API, and the machines of subscribers could impact the latency of the index and, potentially, its robustness.

Overall, stability, robustness, and visibility depended not only on which numbers were selected from the live feed and made into the source of truth but also on the properties of the machines producing it. These properties were not independent of the numbers being fed into machines, and on how they were fed. If too much data – that is, too many volume and price numbers were put on the machine, it took more time to calculate the index (even if "more time" meant more milliseconds). The numbers then kept accumulating and the machine began stalling. When the machine was stalling, the index lost its low latency or, even worse, could become unstable, less robust.

Another aspect was that if the numbers kept accumulating on the machine computing the index, it took more time and computing power to generate the index, which became costlier for the firm to maintain. If that were the case, the index became a less efficient commercial object, eating into the profit margins of the firm. The machine had to work efficiently – that is, producing index numbers within the specified latency but also generating a profit for the firm. If computing became too expensive, the firm started losing money on their flagship product.

During my fieldwork, on more than one occasion, the CTO and surrounding engineers were preoccupied with how to send index data to subscribers. For that, the API and the data needed to be adjusted to each other, but the API wasn't treated by engineers as entirely controllable. The index data could be sent out as it was produced; however, if the firm pushed too many updates per minute, exchanges responded by increasing the data feed coming to the firm, which would have created computational problems. Additionally, when trading activity spiked, the firm had larger incoming dataflows *and* larger outgoing dataflows and needed to compute more (which increased the cost of the index). An alternative would have been to aggregate data instead of queuing it up. Aggregating data, however, required more lines of code, slowing down computations.

The firm was operating on a cloud architecture, not a centralized database: that is, instead of a single server farm, virtual machines were configured on various servers located in the GMT zone (to avoid timestamp issues). An advantage was that machines could be reconfigured according to needs; engineers could also add more machines, as they developed more products. However, this made computing time and machine rental an important cost factor. Engineers were permanently preoccupied with maintaining the index computationally profitable, while, together with the analysts, keeping it robust, stable, and representative of a pure mini-world. In some instances, they were adopting a language of commercial efficiency to rebuke clients: during my fieldwork, the CTO of the firm was on a call with an institutional client, where he refused to modify the index, saying that not only they would lose their license, but the modification itself would cost hundreds of thousands of pounds that would need to be recouped over many months.

Engineers and analysts had to address issues of machine efficiency all the time (e.g., machines stalling, high latency, breakdowns) – after all, the index was an automated, 24/7 object, impossible to produce manually. The head of index operations, whose background was in finance, had learned how to restart a virtual machine:

> So things like basic things, actually, for example, an index is not working, like I can see, like, I see the charts and I see that the prices are not updating, for example. And sometimes, a developer is not always there. For example, so we don't only work for Europe, but we also work for abroad. And we have clients in California, for example. So we have a time difference. So there was an index that broke, and we were all sleeping at, like 1am in the morning, I got a call, like we need these prices, and we're not getting them. So I had to grab my laptop and start checking what was happening because yes, there is what's right, like the prices were wrong. So I had to go and maybe restart the service that was calculating that index. That's something that I wouldn't think I would have to do, right? Like, I don't know, I'm just like, I calculate the model and then pass it to a developer. But yeah, like in the end, *I'm the one that actually understands how the prices are being calculated* [my emphasis]. So maybe they call a developer, they don't realise that the price is actually wrong. So that's why they call me and I need to know this basic stuff, like how to start the calculation again, or things like that. … So it's actually writing a script, but it's running like a Javascript that I wasn't familiar with that before. So yeah, it's going into the cloud connecting into the machine and rewriting our script. That's it.

