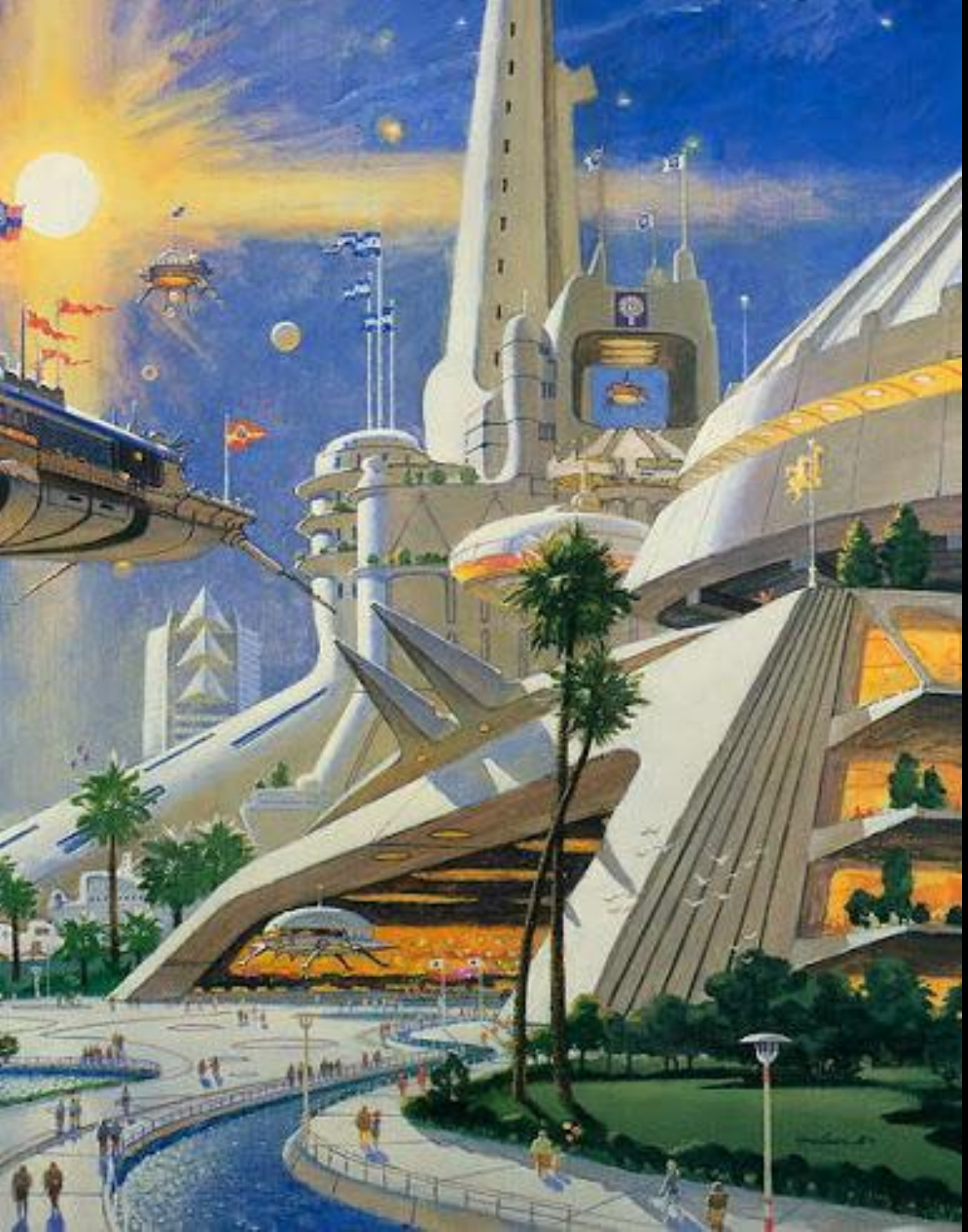
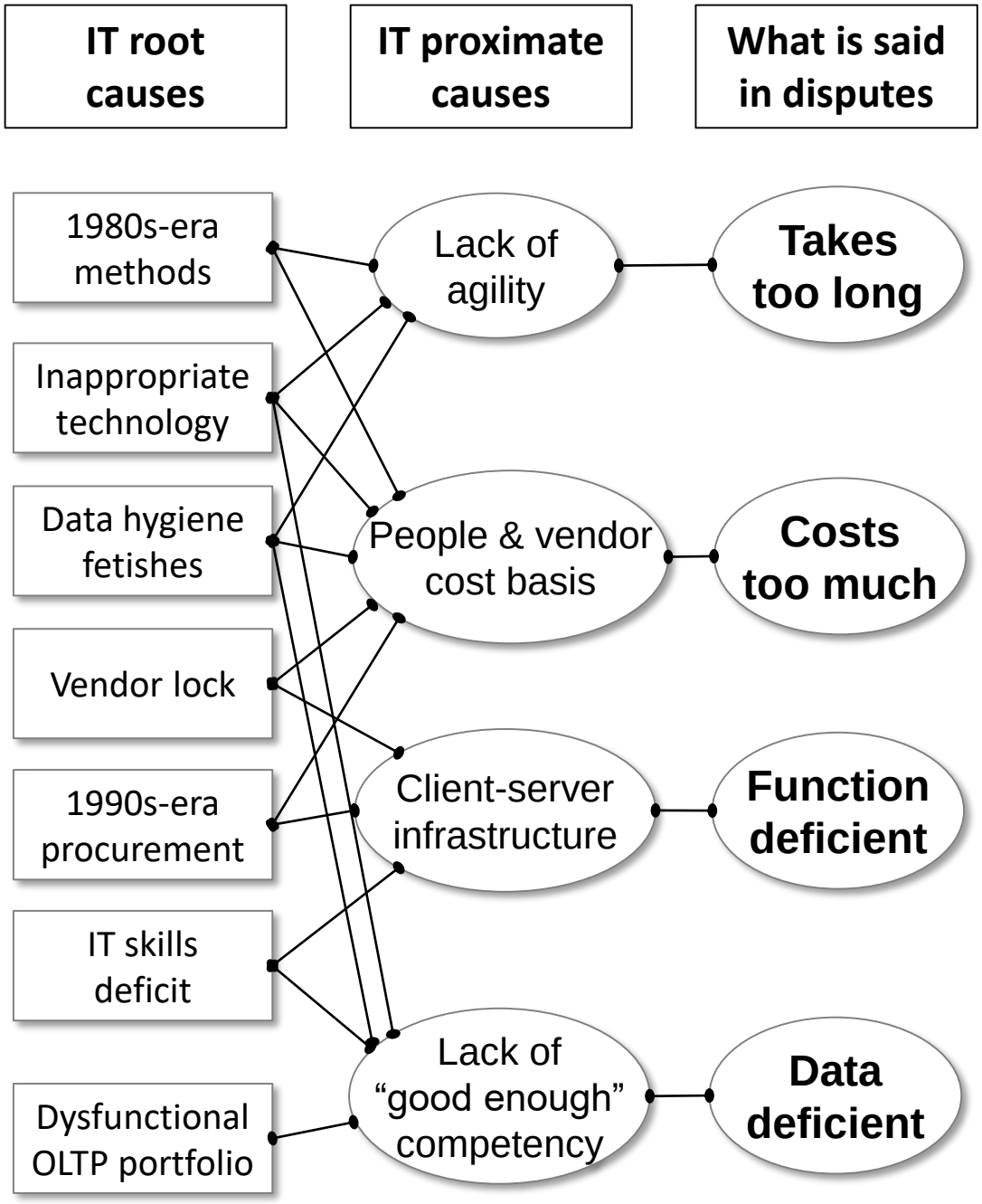


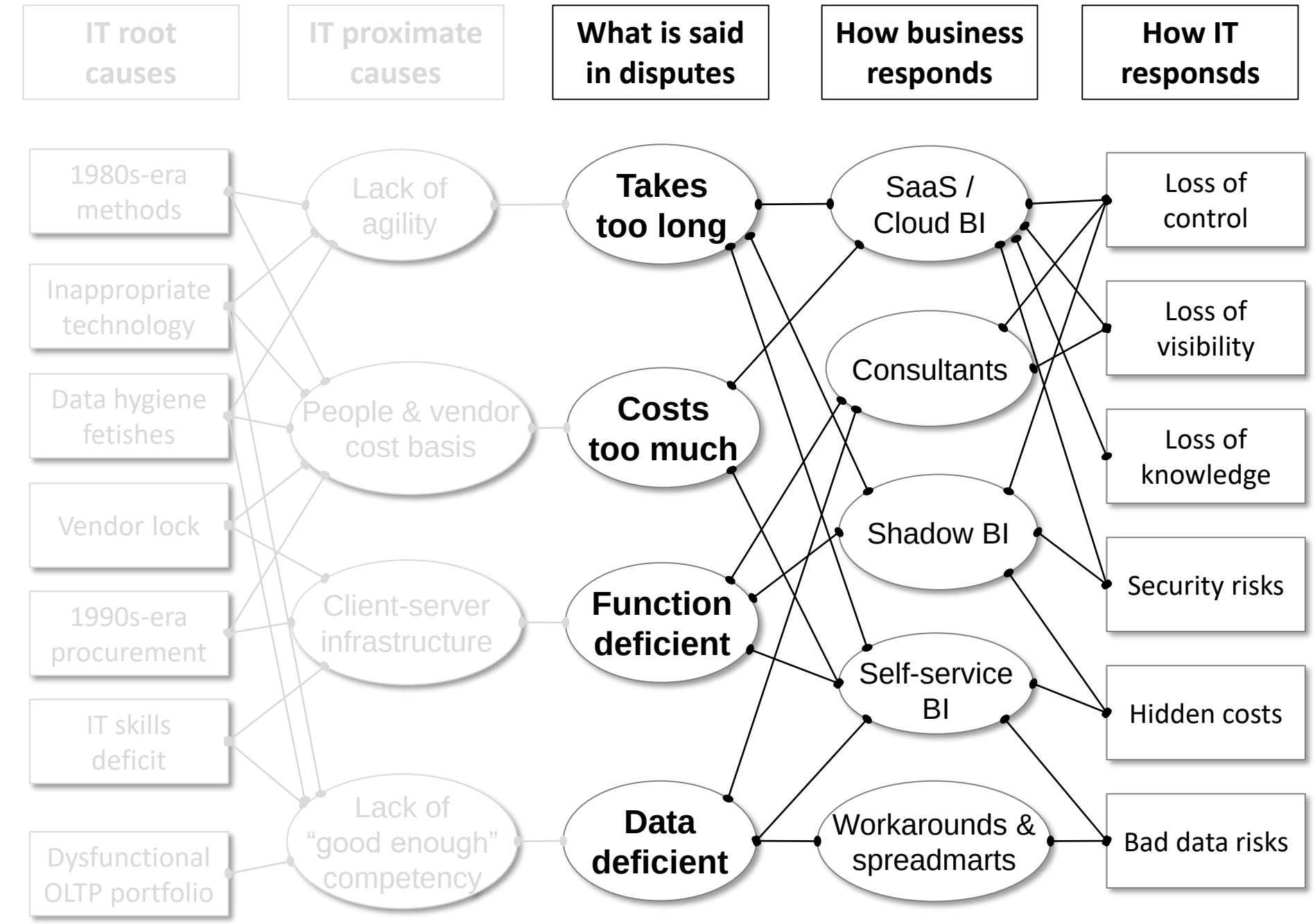
Architecting a Data Platform For Enterprise Use

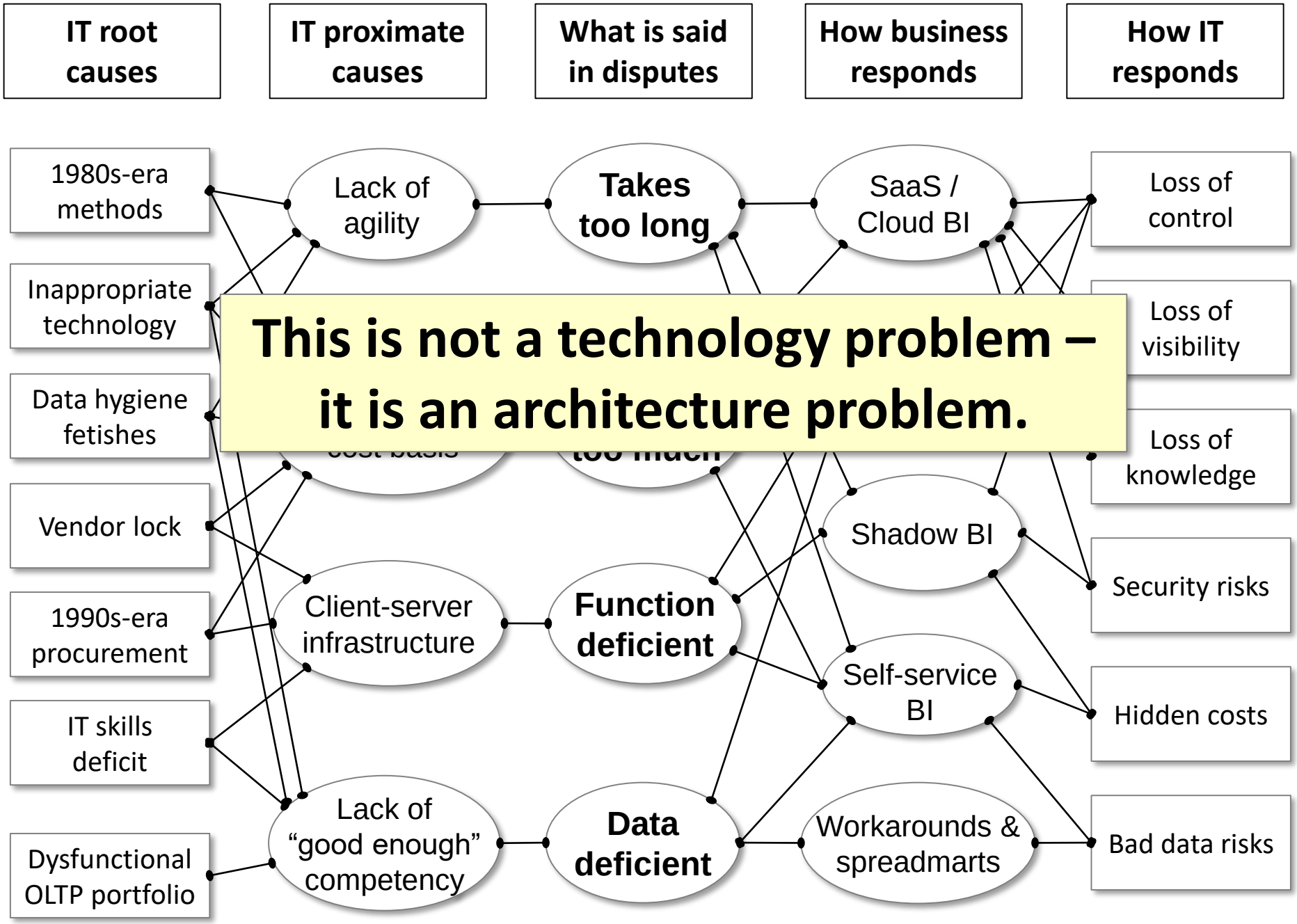
May 2018

Mark Madsen
Todd Walter

The business complaints about data and analytics







What do we hear in the field?

"I want to do analytics, beyond what I can do with BI tools"

"I need an analytics strategy"

"I need an analytics roadmap"

"I have a specific analytics project of type..."

“We have a data lake. What should we do with it?”

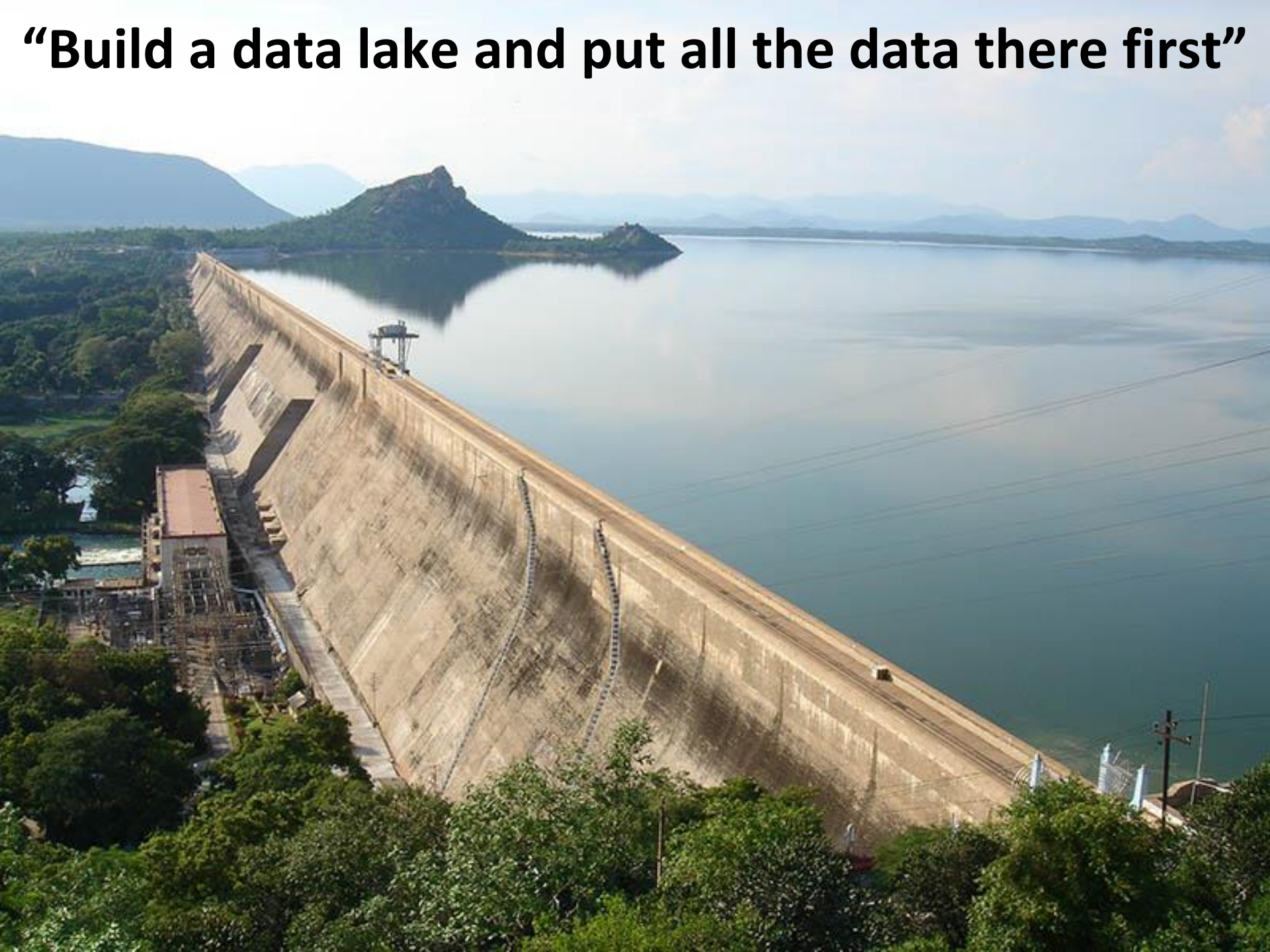
“Technology Y replaces technology X”

"I want to modernize my DW" aka "I want to speed up my DW"