In this excerpt, the head of index operations describes how she had to fix a problem in the middle of the night using skills that she didn't previously regard as being part of her expertise – but which she sees now as intrinsic to her understanding of price calculations. She had to connect to a virtual machine (into the cloud) and rewrite the script so that the machine can restart and recalculate the prices. She made clear that learning how to repair a virtual machine was somehow completing her skills in seeing "fair" prices (she is the one who understands how prices are calculated):

> I have to understand the whole process, because maybe we're growing a lot. And there's not enough people either. So it's better if I understand it, because then I can solve all the issues. There are many things that I think at the beginning, I had no idea. And now I can go into our beautiful machine and fix it myself. But before that I had to rely on a developer.

This extends the notion of calculation from paper formula (the model) to the workings of a machine (the whole process), an extension that is not captured in regulatory instructions.

One could argue that learning to restart a virtual machine would be akin to a simple operation, such as flipping a power switch: the analogy, however, would be inaccurate, since the latter doesn't require complex operations (going into the cloud, connecting to a machine, rewriting a script in code) and does not complete an understanding of something else (how prices are actually calculated). Flipping a switch can be done without all these operations and understandings.

In order to keep the machine working efficiently, another set of purifications had to be undertaken: extracting from the set of already produced "genuine market" data the right amount for the machine. The data had to be purified of amounts that are excessive for a messy world, even if this data is appropriate for a pure one. As the volume of data varied with market activities on the 250 exchanges, there was no way of determining in advance, by calculation for instance, how much should be purified.

During fieldwork, in the engineering area of the office "machines," "messages," and "real time" were the most frequently heard expressions: not only because launching new products required preparing new machines, but also because engineers had to monitor the index machine for stability and robustness. Several factors were in play here: deciding the amount of price and volume data to put on a machine so that it does not stall; deciding whether to send subscribers only index data, or data and metadata (sending out data and metadata made the message lengthier); making sure that the output data fits the API (e.g., if metadata were to be sent too, the API would have had to be adjusted); monitoring possible unforeseen market events that could jeopardize index stability. For instance, a sudden spike in trading would result in more data arriving, and more messages being sent to the index machine, making it stall. In the past, an incident had happened when a crypto coin had forked (i.e., a change had happened in the chain protocol) and, as a consequence, the index had become unstable. Engineers had to monitor for such events and take measures if the index became unstable.

Not only unforeseen events such as forking or increased trading volumes could pose a challenge for the machine but also unexplained malfunctionings. One day during my fieldwork, the machine started putting messages inside each other instead of ordering them one after each other, as they were coming. Bloated data strings slowed down computations to a point where the machine was working with a much reduced capacity.

While there was a monitoring team in a different country supposed to alert the engineers for accidents or incidents, it was ultimately the latter's task to address both kinds of issues, which could happen globally, in an unforeseen way, and 24/7. On more than one occasion, the CTO, engineers, and the head of index operations had been called at weekends, in the wee hours, because the index was not updating correctly. Hence, intervening to keep the machines stable and robust was a collaborative activity engineers engaged in all the time during fieldwork.

PURIFICATION II: FITTING MESSAGES ONTO THE MACHINE

In a previous section, I have discussed the spontaneous collaborations between analysts and software engineers who were coming together in front of the screens to see numbers that could be unfair, inauthentic, or temporally uncoordinated. I have called this purification I, a set of operations that sorted out from the live feed data that fitted a mini-world. Purification I relied on engineers and analysts collaboratively monitoring the data feed in two scripts simultaneously, and deciding whether to keep data or send it to the repository of "strange cases." This monitoring required reading and writing together in multiple scripts at once.

Purification II removed excessive, yet "fair" data that could impede the functioning of the machine. It brought together engineers deciding how to put messages on machines. A "message" was data deemed "fair," with or without metadata. Engineers cared about how many messages should be let onto the machine so that the index stays stable, robust, with a low latency, and it the machine does not stall. They had to care about how often they can update the index depending on the properties of their machine, the properties of their clients' machines, and the fit with the API and the library too. On several occasions, the CTO made statements such as "if you're on this machine, the maximum [for the frequency of updates] is every second" or asked other engineers to test the machines of their clients. For engineers, everything was "messages" and "endpoints," terms I never heard analysts using. If trading activity was high, engineers could throttle the API – that is, limit the number of requests their subscribers could make per unit of time. The CTO, however, was reluctant to throttle ad hoc too much, as the firm would have to provide a methdological explanation as to how it was being done.

Messages could be added or taken off the machine, depending on how overloaded it was. Engineers came together on Slack, sharing coding screens, and decided how many "fair" messages they should let in. During fieldwork, they sometimes worked in pairs on this task, and sometimes in groups of three or four. Even if they were physically co-present, engineers preferred to work online and share screens rather than come together huddling in front of the same screen, as I had witnessed them doing during Purification I. They usually worked with three screens: a coding screen, a Slack screen for messaging, and a screen for monitoring the performance of the machine through a series of dahsboards (latency, number of messages coming in, how heavy the message load was, free capacity). Oftentimes, I observed the CTO working together with engineers on coding via a shared code screen, while talking to and messaging his collaborators (see my screen, look at my screen!). To some extent, the setup was similar with that used by analysts and engineers for producing fair numbers, in the sense that messages were displayed in two scripts: decimal (on the machine dashboards) and hexadecimal (the coding screen).

As the volume of trading varied, engineers could not assume that they were going to get the same volume every day, and hence apply the same filtering rules. Neither could they assume that the machine will behave exactly like the day before. Deciding how many fair messages to put onto the machine was a task that had to be done anew every day. After all, the index was indeed an epistemic object: one couldn't know what the next day will bring. Not only this: sometimes, messages could be sent to the wrong place and this made the machine malfunction. Engineers had to work backward from the malfunction, find what was causing it, find the repair tools, and then find a way of repairing the machine. The following excerpt renders a situation in which messages were placed within other messages (instead of being queued up), making them more than double in size and stalling the machine:

> CTO [coding with two other engineers on a shared screen]: something has happened … it's doubling every time … why is it doubling every time? Did somebody change something? … oh it puts its deb file back into itself! That's what it is, that's why it doubles in size every time, 'cause it's putting itself into itself we know why [expletive] 'cause it's doubling plus 90 meg … I don't even know where the Jenkins machine [a repair tool] is […] I'm down to 7% usage 'cause you can't physically … it takes time to download and not it's got physically 3 times … poor machine …

In collaborations with other engineers, the CTO used digital markers (usually red or green) to highlight areas of code that needed to be changed in order to reduce or augment the amount of messages on the machine. As with the collaborative work on "fair" numbers, this work was largely indexical: the CTO would circle with the marker an area of code on the shared screen and just say, "here… and here, we need to take this price and volume out." The skill of directly seeing price and volume in lines of code was apparently shared by engineers, who never raised questions about this. With an eye to the diagrams showing machine performance, the engineers would then decide whether to add or delete messages to it. This skill of seeing was used to make a distinction between right and excessive message loads (and to intervene accordingly), crucial for keeping machines aligned with users and with the mini-world of the formula.

CONCLUSION: ALIGNMENT EXPERTISE

I have started this paper by asking, What expert operations make it possible for market indices to be generated as authoritative numbers with global impact? Authoritative numbers with constraining force are key components of economic institutions big and small. They shape economic policies at global and state level. Market indices could not be generated, maintained, and disseminated without aligning two distinct worlds, namely formulae and machines. As indices play such a role on a global scale, the question about the operations that produce them is consequential for, among others, understanding how global markets work. In a broader perspective, indices as authoritative numbers illustrate a puzzle of our data world: how do skills-based group activities produce authoritative data?