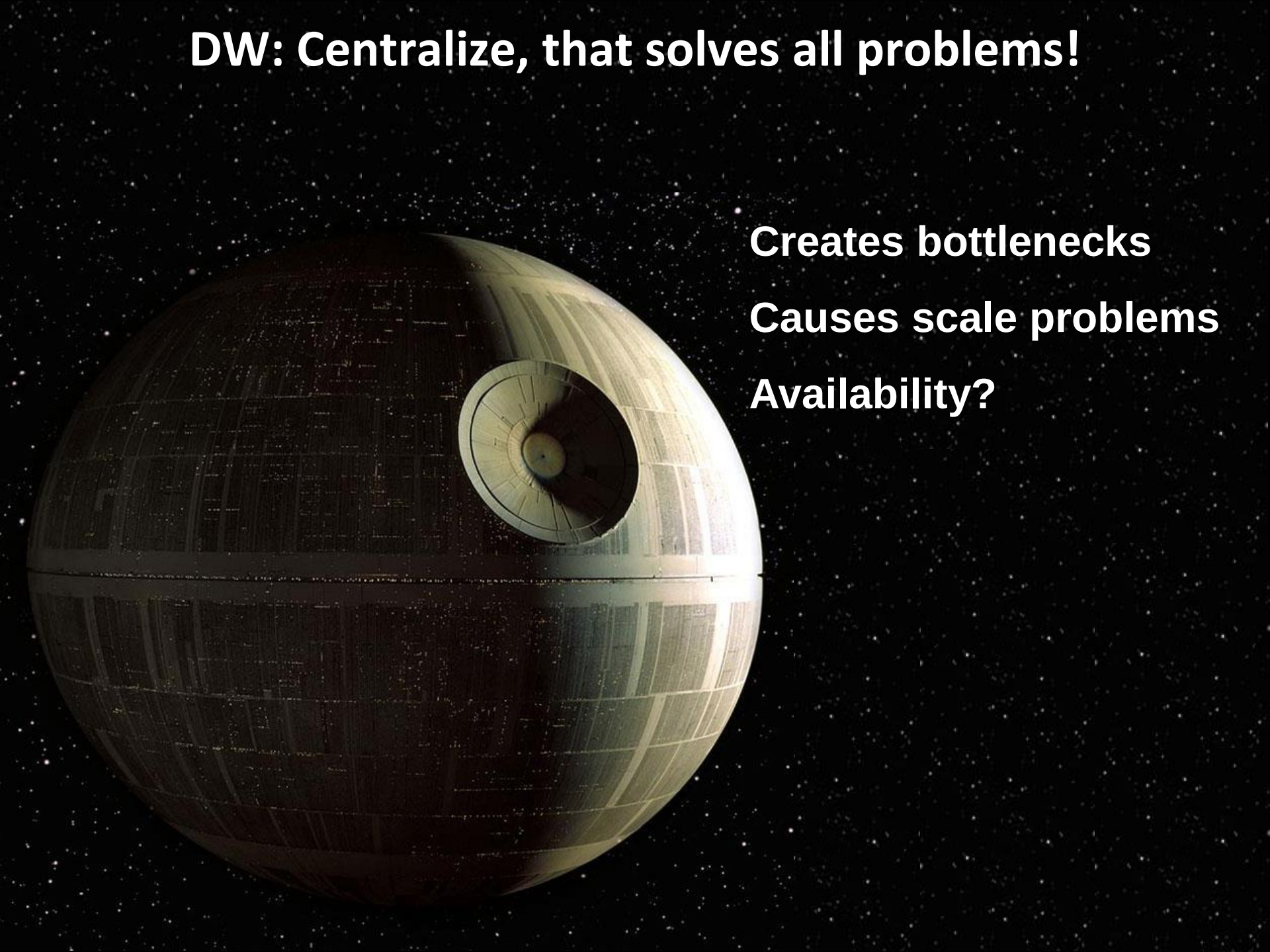
“Don’t need schema, curation, ETL, governance … anymore”

Good judgement is the result of experience.
Experience is the result of bad judgement.
—Fred Brooks

“Build a data lake and put all the data there first”



DW: Centralize, that solves all problems!

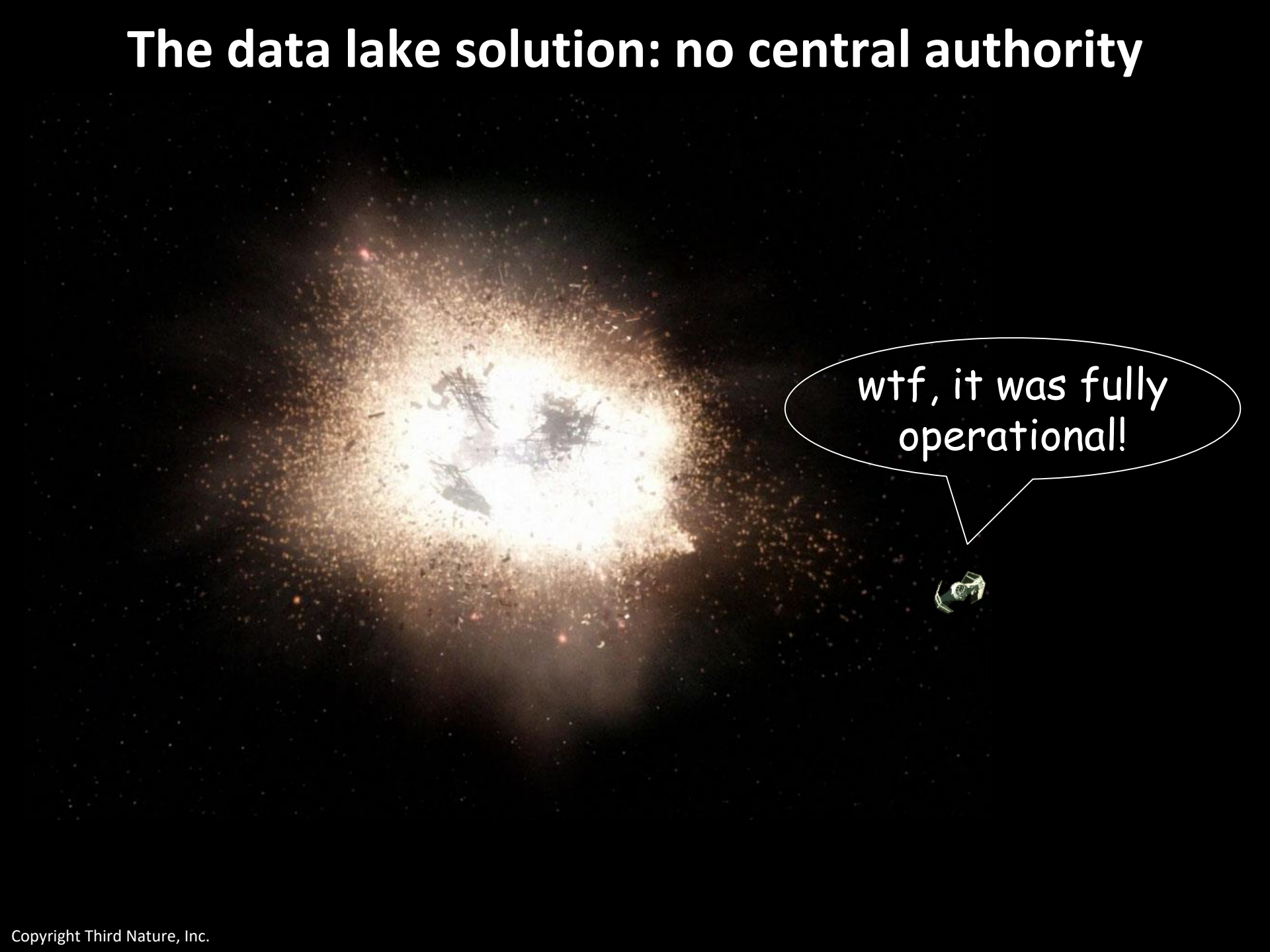


Creates bottlenecks

Causes scale problems

Availability?

The data lake solution: no central authority

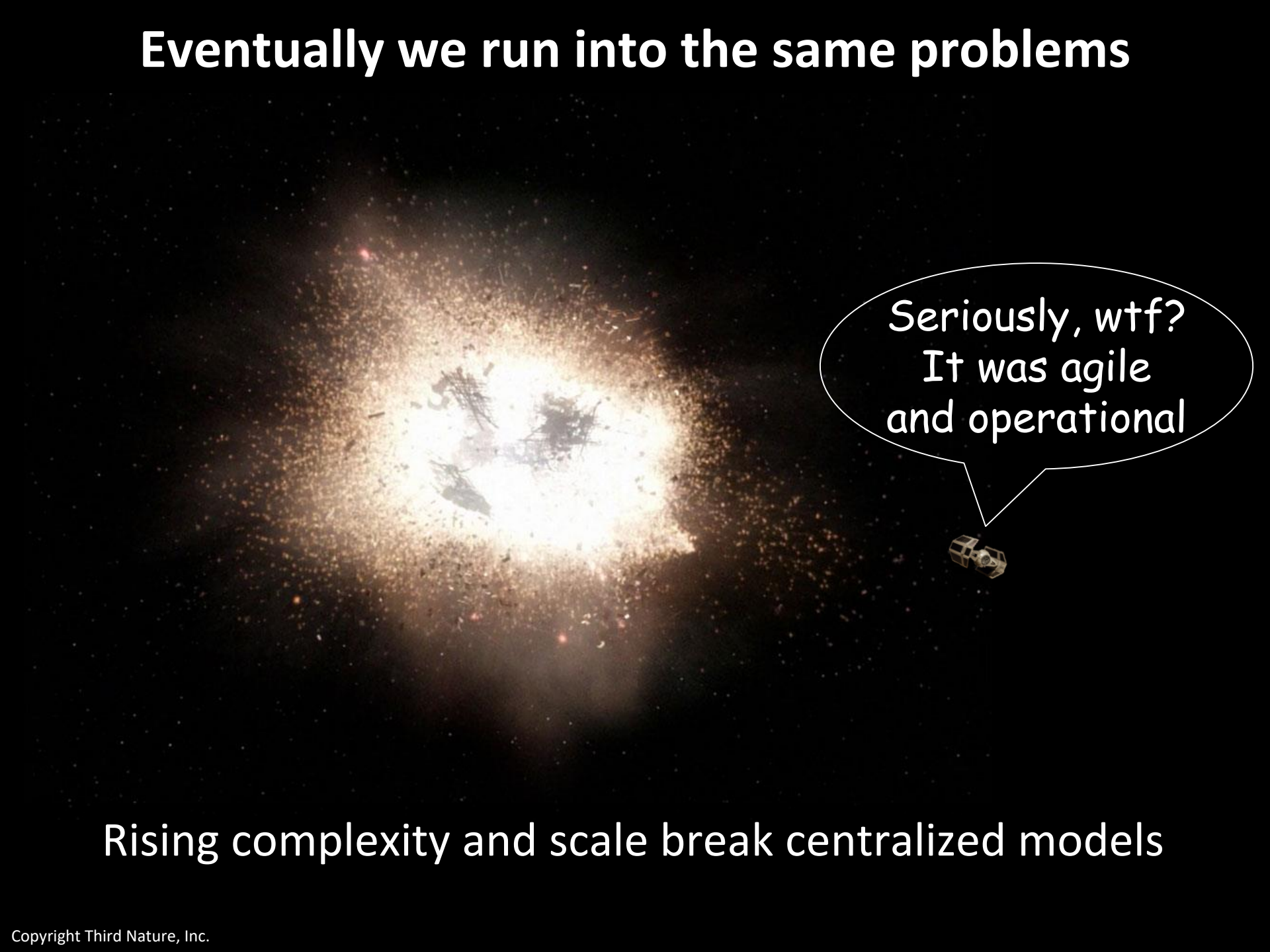


The data lake solution?

There's a problem: as the lake is envisioned, it is still a centralized data architecture, but this time there is no single global model. Instead it's files and not modeled. It can be operational while under construction.

It's still a death star.

Eventually we run into the same problems



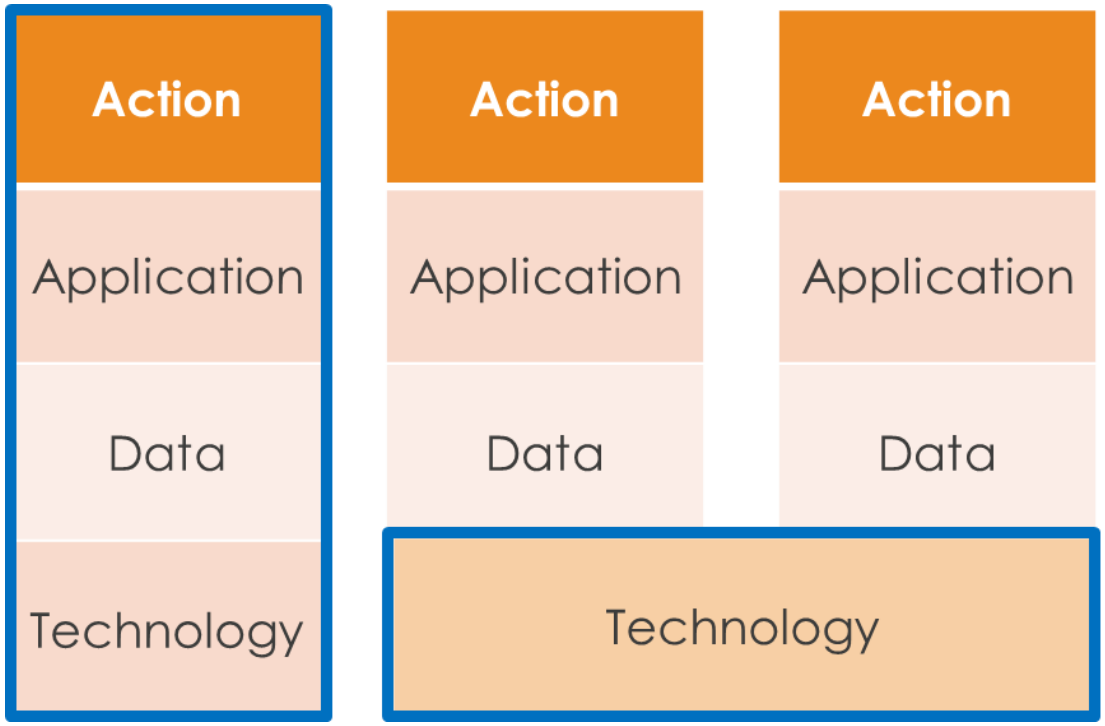
Rising complexity and scale break centralized models

“Standardize on one tech, it’s simpler for everyone”

It’s called stack think. Pick your vendor. Cede all architecture.

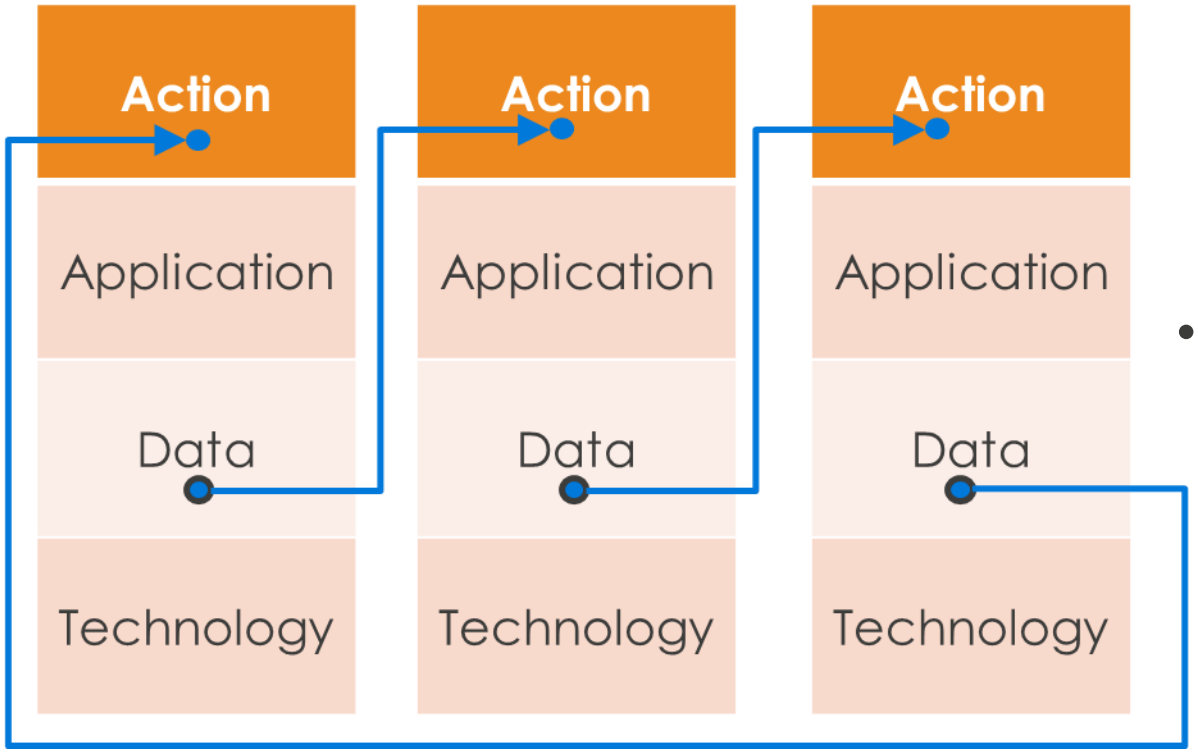
Agile your way into the Analytics Shantytown

“Infrastructure one PoC at a time”



- Use case driven
- Leverage (any) new technology
- Re-use of the technology stack
- Data is captive to the application or to the technology stack
- Developers tend to think “function first”
- Analytics people also think “function first”, but generally require integrated data

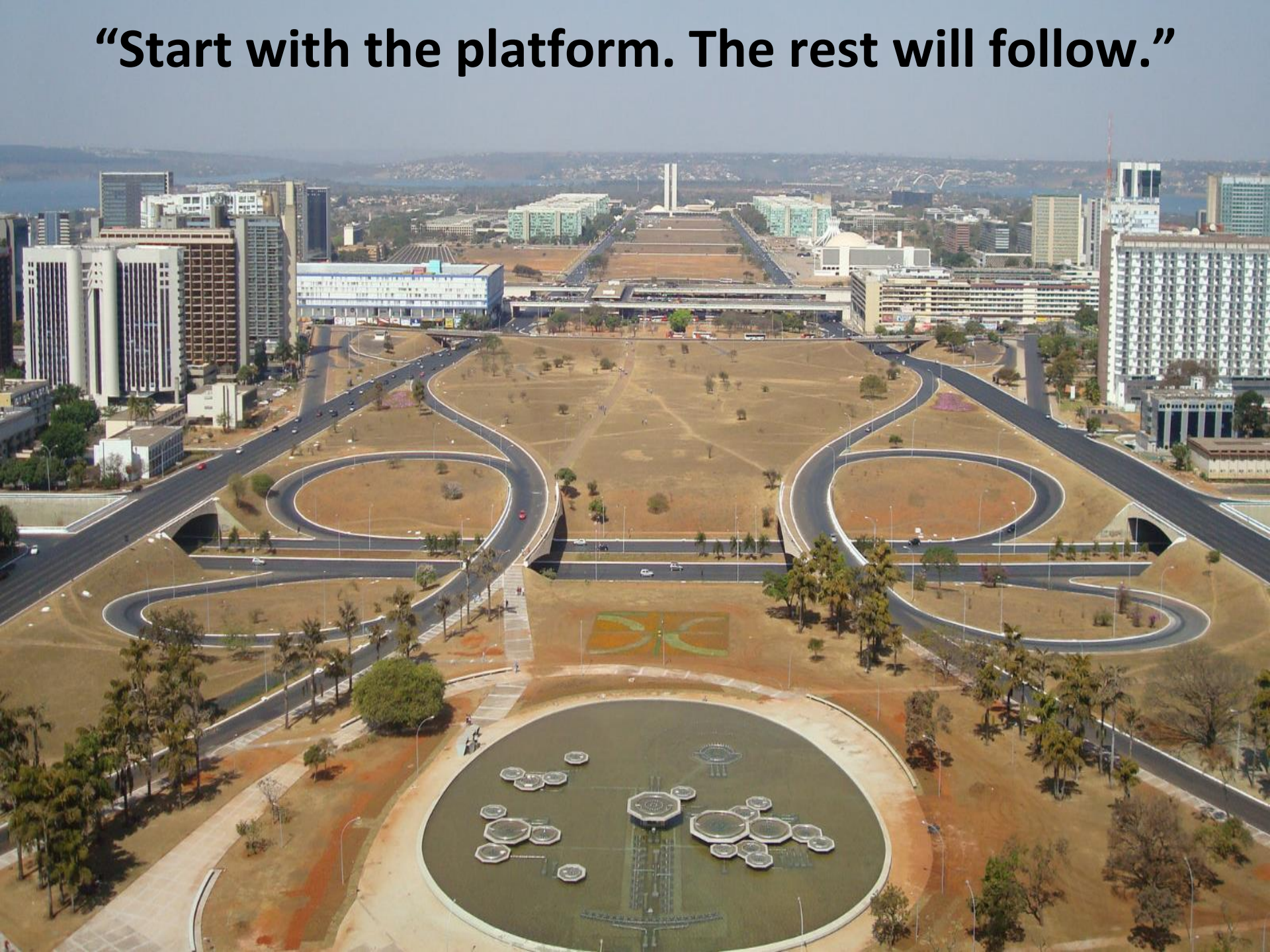
A key aspect of operational analytics is “end-to-end”



- Most data applications are organized around a process, not a task or function.
- Real-time analytics systems pass their data over the network, but the dependencies can cross application boundaries, as well as requiring persistence.
- Developers are being forced to think of data outside the local context.