A commonly held trope is that "algorithms produce data" – that is, that formulae that are input into machines would somehow automatically generate authoritative data. This would imply a kind of fusion between formulae and machines that leaves no place for the intervention of human, skill-based operations. My argument, however, has been that this production and display requires the alignment of two very different worlds: the pure one of (econometric) formulae, and the messy one of computing machines. This is neither given nor unproblematic, nor automatic. It is achieved by two purification operations deployed within small groups. One is the purification of numbers as "source of truth" that respects regulatory conventions: it sorts out pure from impure data, without discarding the latter (as it has its own uses) and re-writes pure data as it fits them onto machines and the other is the purification of excess, so that machines can support this source. The other operation removes the excess of pure data, adjusts its qualities to the machines processing it, and makes it fit for being displayed. These operations appear to play a key role with respect to how the constraining force of numbers is achieved, and they could not be deployed without expert skills.

Based on the above, I would submit here the notion of *alignment expertise* as encapsulating the task oriented operations that simulatenously link and coordinate: (1) apparently incompatible worlds (formulae and machines), without dissolving boundaries across worlds or knowledge domains; (2) heterogeneous layers of the object of expertise at stake (indices, in this case); (3) disparate domains of knowledge (in this case, finance and software engineering).

Operations that achieve alignments are grounded in collaborations across domains of knowledge. In order to produce "fair" numbers, engineers and financial analysts had to engage in collaborative activities that relied on indexical, coordinated skills of seeing data in multiple scripts at once and making judgments about the data. Ethnographic observations provide no evidence that such skills were simply transferred from previous activities; rather, the evidence points to them being developed jointly in repeated, spontaneous collaborations where participants summoned each other to look at the screens. At the same time, there were skills related to seeing machines at work, and seeing numbers on machines, that remained the domain of collaborations among engineers. Finally, there were skills that, at least in some cases, pertained to engineering but were learned by financial analysts: for instance, how to connect into a machine and restart it. While some critical skills were jointly produced, this was not a mix of pre-existing skills, but rather the outcome of collaborations across different domains of expertise, the boundaries of which were blurred in practice, but not entirely removed. Other critical skills, such as handling messages and machines, remained the exclusive domain of engineers.

Engineers and analysts did indeed understand the index as a commercial object too, which was made clear not only to the ethnographer but also in conversations with clients. They understood that machine time, as well as their own working time were factored in when it came to pricing the index, and they did not hesitate to relay this to clients. A creole "cryptoindexicalese" (see Galison, 1997) was not observed on the ground, and certain terms (messages, machines) largely remained in the engineering domain. Syntactic and semantic transfers across boundaries took place, yes, but such transfers did not fully remove boundaries. As for interactional expertise (Collins & Evans, 2007), observations on the ground suggest that it was both asymmetric and limited: engineers understood more of the financial terminology than analysts understood engineering terminology, perhaps also because terms such as machines and messages were used mostly (if not exclusively) in collaborations among engineers. This being said, terms such as "real time" took a prominent place in a great deal of the conversations involving both expertise domains, perhaps also because it was of paramount importance to the index: both sides understood the importance of latency, as well as that proper "real time" could not be achieved. The argument I am making here though is that, at least in some instances, (full) interactional expertise is not needed for alignment to be successful. Key aspects of collaborations were entirely indexical and did not involve full verbalizations.

The above observations are relevant against the background of historically developed rhetorical positions according to which, as varied domains of expertise (e.g., finance, or biology) increasingly automate, they will ultimately become subordinated, if not outright absorbed (in)to the expertise of software engineering (see Ribes et al., 2019). There is little on the ground (at least in this case) that would corroborate these positions.

Another relevant trope is the "algorithms produce data"[6] mentioned above that implies that formulae can be fused with machines and this fusion will generate data. It too ultimately excludes expert operations from the generation of data. Again, observations on the ground suggest the contrary: formulae cannot generate data without complex alignment interventions, collaboratively performed in small groups.

It is said that we live in a data world, one in which data science, or expertise plays an essential role. Understanding this world, and the normative force of data (one that goes well beyond the case of market indices) requires understanding the alignment expertise that produces and expands it.

NOTES

1. I borrow the term from Erving Goffman (1986 [1974], p. 82) to designate layered framings of an activity (or object, in my case) that depend on each other and that cannot be taken apart without the activity unravelling.

2. Since the 1970s, regulatory agencies have increasingly adopted an economics-based benchmark model of fair and robust markets, coupled with utility maximization and rational choice (Jones, 2012, p. 129).

3. The thin arrows in Fig. 1 indicate the data flows between exchanges, databases, and virtual machines. The thick arrows indicate the purification interventions undertaken by analysts and engineers.

4. Level 3 data includes individual, non-aggregated order data: real time bid/ask size, quote price, size of last trades, high and low price for the day. This data level is accessed by registered market makers.

5. Natalie Heinich (2017, p. 78) notices that detected art forgeries are not destroyed but kept as a special category against which the authenticity of a painting can be evaluated.

6. A version of this argument is encapsulated in varieties of the performativity thesis, according to which formulae and models have performative power – that is, they are capable of automatically generating data that are integrated within institutions to the effect of reshaping the latter (e.g., MacKenzie et al., 2007; MacKenzie & Millo, 2003). Expertise is deployed in constructing and testing formulae, but generally formulae and machines do not need a work of alignment (but see MacKenzie, 2021).

ACKNOWLEDGMENTS

Research for this paper has been supported by the Economic and Social Research Council UK grant ES/T008237/1. I am grateful for the feedback and suggestions of the anonymous reviewers.

REFERENCES

Anteby, M., & Holm, A. L. (2021). Translating expertise across work contexts: U.S. puppeteers move from stage to screen. *American Sociological Review*, *86*(2), 310–340.

Bechky, B. (2003). Sharing meaning across occupational communities. The transformation of understanding on a production floor. *Organization Science*, *14*(3), 312–330.

Bunderson, J. S. (2003). Recognizing and utilizing expertise in work groups. A status characteristics perspective. *Administrative Science Quarterly*, *48*(4), 557–591.

Callon, M. (1998). An essay on reframing and overflowing: Economic externalities revisited by sociology. In M. Callon (Ed.), *The laws of the markets* (pp. 244–269). Blackwell.

Collins, H. (2018). Studies of expertise and experience. *Topoi*, *37*, 67–77.

Collins, H., & Evans, R. (2007). *Rethinking expertise*. The University of Chicago Press.

Collins, H., Evans, R., & Gorman, M. (2010). Trading zones and interactional expertise. In M. E. Gorman (Ed.), *Trading zones and interactional expertise. Creating new kinds of collaboration* (pp. 7–23). MIT Press.

Collins, H., Evans, R., & Weinel, M. (2017). Interactional expertise. In U. Felt, R. Fouché, C. A. Miller & L. Smith-Doerr (Eds.), *Handbook of science and technology studies* (pp. 765–792). MIT Press.

Borch, C. (2022). Machine learning and social theory. Collective machine behaviour in algorithmic trading. *European Journal of Social Theory*, *25*(4), 503–520.

Borch, C., & Min, B. H. (2023). Machine learning and social action in markets: From first- to second-generation automated trading. *Economy and Society*, *52*(1), 37–61.

Elam, M. (1999). Living dangerously with Bruno Latour in a hybrid world. *Theory, Culture & Society*, *16*(4), 1–24.

Eyal, G. (2013). For a sociology of expertise: The social origins of the autism epidemic. *American Journal of Sociology*, *118*, 863–907.

Faulkner, A., Kent, J., Geesink, I., & Fitzpatrick, D. (2006). Purity and the dangers of regenerative medicine. Regulatory innovation of human-engineered tissue technology. *Social Science and Medicine*, *63*, 2277–2288.

Galison, P. (1997). *Image and logic. A material culture of microphysics*. The University of Chicago Press.