E2E does not allow you to take a random walk through technologies and BYOT because the complexity of integration leads to a mountain of technical debt.

“Start with the platform. The rest will follow.”



Big data promise: What you see

Big data reality: What users see


Persisting data is not the end of the line.
If you stop here you win the battle and lose the war

We don't have an analytics problem, just like we didn't have a BI problem

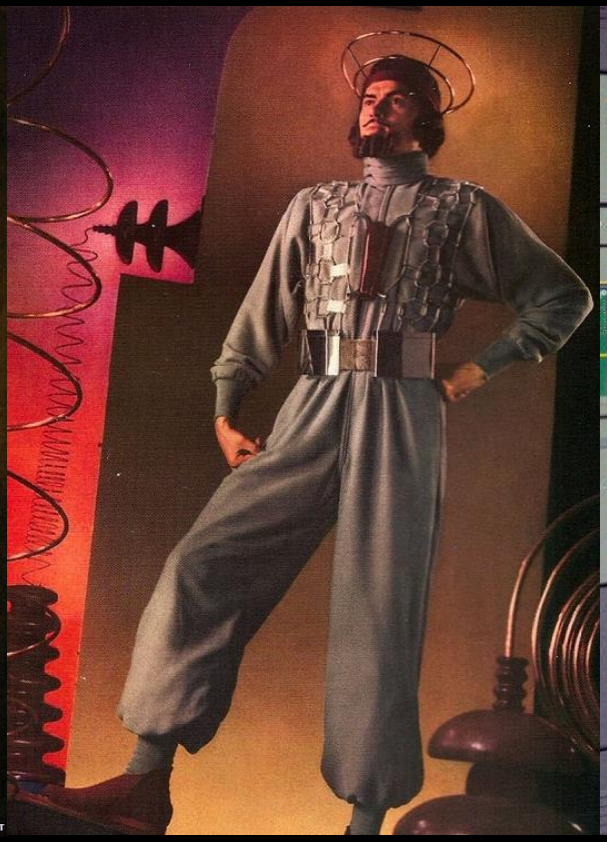
The origin of analytics as "business intelligence" was stated well in 1958:

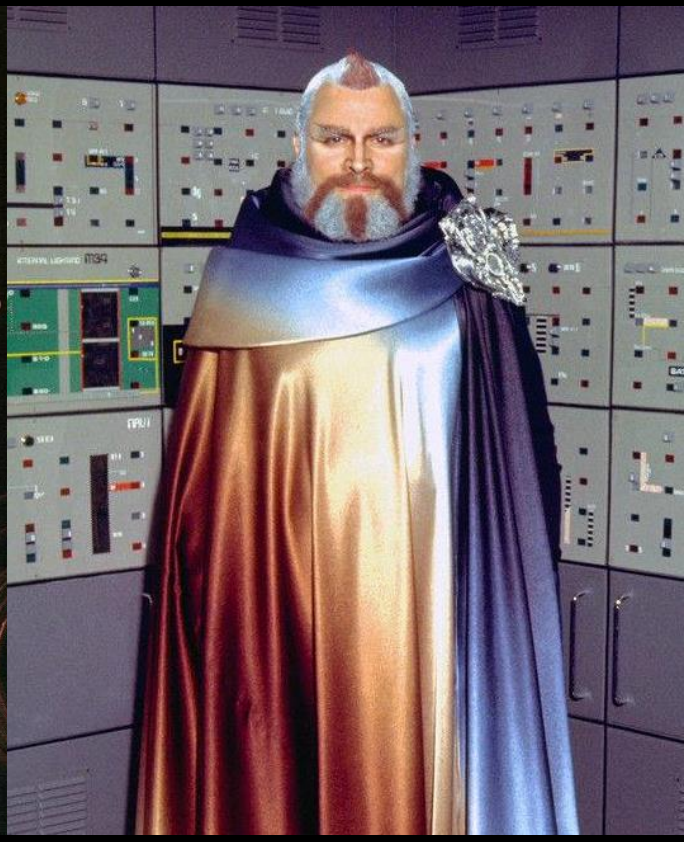
> "…the ability to apprehend the interrelationships of presented facts in such a way as to guide action towards a desired goal." ~ *H. P. Luhn*

Our goal is analytics as a capability, not a technology

"A Business Intelligence System", http://altaplana.com/ibmrd0204H.pdf

Three constituencies





Stakeholder

aka the recipient

Analyst

aka the data scientist

Builder

aka the engineer

Starting points for analytics strategy

Many organizations choose to start with the analysts. Create a data science team. Turn them loose to find a problem.

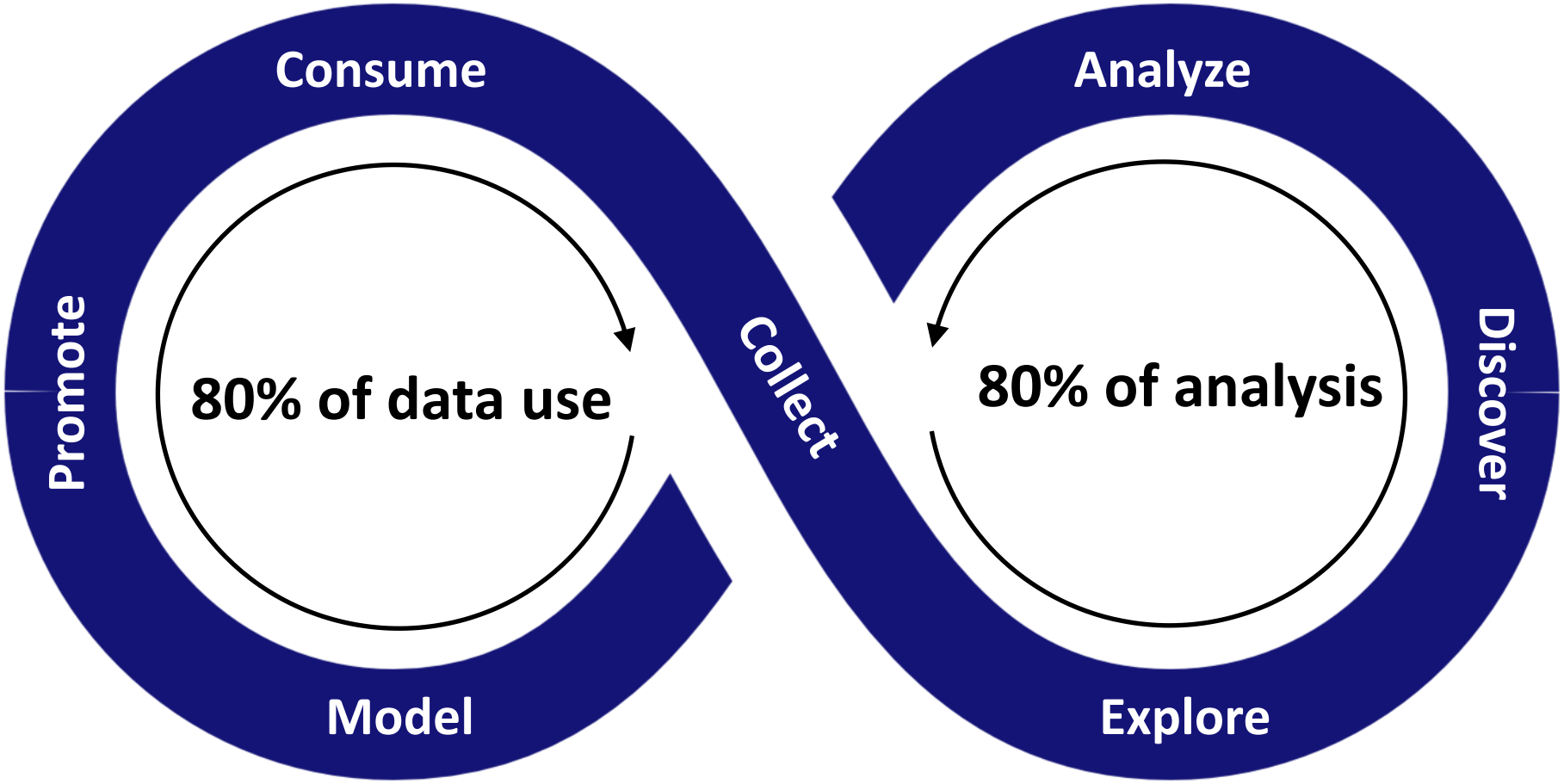
Many more start with builders: technology solutions looking for problems, e.g. 65% of the IT driven Hadoop and Spark projects over the last five years.

The right place to start? Stakeholders. The goal to achieve, the problem to solve.

Each constituency has their own set of problems to deal with

NATURE OF THE PROBLEM FROM THE STAKEHOLDER'S PERSPECTIVE

BI and Analysis: Repeatability and Discoverability



Focus on repeatability

Application cycle time

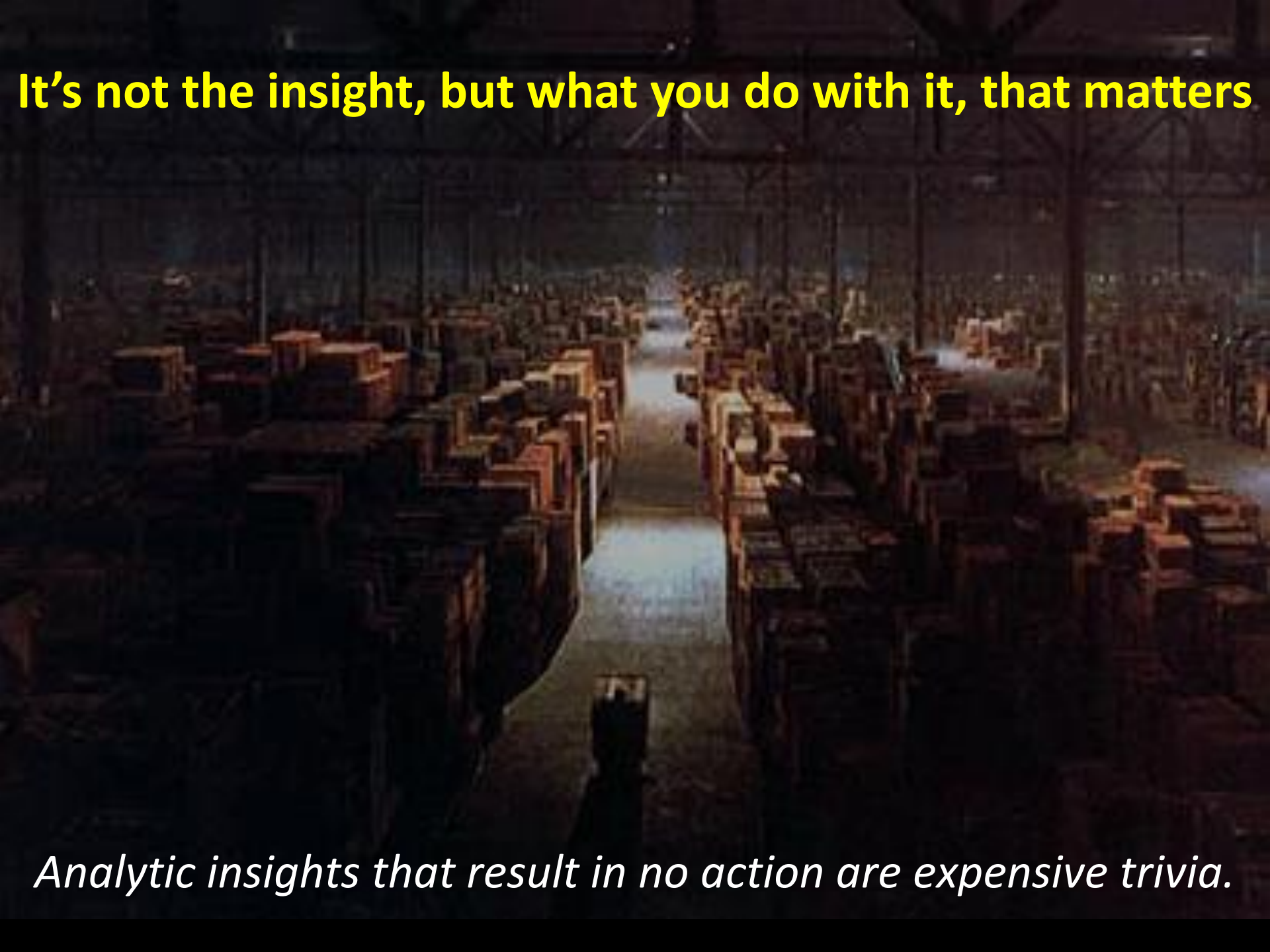
90% of users

Focus on discoverability

Analyst cycle time

9% analysts, 1% data scientists

The myth that still drives analytics – analytic gold


It’s not the insight, but what you do with it, that matters
Analytic insights that result in no action are expensive trivia.

Applying analytics is not an analytics problem

Applying analytics is not in the analyst's control.

It's not in the engineer's control.

It's in the control of the people involved in the process.

Failures are often in execution, not in analytics development.

For example, we saw unexpectedly poor performance in a number of geographies. Was it the new analytics we tried? Was it a data problem? No, it was a simple compliance problem.

NATURE OF THE PROBLEM FROM THE ANALYST'S PERSPECTIVE

The analytics process at a high level

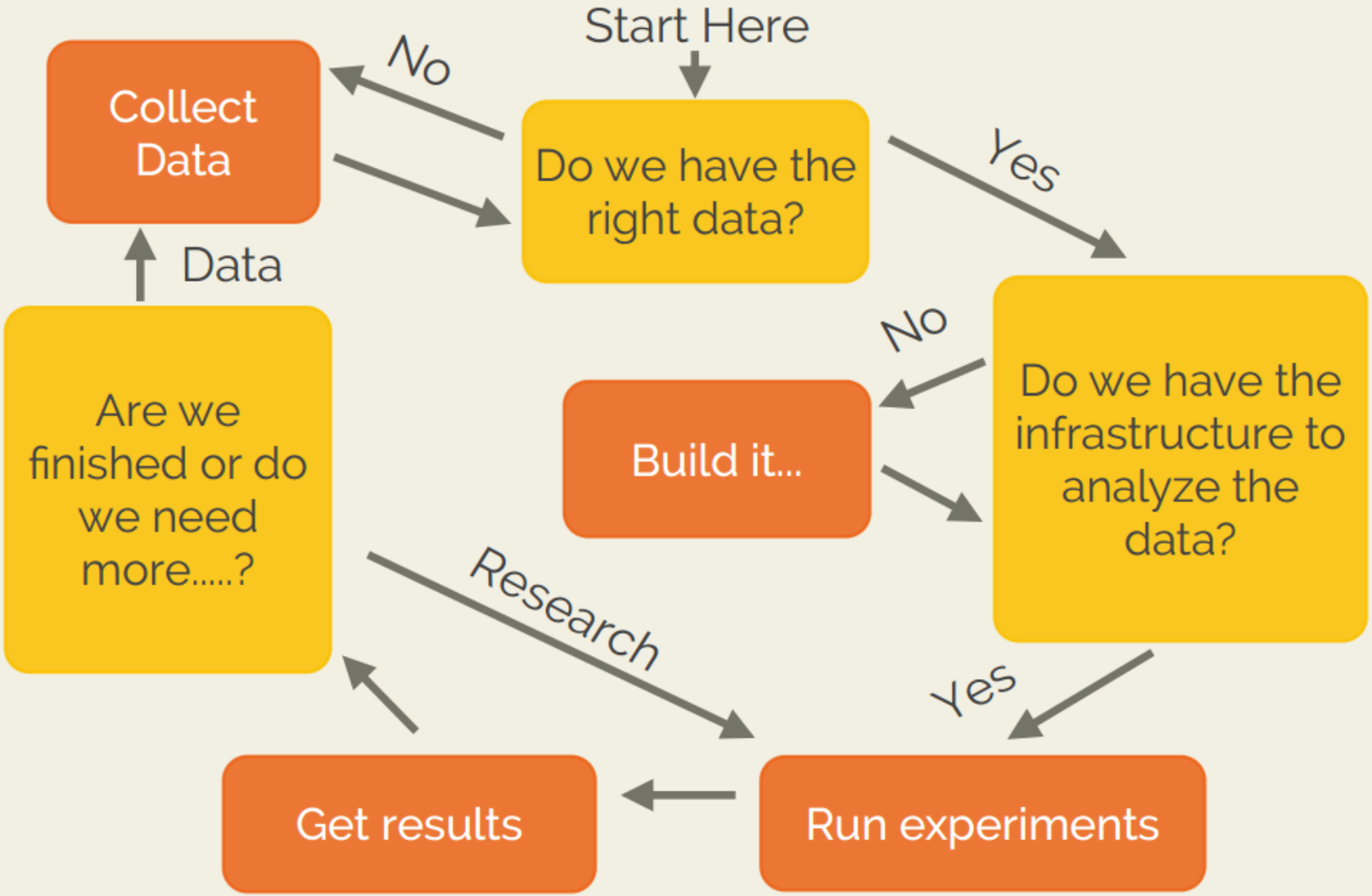
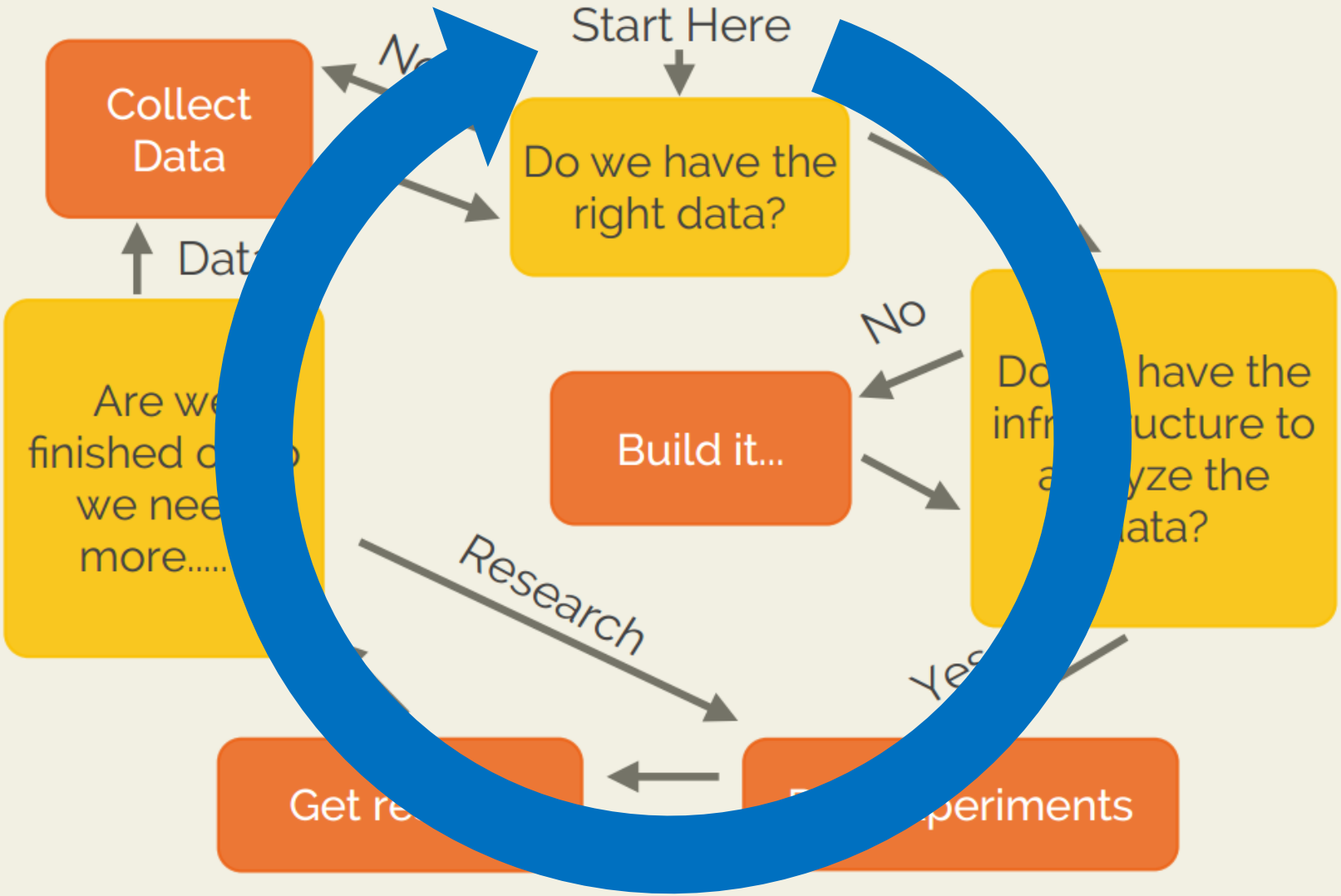


Diagram: Kate Matsudaira

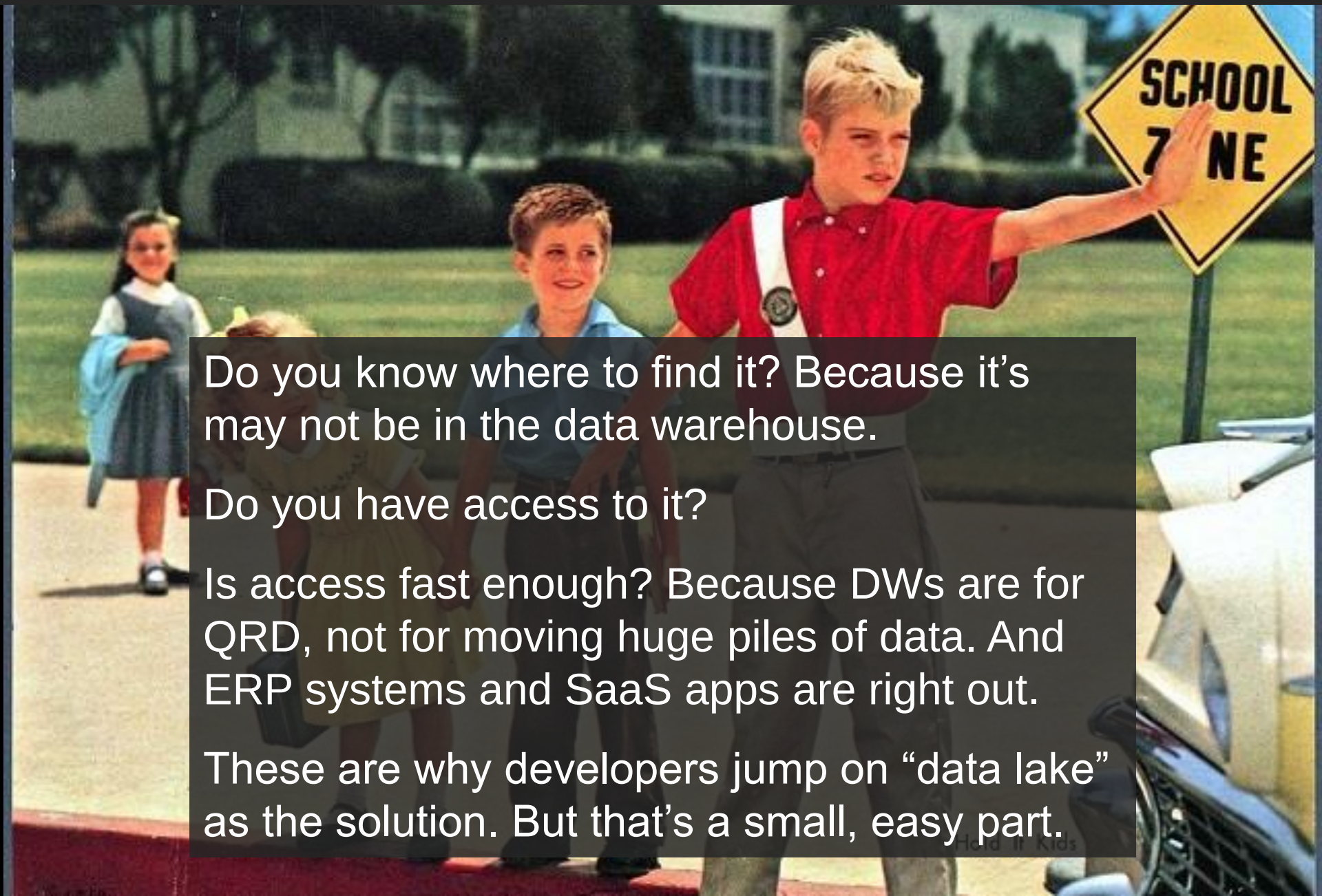
Repeat for each new problem



The nature of analytics problems is researching the unknown rather than accessing the known.

Diagram: Kate Matsudaira

The main hurdle: just getting the data



Do you know where to find it? Because it's may not be in the data warehouse.

Do you have access to it?

Is access fast enough? Because DWs are for QRD, not for moving huge piles of data. And ERP systems and SaaS apps are right out.

These are why developers jump on "data lake" as the solution. But that's a small, easy part.

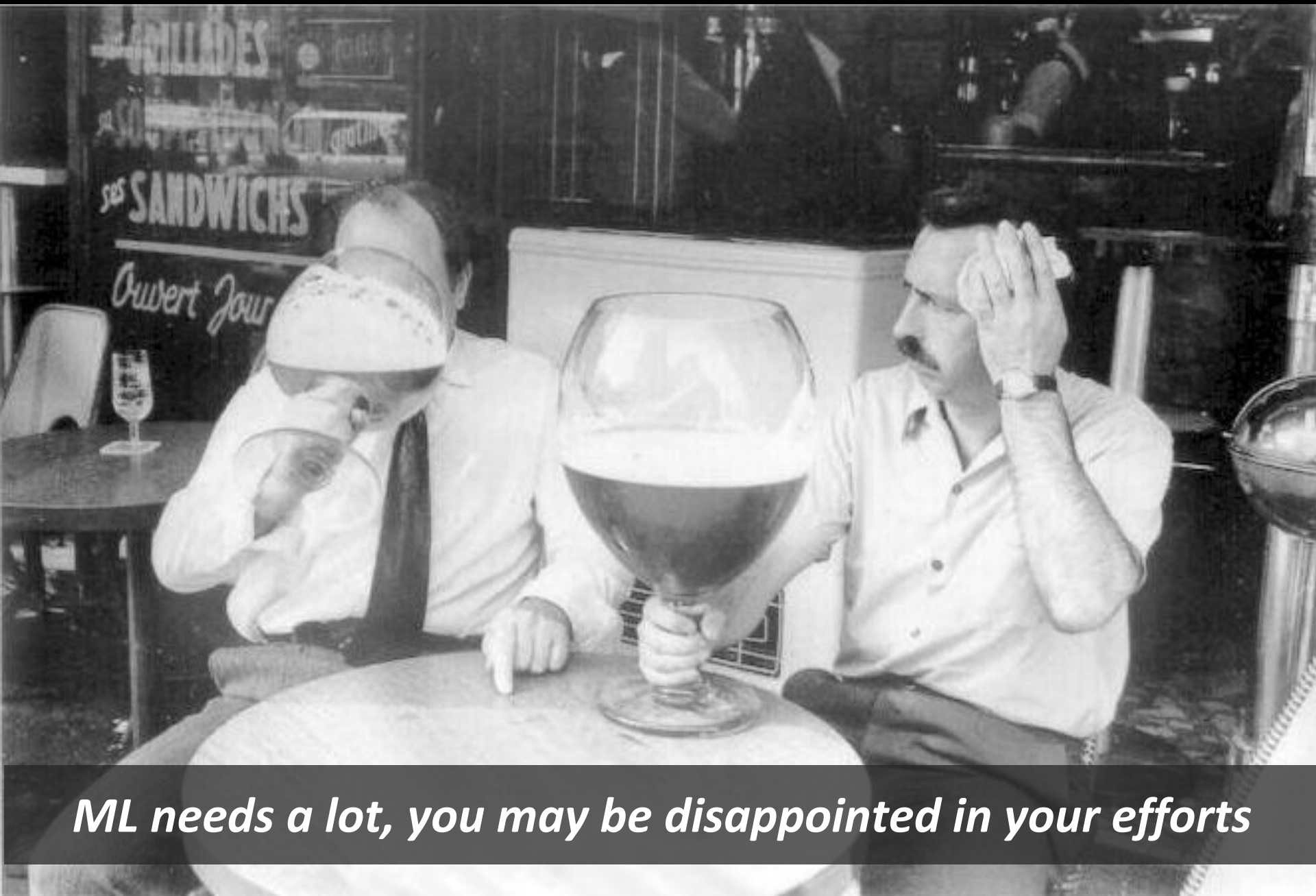
Do you have the right data?

Many machine learning techniques require labeled (known good) training data:

Supervised learning: a person has to define the correct output for some portion of the data. Data is divided into training sets used for model building and test sets for validating the results.

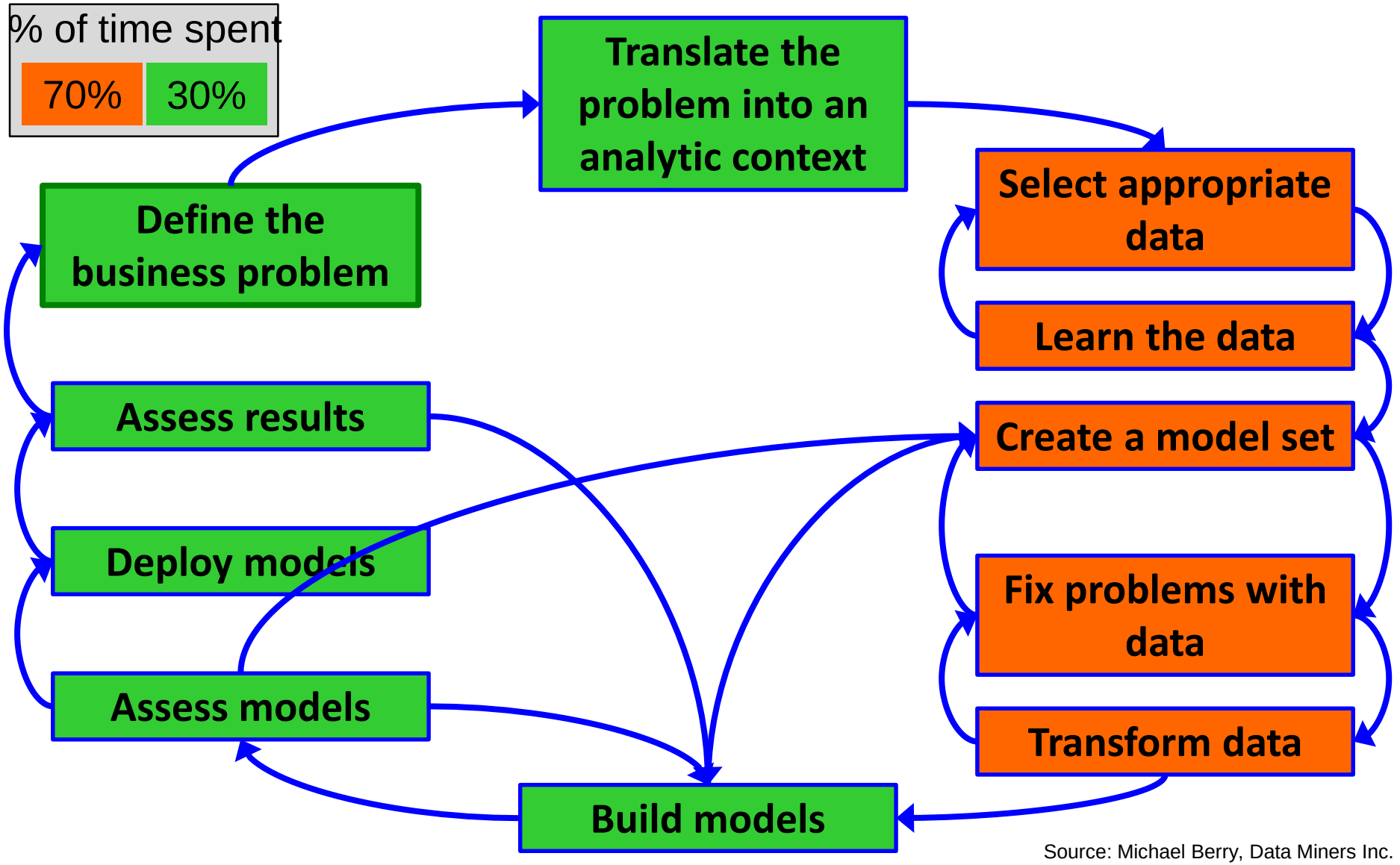
- *What is spam and what isn't?*
- *What does a fraudulent transaction look like*

Do you have enough of the right data?



ML needs a lot, you may be disappointed in your efforts

Where do analysts spend their time? *mostly data work*



Source: Michael Berry, Data Miners Inc.

The analyst's workspace in BI is relatively spare

The analyst's workspace needs to be more like a kitchen than like BI vending machines

Analytics work is a production workload, but…

Schema control, space limitations, resource limitations, production lockdown

NATURE OF THE PROBLEM FROM THE BUILDER'S PERSPECTIVE

IT and Ops people want to know “what to build?”

Giant data platform?

Self service tools?

Analytics has different processes and workloads

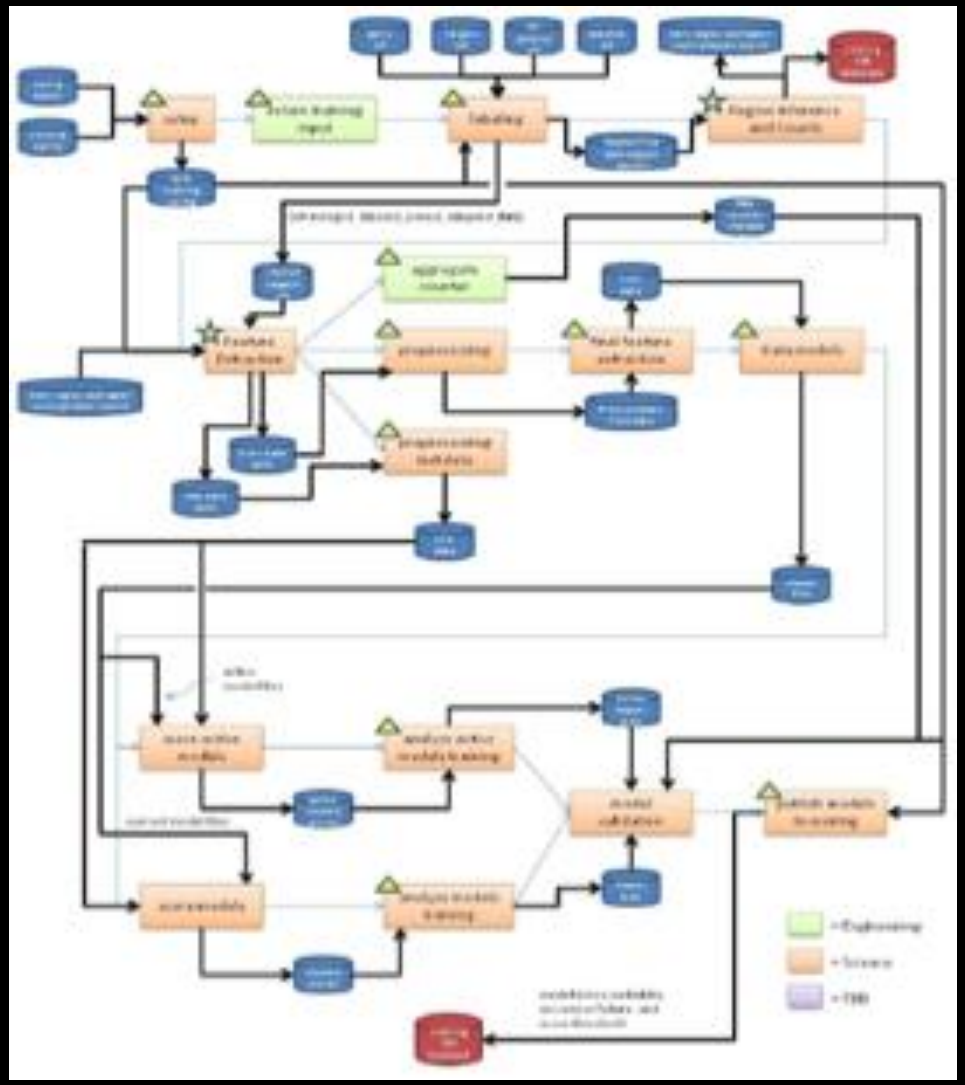
None of this analytics work is the same as what IT considered "analysis" to be, which is usually equated with BI or ad-hoc query.

Ad-hoc analysis ≠

Exploratory data analysis ≠

Batch analytics ≠

Real-time analytics



A real analytics production workflow

Hatch, CIKM '11

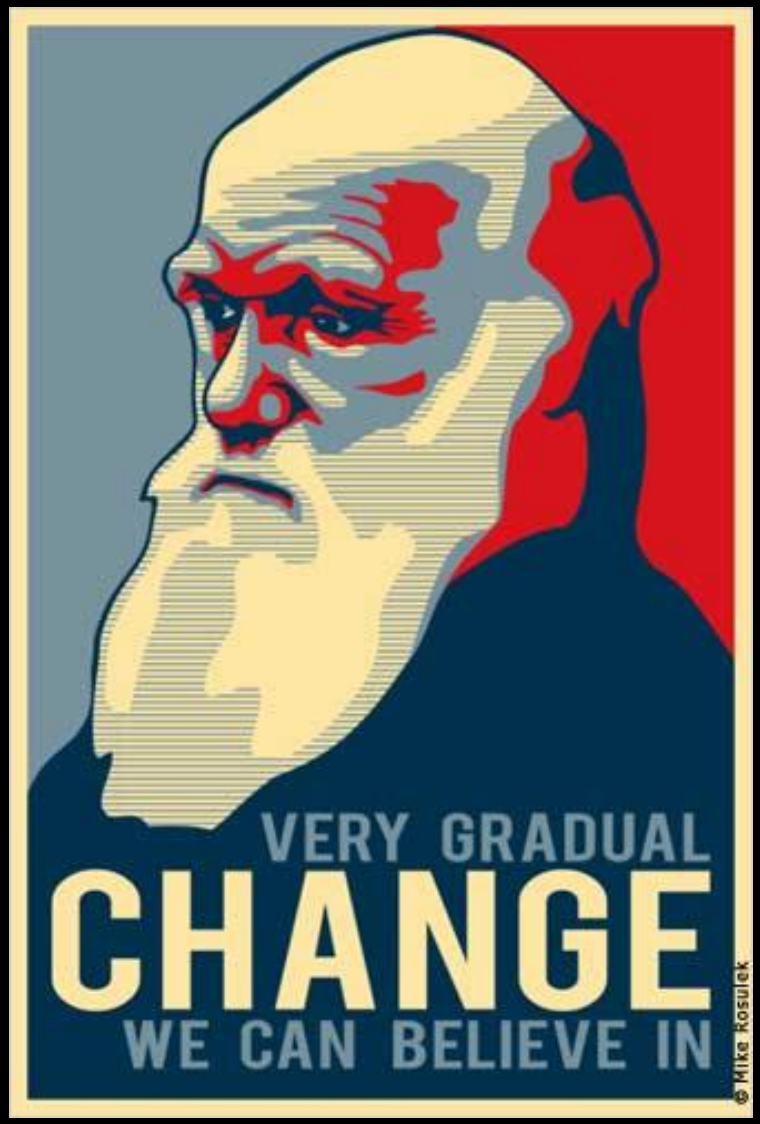
The world changes, do the models?