Goffman, E. (1986). *Frame analysis. An essay on the organization of experience*. Northeastern University Press. (Original work published 1974.)

Hansen, K. B., & Borch, C. (2021). The absorption and multiplication of uncertainty in machine-learning driven finance. *British Journal of Sociology*, *72*(4), 1015–1029.

Hansen, K. B., & Borch, C. (2022). Alternative data and sentiment analysis: Prospecting non-standard data in machine learning-driven finance. *Big Data and Society*. 10.1177/20539517211070701

Hautcoeur, P.-C. (2006). *Why and how to measure stock market fluctuations? The early history of stock market indices, with special reference to the French Case*. Paris Jourdan Sciences Économiques Working Paper Nr. 10. https://shs.hal.science/halshs-00590522

Heinich, N. (2017). *Des valeurs. Une approche sociologique [Values. A sociological approach]*. Gallimard.

Jones, D. S. (2012). *Masters of the universe. Hayek, Firedman, and the birth of neoliberal politics*. Princeton University Press.

Knorr Cetina, K. (1999). *Epistemic cultures. How the sciences make knowledge*. Harvard University Press.

Knorr Cetina, K. (2003). From pipes to scopes. The flow architecture of financial markets. Distinktion. *Scandinavian Journal of Social Theory*, *4*, 7–23.

Knorr Cetina, K., & Bruegger, U. (2002). Global microstructures. The virtual societies of financial markets. *American Journal of Sociology*, *107*(4), 905–950.

Latour, B. (1988). *The pasteurization of France*. Harvard University Press.

Leigh Star, S., & Griesemer, J. (1989). Institutional ecology, 'translations,' and boundary objects. Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. *Social Studies of Science*, *19*(3), 387–420.

Lepinay, V. (2011). *Codes of finance. Engineering derivatives in a global bank*. Princeton University Press.

Li, T. M. (2007). Practices of assemblage and community forest management. *Economy & Society*, *36*(2), 263–293.

MacKenzie, D. (2009). *Material markets. How economic agents are constructed*. Oxford University Press.

MacKenzie, D. (2021). *Trading at the speed of light. How ultrafast algorithms are transforming financial markets*. Princeton University Press.

MacKenzie, D., & Millo, Y. (2003). Constructing a market, performing a theory. The historical sociology of a financial derivatives exchange. *American Journal of Sociology*, *109*(1), 107–145.

MacKenzie, D., Muniesa, F., & Siu, L. (Eds.). (2007). *Do economists make markets? On the performativity of economics*. Princeton University Press.

Nicolini, D., Mengis, J., & Swan, J. (2012). Understanding the role of objects in cross-disciplinary collaboration. *Organization Science*, *23*(3), 612–629.

Pakarinen, P. (2024). *Experts and machines. How central bank economists produce inferences via simulation model*. Unpublished manuscript.

Pakarinen, P., & Huising, R. (2023). Relational expertise. What machines can't know. *Journal of Management Studies*. http://doi.org/10.1111/joms.12915

Pardo-Guerra, J. P. (2019). *Automating finance. Infrastructures, engineers, and the making of electronic markets*. Cambridge University Press.

Perold, A. F. (2004). The capital asset pricing model. *Journal of Economic Perspectives*, *18*(3), 3–24.

Porter, T. (1995). *Trust in numbers. The Pursuit of objectivity in science and public life*. Princeton University Press.

Ribes, D., Hoffman, A. S., Slota, S., & Bowkeer, G. C. (2019). The logic of domains. *Social Studies of Science*, *49*(3), 281–309.

Van der Voort, H., van Bulderen, S., Cunningham, S., & Janssen, M. (2021). Data science as knowledge creation. A framework for synergies between data analysts and knowledge professionals. *Technological Forecasting and Social Change*, *173*, 1–10.

Yonay, Y., & Breslau, D. (2006). Marketing models: The culture of mathematical economics. *Sociological Forum*, *21*(3), 345–386.