In BI you maintain ETL and schemas, in ML you maintain models.

"Model decay" happens as the assumptions around which a model is built change, e.g. spam techniques change.

When you adjust the model you need to know it is normal again

- Better save the data used to build the model
- Better save the model
- Baseline and measurements



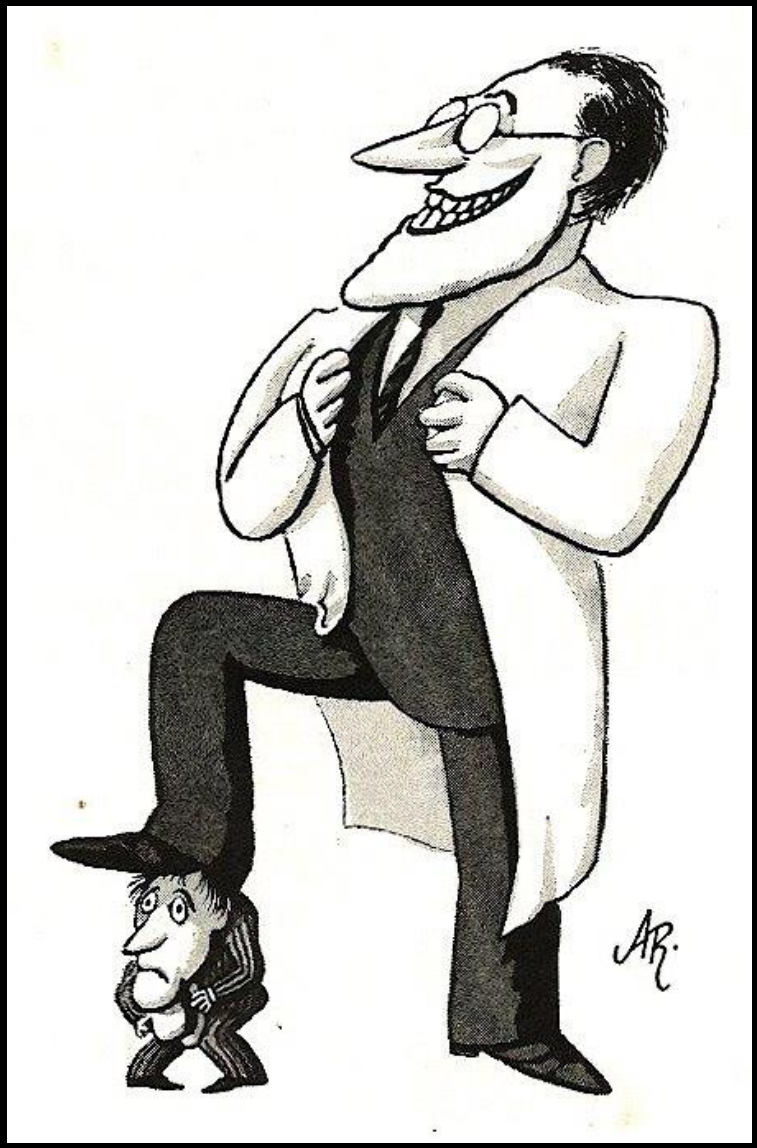
There are requirements from all constituents. You need to put them together to have a complete picture of what's needed.

THREE PERSPECTIVES, ONE SOLUTION?

The missing stakeholder

There is another stakeholder: analytics management - the CAO, CDO, VP of analytics, aka “your boss” if you’re a data scientist.

This is the perspective and problems of the person responsible for oversight of the team and efforts is across the organization and across multiple projects



Job #1 - Repeatability

Job # 2 - Operational predictability

Job #3 - Reproducibility

You need a system of record for analytics

There is an extensive list of requirements to support

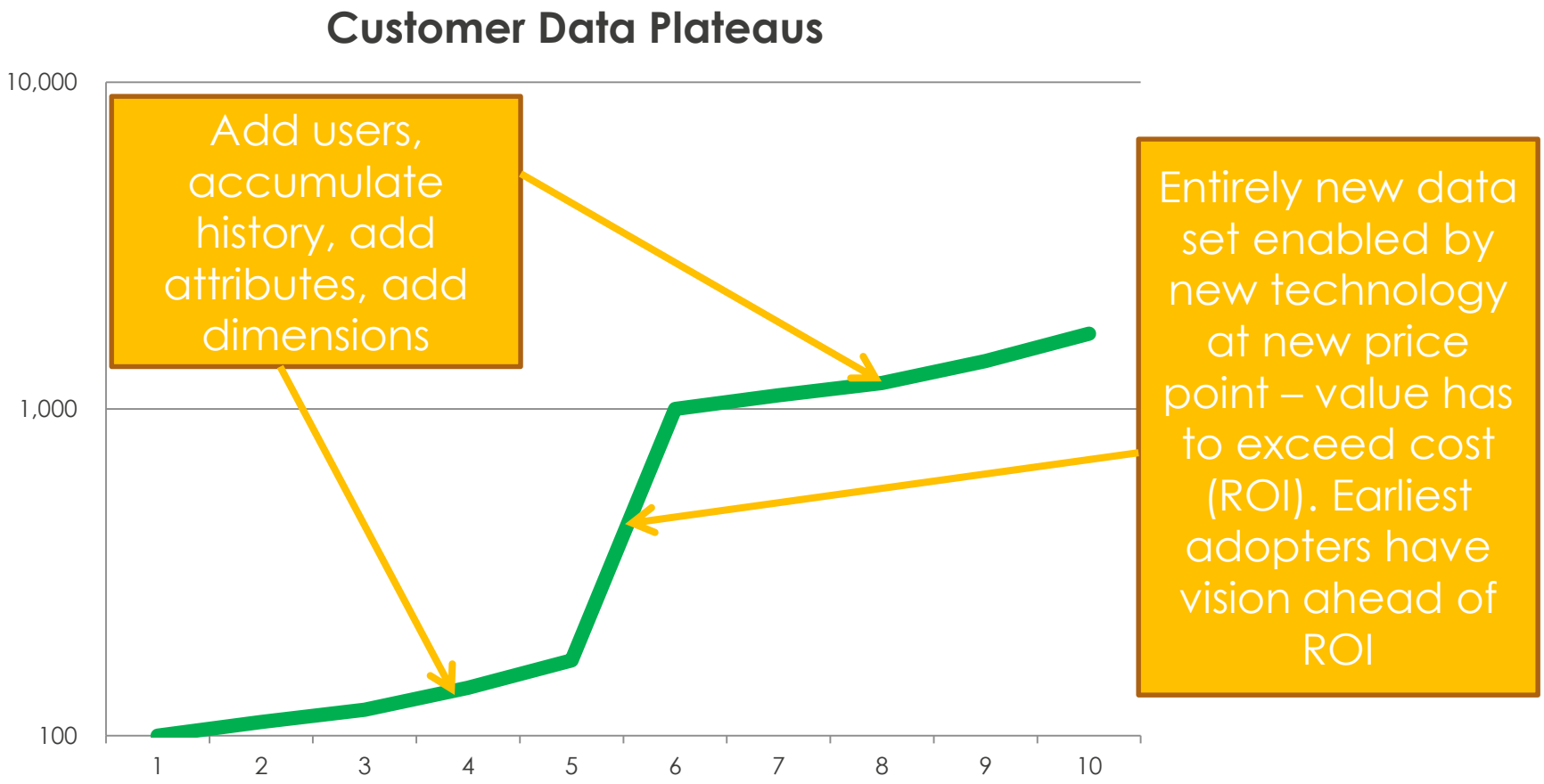
Primary requirements needed by constituents	S	D	E
Data catalog and ability to search it for datasets		X	X
Self-service access to curated data		X	
Self-service access to uncurated (unknown, new) data		X	X
Temporary storage for working with data		X	
Data integration, cleaning, transformation, preparation tools and environment		X	X
Persistent storage for source data used by production models		X	X
Persistent storage for training, testing, production data used by models		X	X
Storage and management of models		X	X
Deployment, monitoring, decommissioning models			X
Lineage, traceability of changes made for data used by models		X	X
Lineage, traceability for model changes	X	X	X
Managing baseline data / metrics for comparing model performance	X	X	X
Managing ongoing data / metrics for tracking ongoing model performance	X	X	X

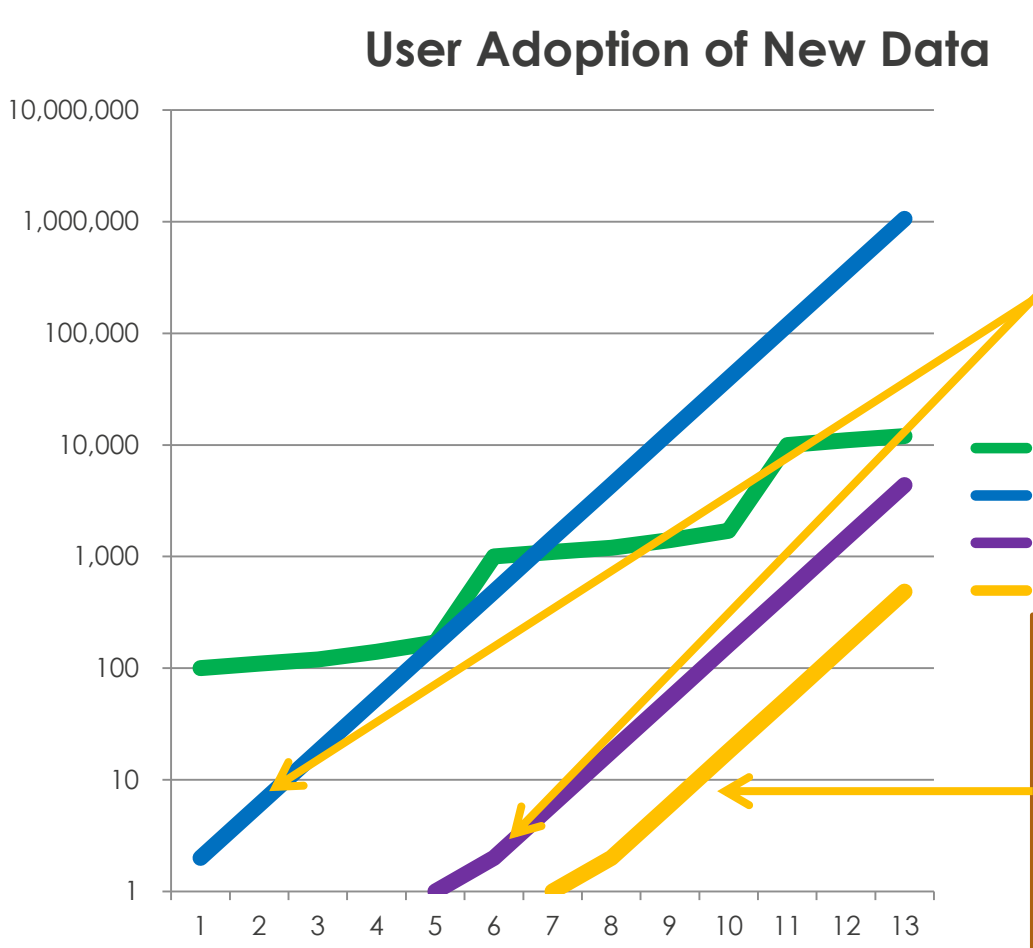
S = stakeholder, user, D = data scientist, analyst, E = engineer, developer

How did we get to this point with BI & analytics?

There's a difference between having no past and actively rejecting it.

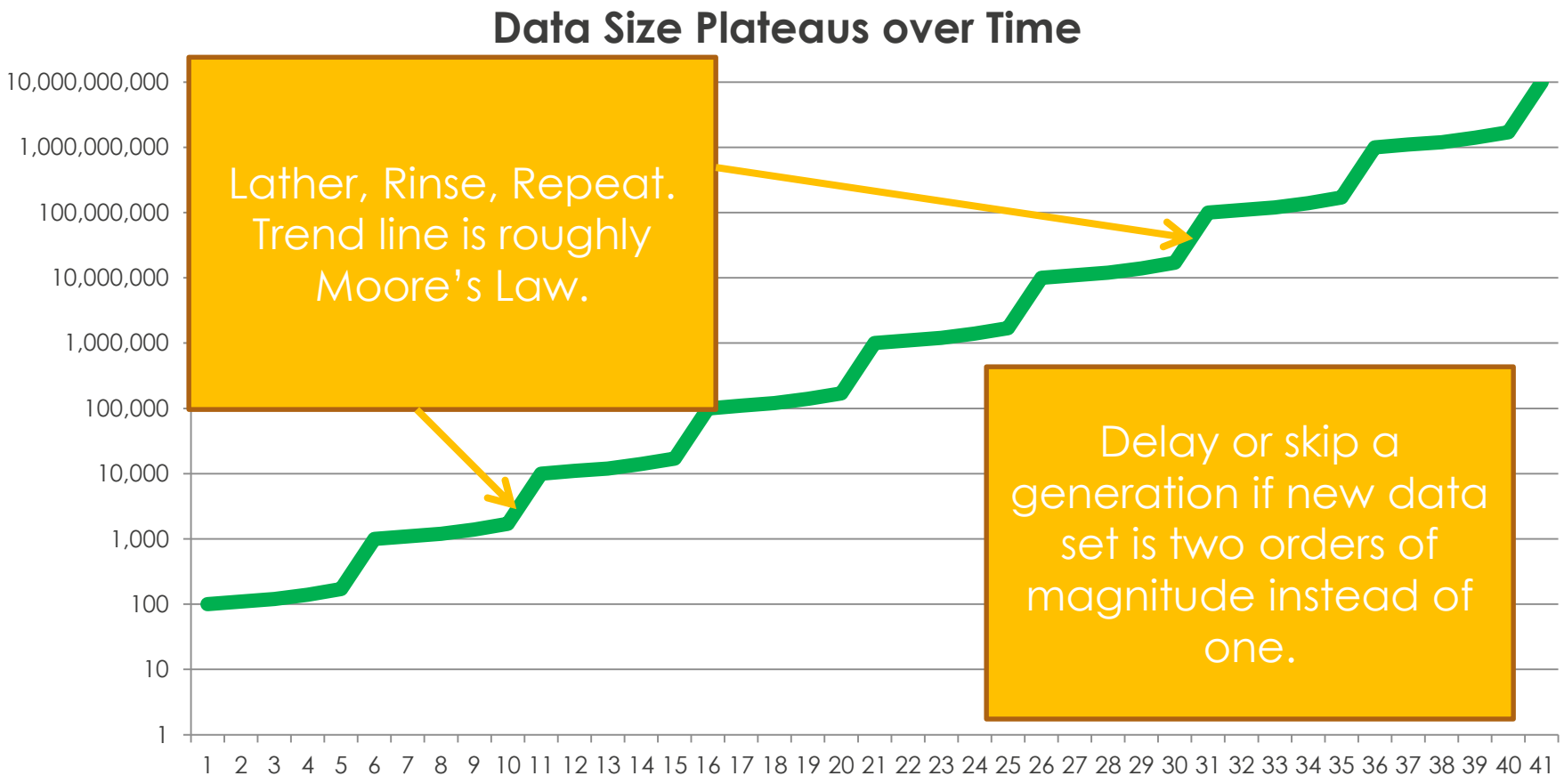
Third Nature

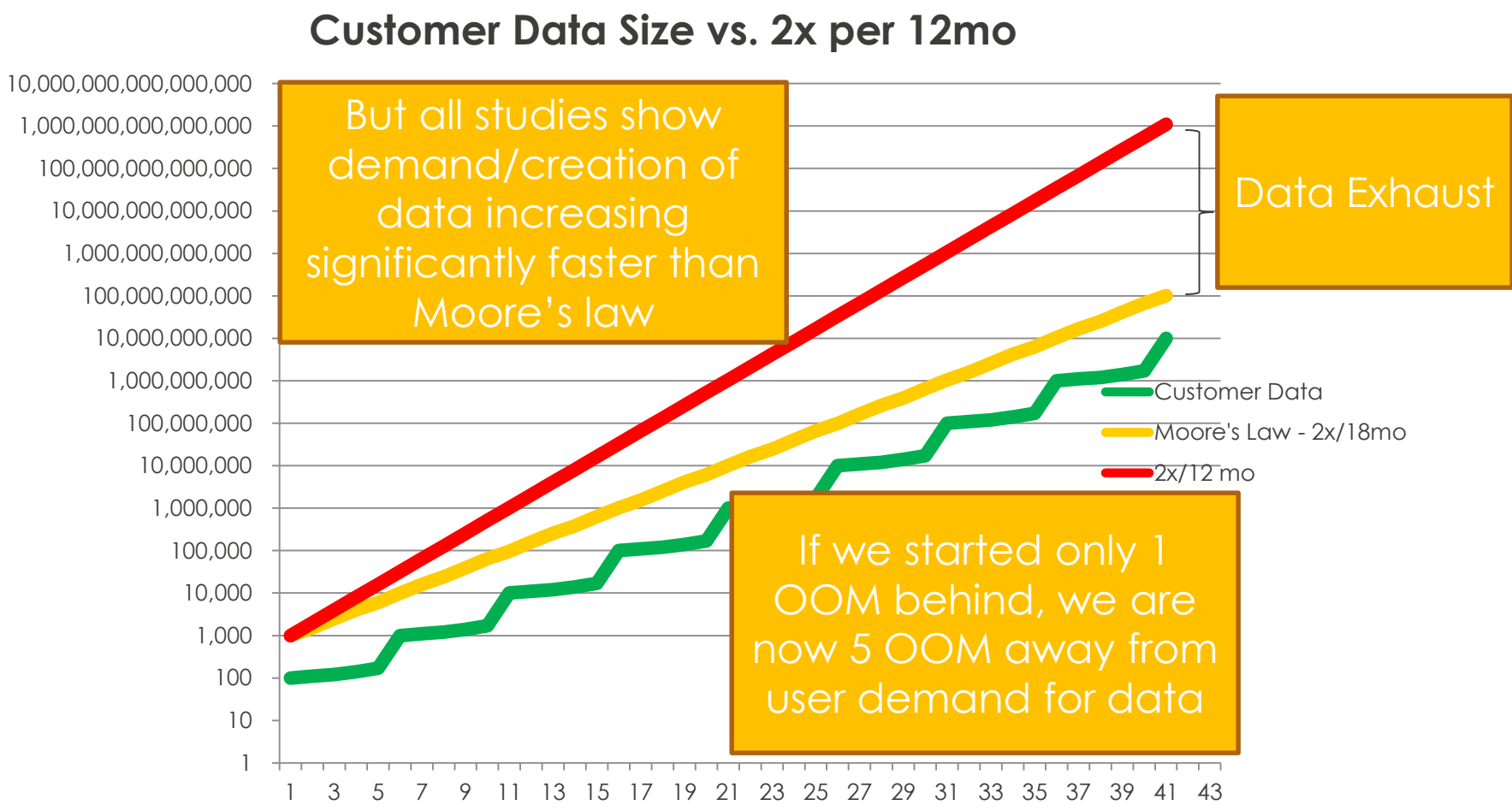

Customer Data Plateaus
10,000
1,000
100
1
2
3
4
5
6
7
8
9
10
Add users, accumulate history, add attributes, add dimensions
Entirely new data set enabled by new technology at new price point – value has to exceed cost (ROI). Earliest adopters have vision ahead of ROI



User adoption of new data sets starts over. Very small number of experts growing to wider audience, sophisticated users moving to business analysts, then business users, B2B customers and even to consumers

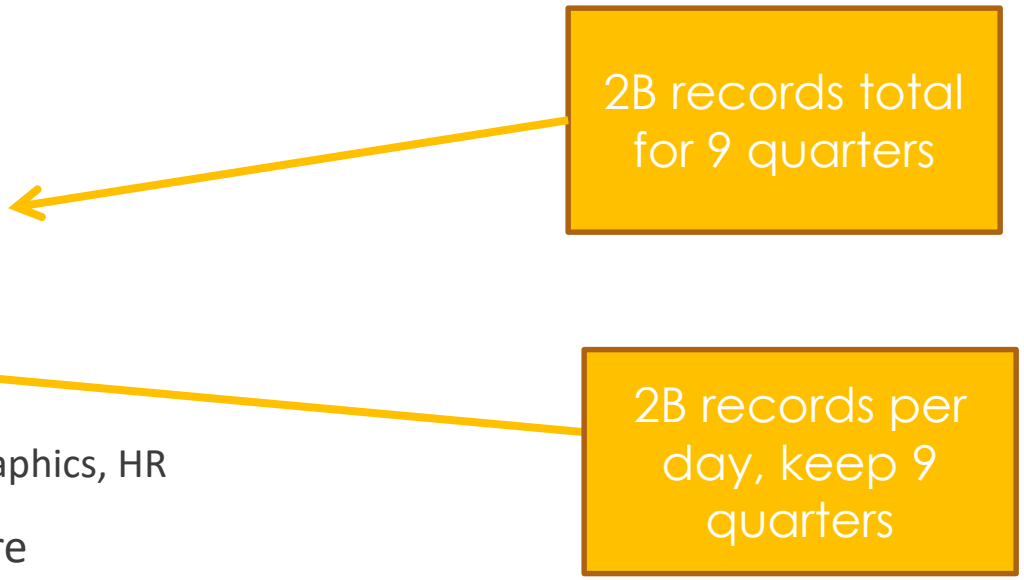
Greater value is derived when data sets are linked – see bigger picture (eg who buys what, when). Comes after initial extraction of easy value from standalone data


Data Size Plateaus over Time
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
100
10
1
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
Lather, Rinse, Repeat. Trend line is roughly Moore's Law.
Delay or skip a generation if new data set is two orders of magnitude instead of one.

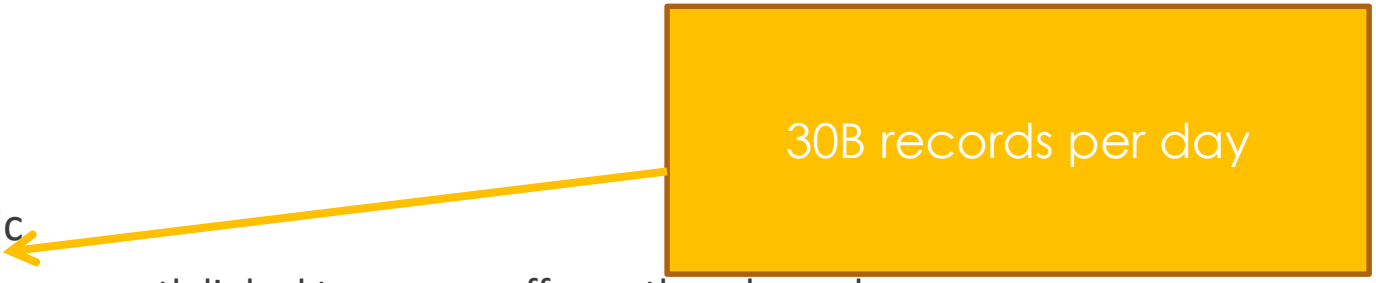

Customer Data Size vs. 2x per 12mo
10,000,000,000,000,000
1,000,000,000,000,000
100,000,000,000,000
10,000,000,000,000
1,000,000,000,000
100,000,000,000
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
100
10
1
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43
But all studies show demand/creation of data increasing significantly faster than Moore's law
Data Exhaust
Customer Data
Moore's Law - 2x/18mo
2x/12 mo
If we started only 1 OOM behind, we are now 5 OOM away from user demand for data

Retail Plateaus

- <Store, Item, Week>
- <Store, Item, Day>
  - Simple aggregations
- Market Basket
  - Affinity
  - Link to person, demographics, HR
- Inventory by SKU by store
  - Temporal, time series, forecasting
  - Link to product, marketing, market basket

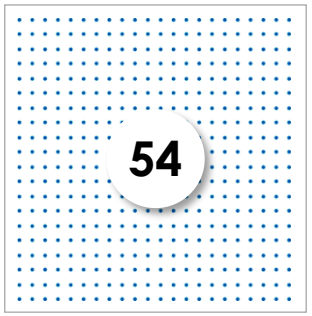


Retail Plateaus



- Web Logs and traffic
  - Behavioral patterns – eg path linked to person, offers, other channels
  - Operations of the web site
- Supply chain sensors – sampled at major event
  - Activity Based Costing
  - Link to customer, product, HR, planning
- Social Media
  - Text analysis, Filtering, languages
  - Link to customer, sales, other channel interactions
- Supply chain sensors – sampled at minutes or seconds
  - Telematics
  - Real time, Event detection, trending, static and dynamic rules
  - Link to HR, thresholds, forecasts, routing, planning

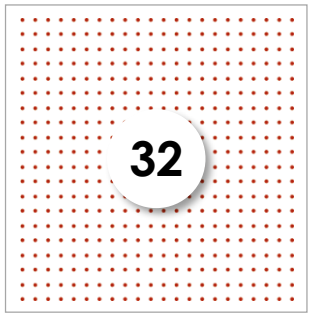
How many batteries are in inventory by plant?



OPERATIONS

Inventory
Returns
Manufacturing
Supply Chain

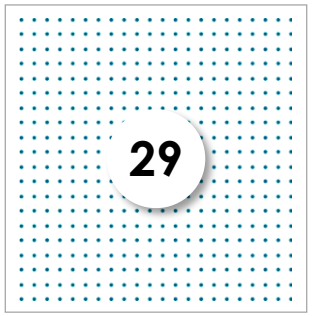
What is the trend of warranty costs?



FINANCE

Revenue
Expenses
Customers

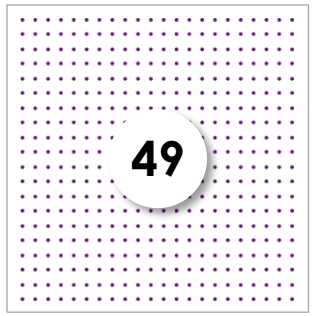
How many people made a warranty claim last week?



CUSTOMER CARE

Customer
Products
Orders
Case History

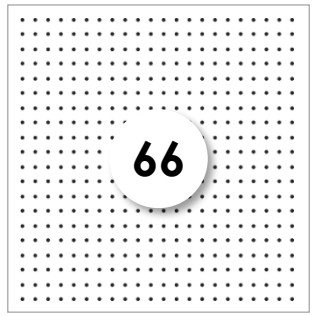
How many sales have been made quarter to date?



SALES

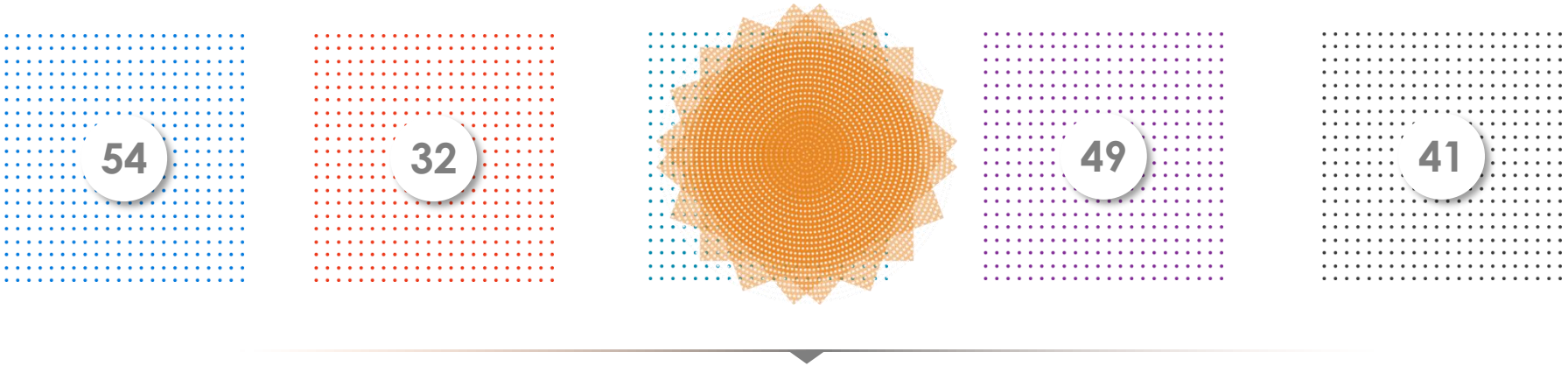
Orders
Customers
Products

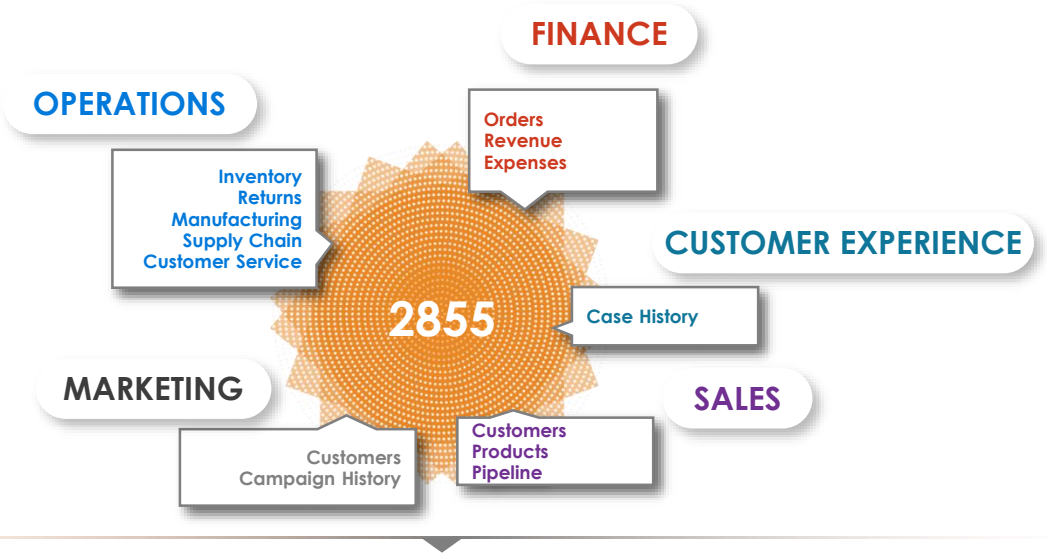
Which customers should get a communication on extended warranties?



MARKETING

Customers
Orders
Campaign History


54
32
49
41



Given the rise in **warranty costs**, isolate the problem to be a **plant**, then to a **battery lot**. Communicate with **affected customers**, who have not made a warranty claim on batteries, through **Marketing** and **Customer Service** channels to recall cars with affected batteries.

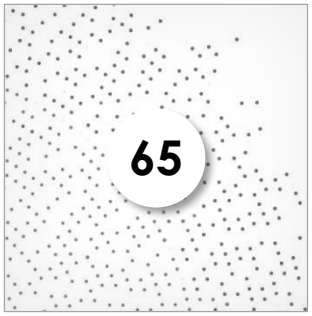
Manufacturing: Data Overlap Analysis

New Business Improvement Opportunities through Data Leverage

Then ↓ / If →	Sales Force Profitability Analysis	Transportation Planning	Production Planning	Vendor Managed Inventory	Global Pricing Rationalization	Fulfillment (Perfect Order)	Manufacturing Quality Optimization	Preventative Maintenance Analysis	Warranty Claims Analysis	Quality Life Cycle Improvement
Sales Force Profitability Analysis	**100%**	**80%**	66%	24%	41%	66%	0%	24%	0%	24%
Transportation Planning	11%	**100%**	**87%**	28%	64%	56%	22%	34%	13%	45%
Production Planning	6%	57%	**100%**	19%	**83%**	35%	17%	28%	9%	40%
Vendor Managed Inventory	12%	**100%**	**100%**	**100%**	**78%**	**100%**	61%	**100%**	39%	**100%**
Global Pricing Rationalization	4%	50%	**100%**	17%	**100%**	27%	15%	29%	11%	41%
Fulfillment (Perfect Order)	16%	**94%**	**90%**	48%	58%	**100%**	35%	53%	19%	56%
Manufacturing Quality Optimization	0%	**76%**	**88%**	60%	66%	**71%**	**100%**	**79%**	43%	**73%**
Preventative Maintenance Analysis	7%	**75%**	**93%**	61%	**80%**	68%	50%	**100%**	27%	**82%**
Warranty Claims Analysis	0%	**94%**	**100%**	**83%**	**100%**	**83%**	**94%**	**93%**	**100%**	**98%**
Quality Life Cycle Improvement	4%	57%	**75%**	35%	64%	41%	26%	47%	16%	**100%**

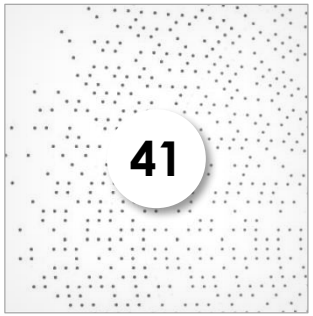

BIAS

How many visitors did we have to our hybrid cars microsite yesterday?



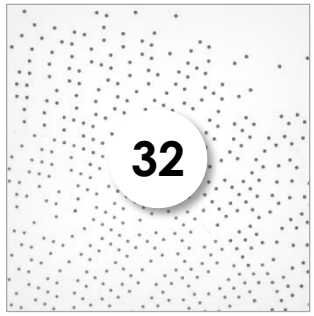
CLICKSTREAM

What are the temperature readings for batteries by Manufacturer?



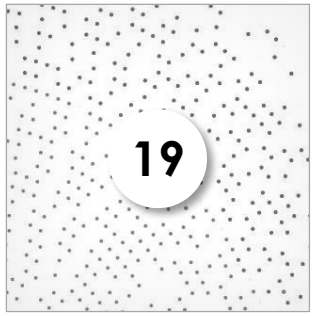
PRODUCT SENSOR

What is the sentiment towards line of hybrid vehicles?



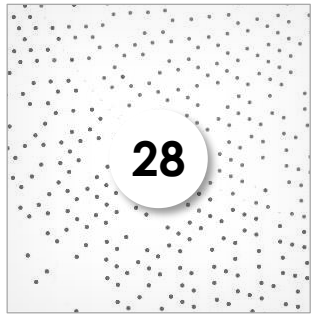
SOCIAL MEDIA

Which customers likely expressed anger with customer care?



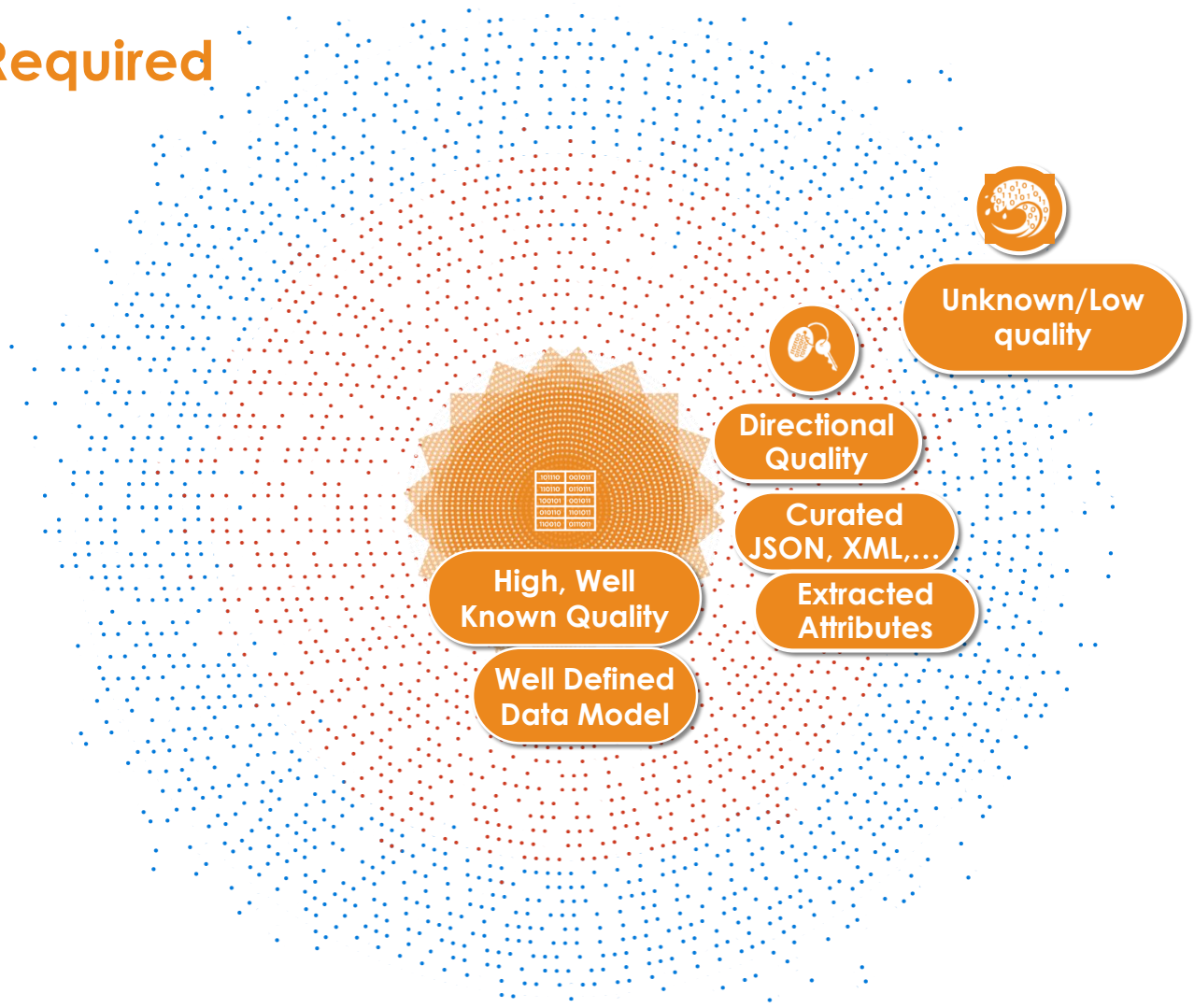
CUSTOMER CARE AUDIO RECORDINGS

Which ad creative generated the most clicks?

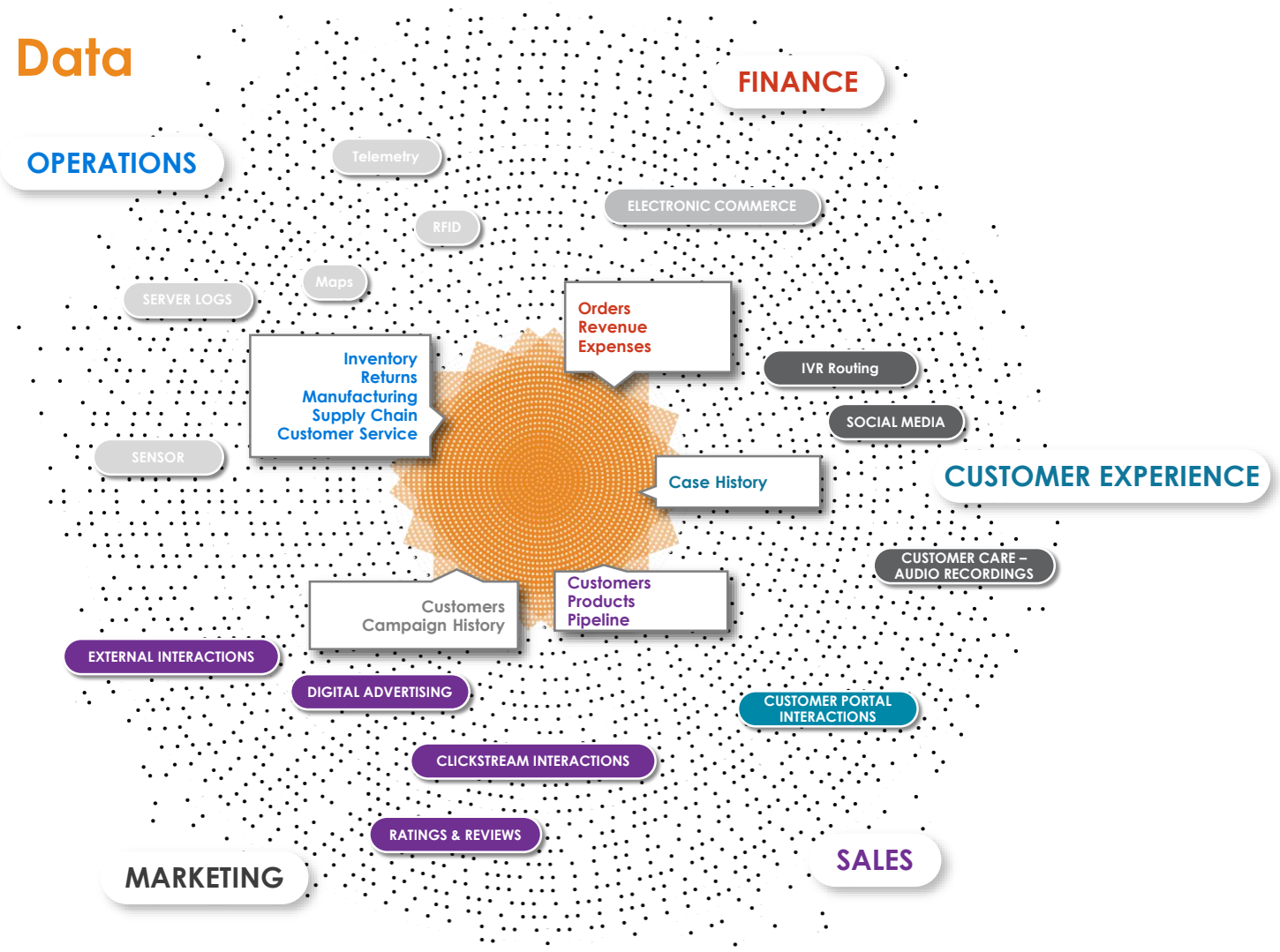


DIGITAL ADVERTISING

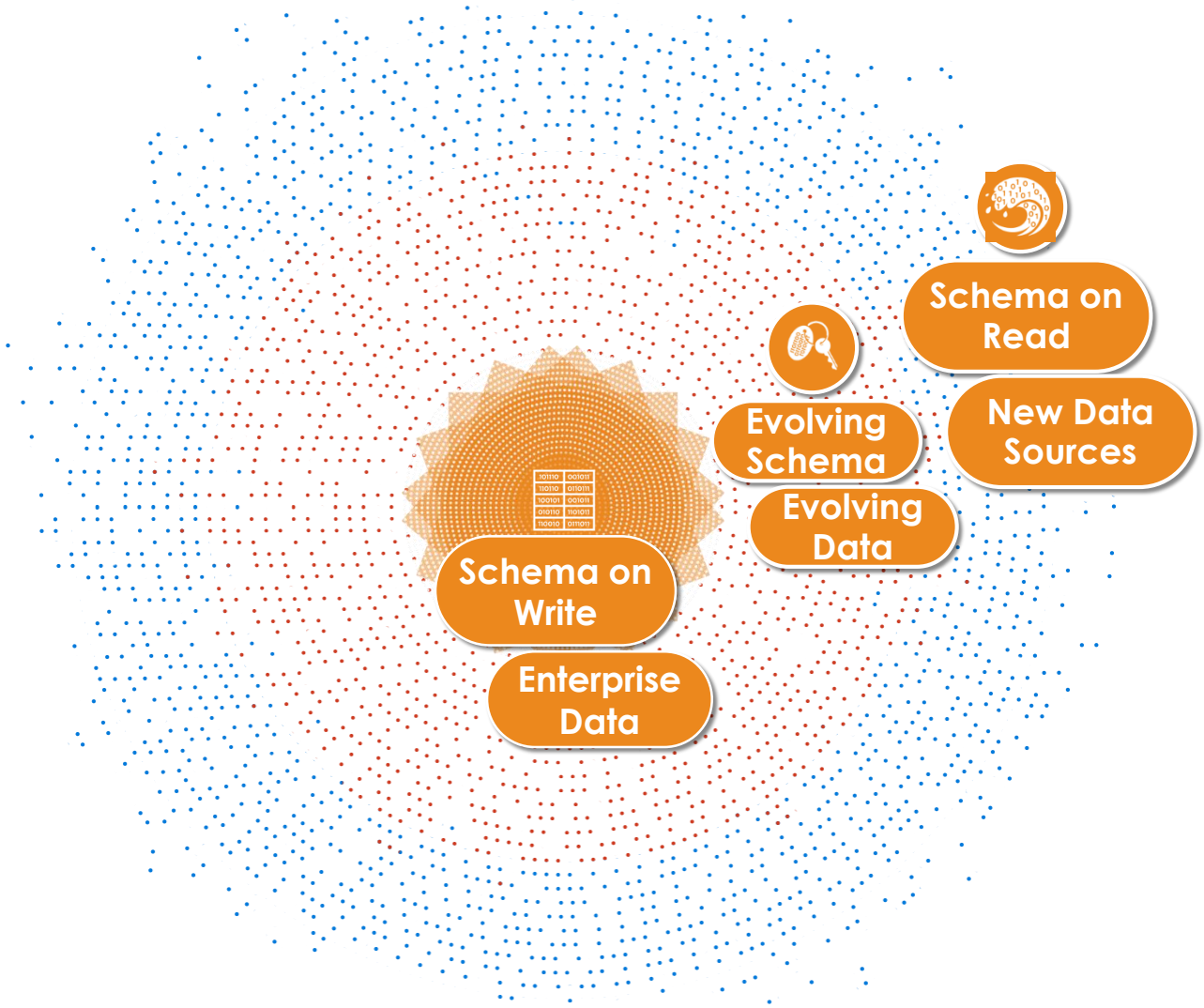
Curation Required



Enterprise Data

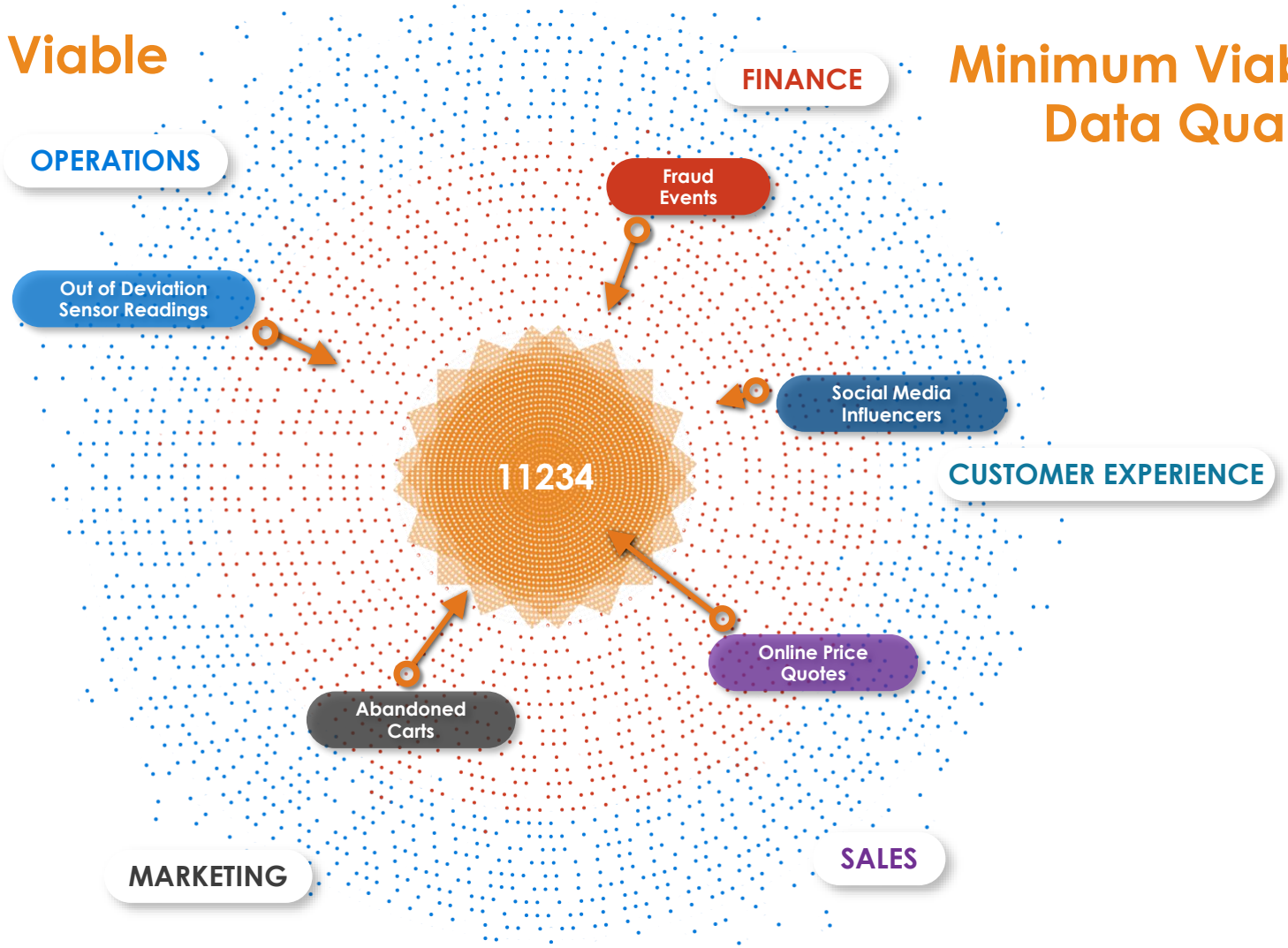


Schema?

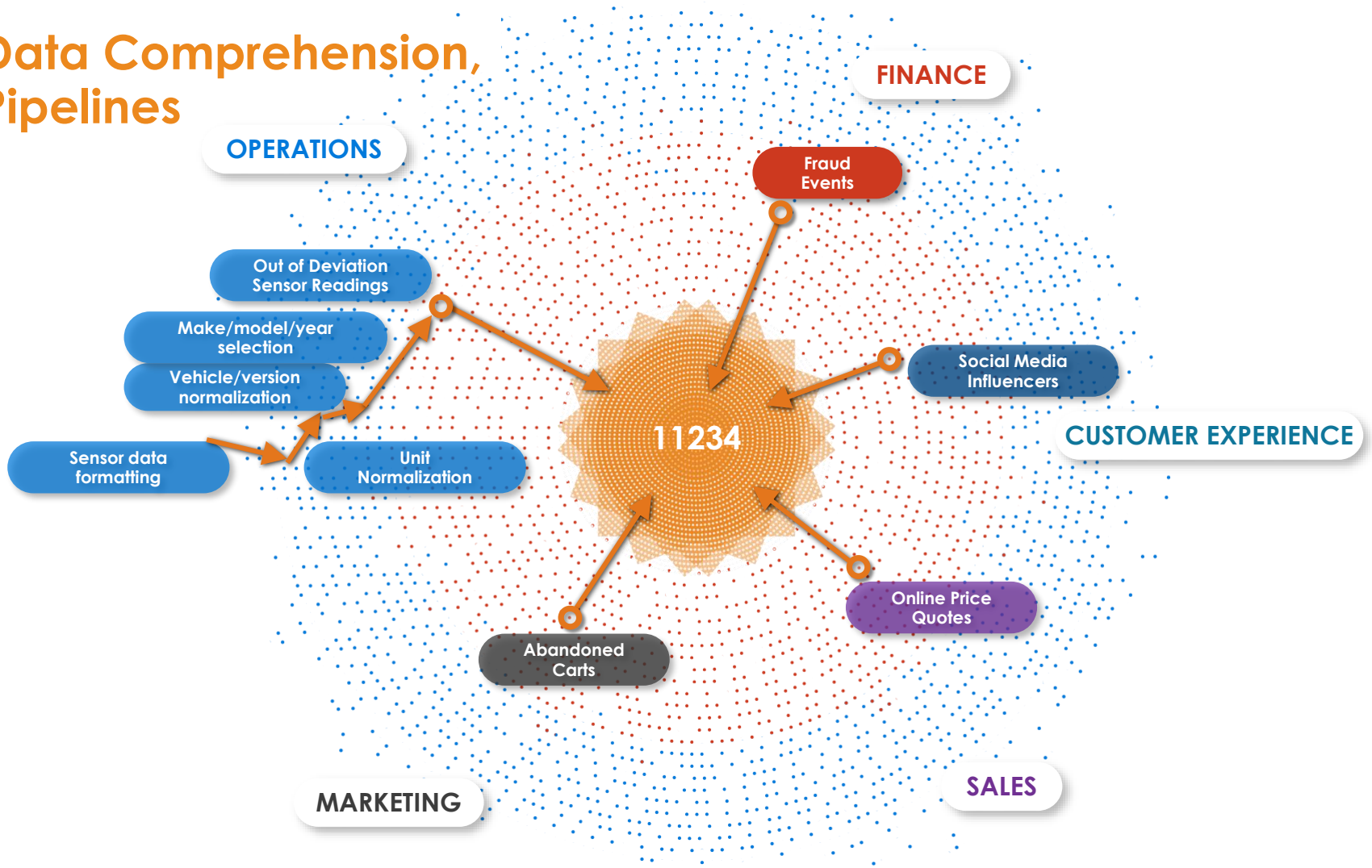

Schema on Read
New Data Sources
Evolving Schema
Evolving Data
Schema on Write
Enterprise Data

Minimum Viable Curation

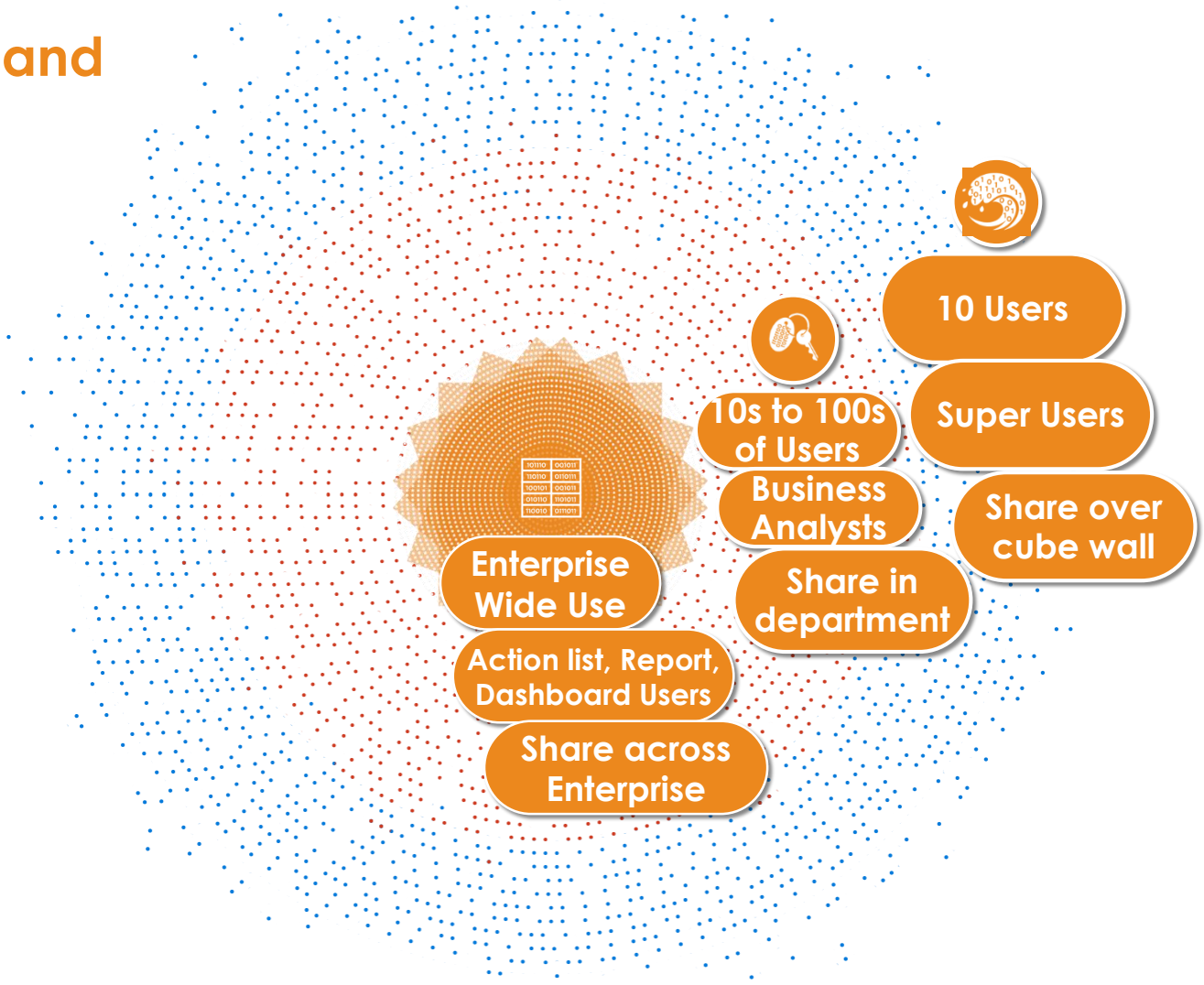
Minimum Viable Data Quality



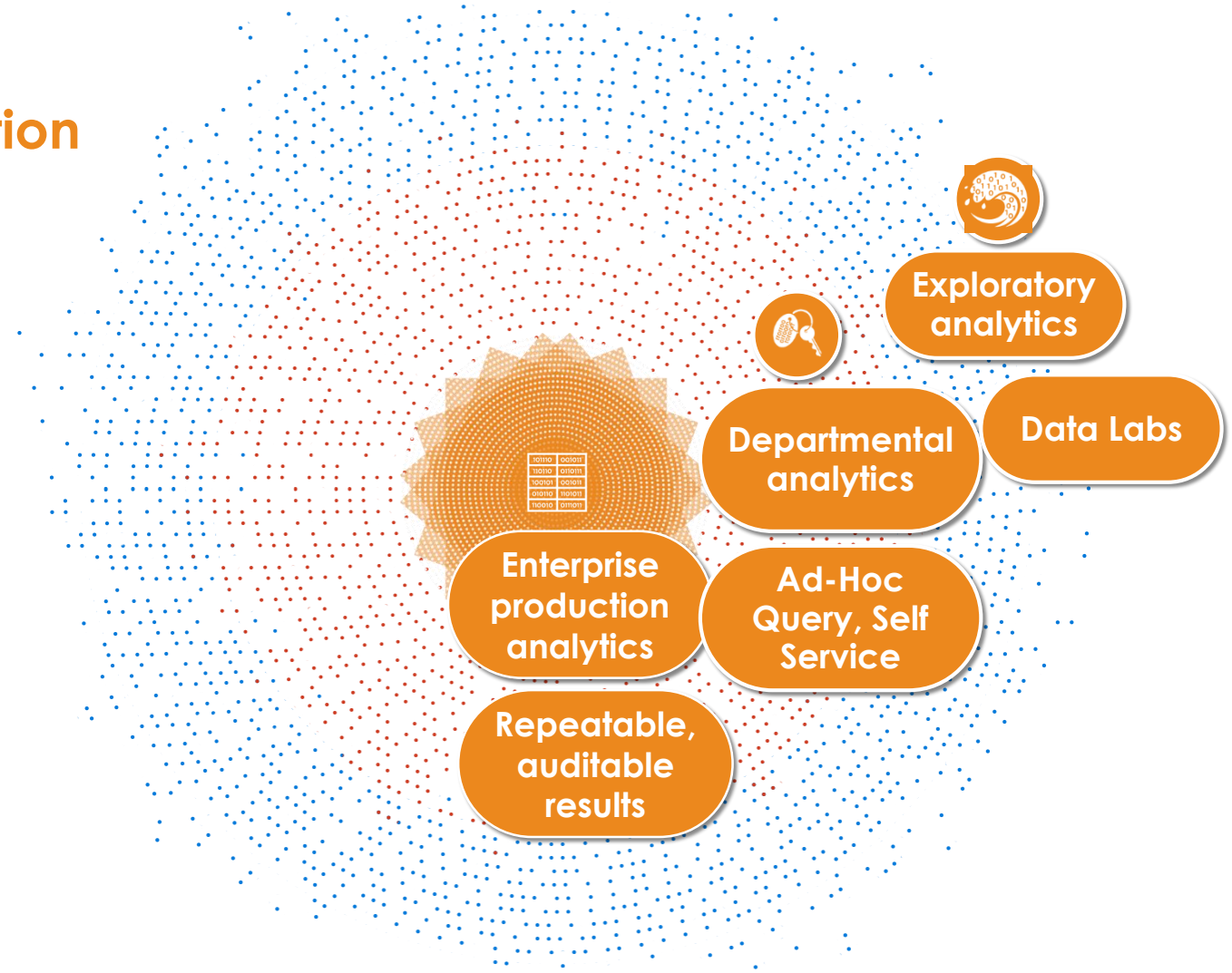
Data Comprehension, Pipelines



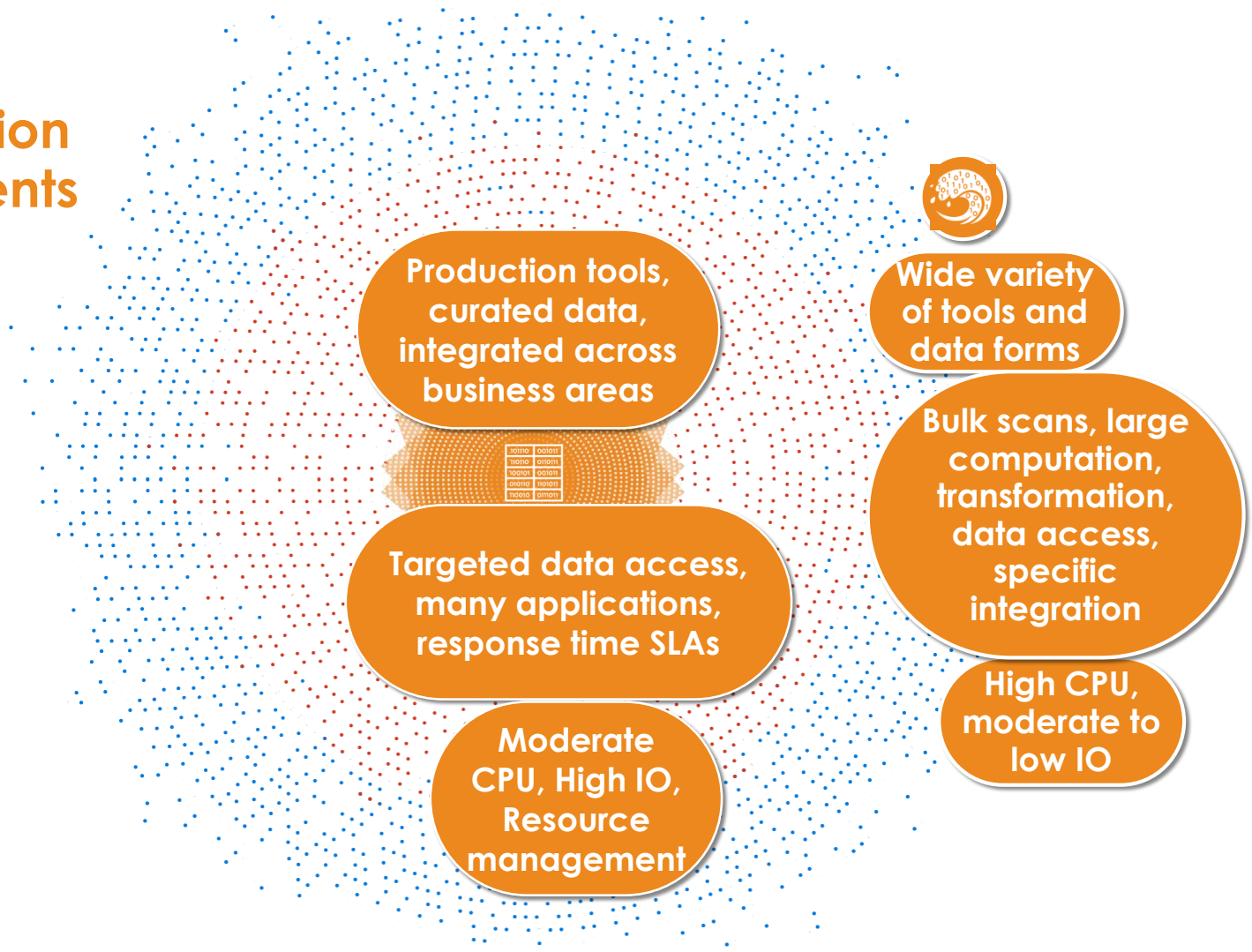
User Base and Sharing



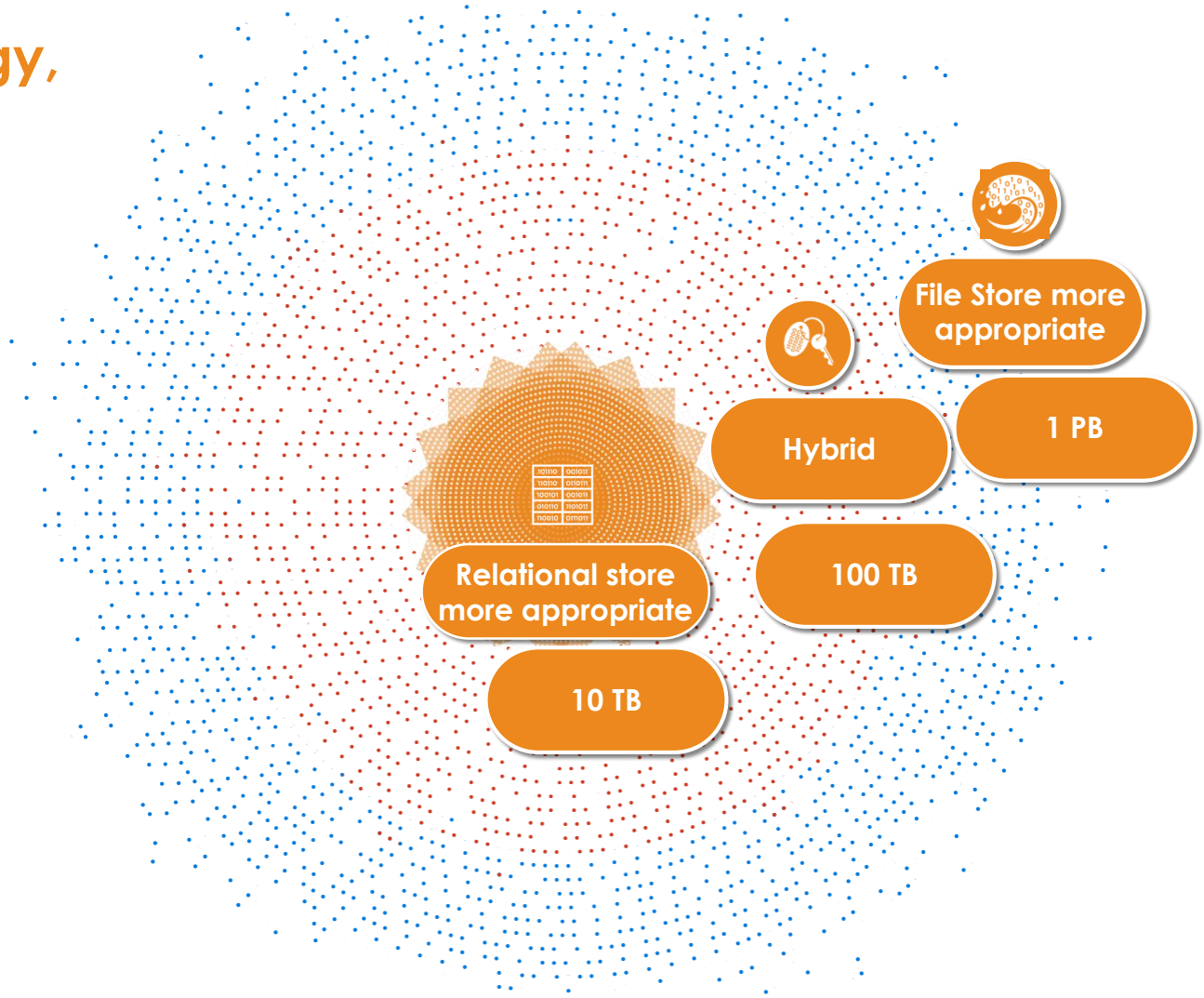
Evolving Consumption



Evolving Consumption Requirements



Technology, Capacity

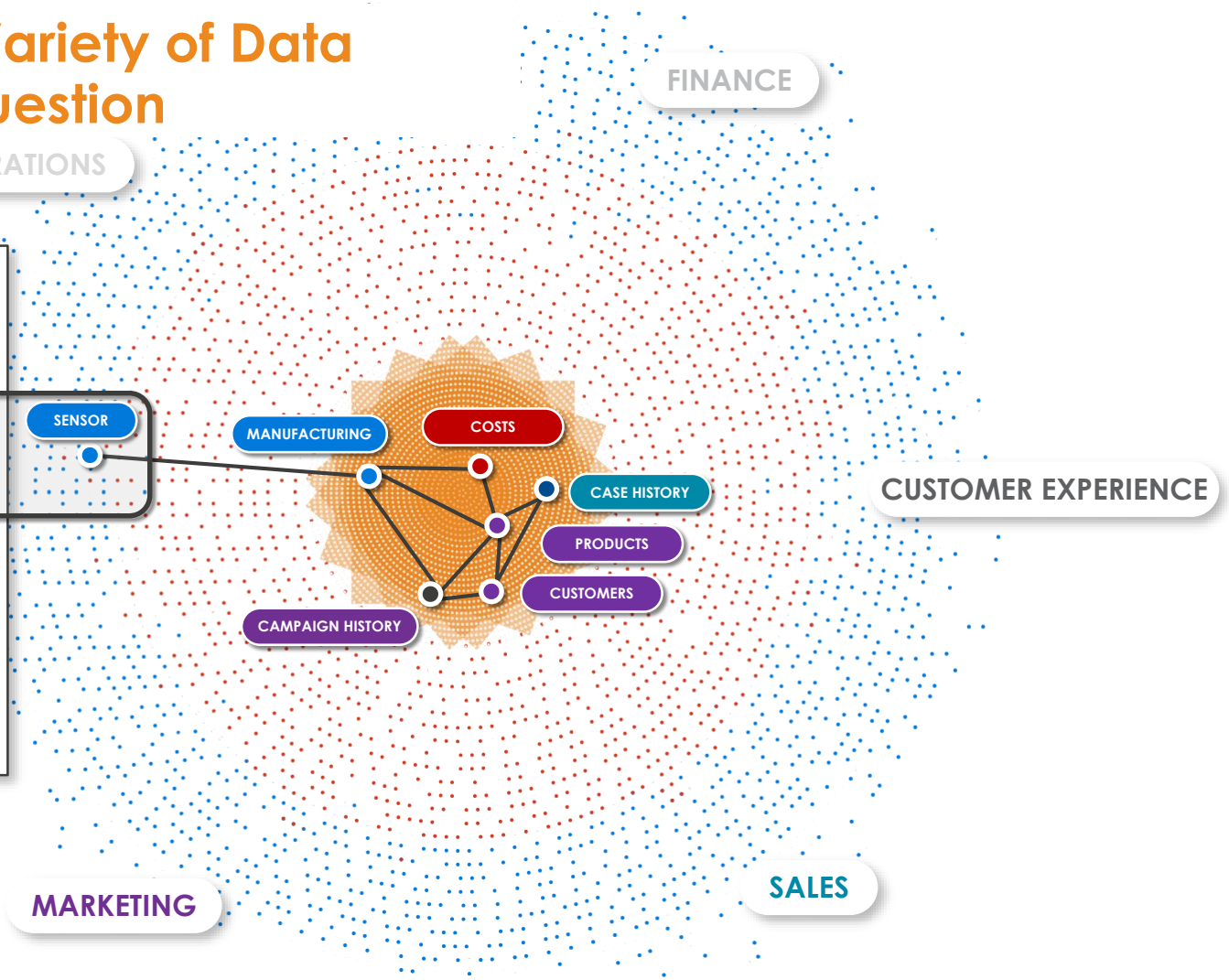


Access Wide Variety of Data to Answer a Question

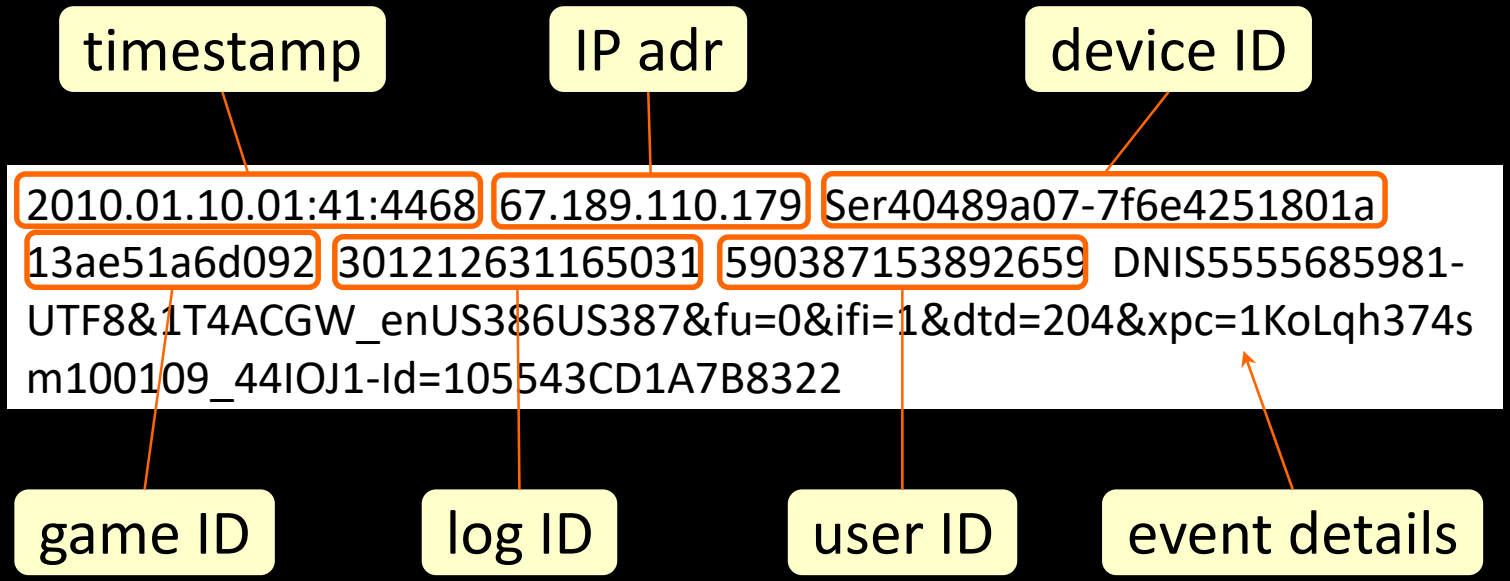
Given the rise in warranty costs, isolate the problem to be a plant and the specific lot.

Exclude 2/3rd of the batteries from the lot that are fine.

Communicate with affected customers, who have not made a warranty claim, through Marketing and Customer Service channels to recall cars with affected batteries.



An event contains mainly IDs…that reference other data



Log de-referencing and enrichment is difficult since you can't enforce integrity like you can in a DB.

What's the glue that holds it together?

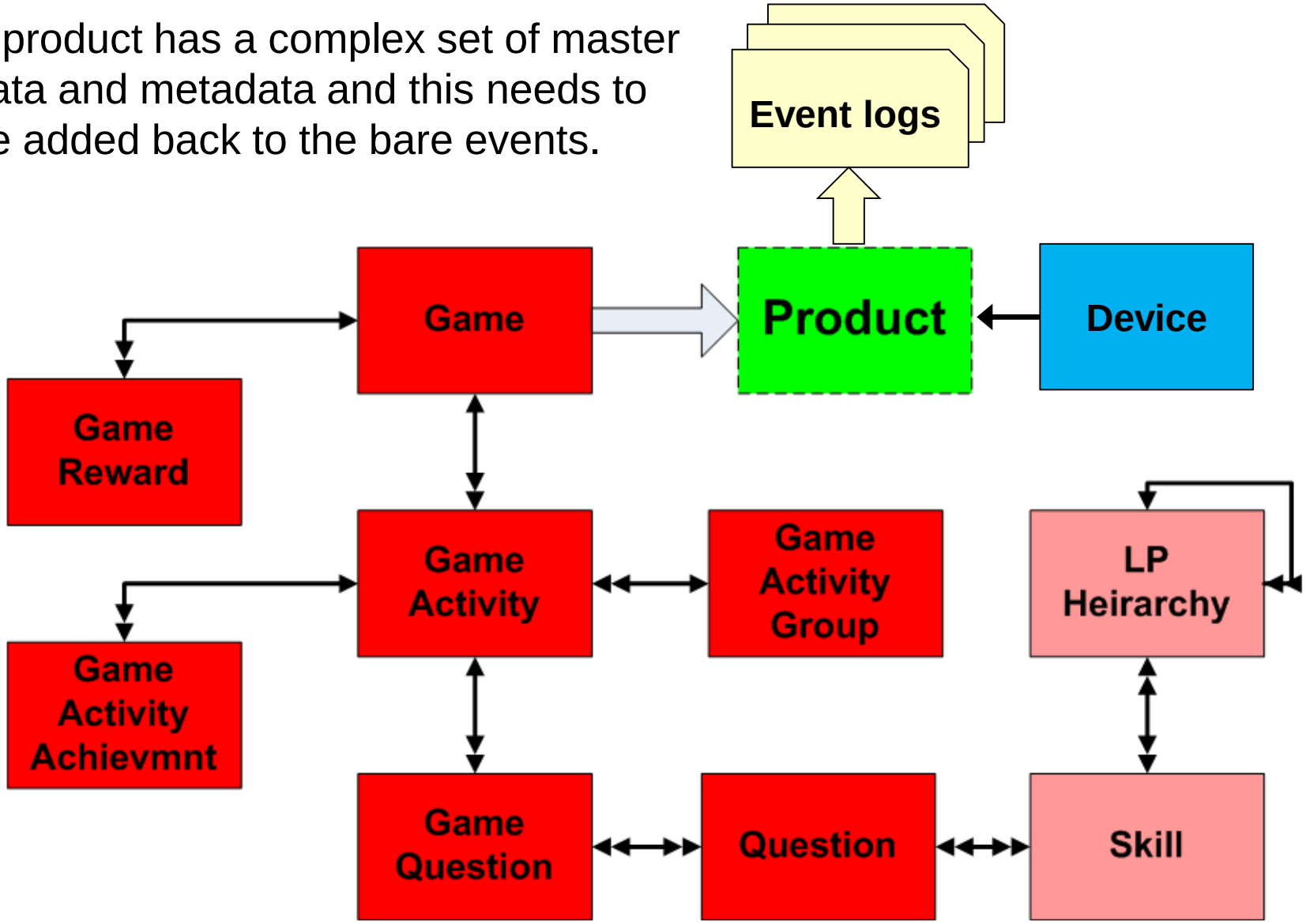
It's just keys to other data.

e.g. remember that device ID 0 problem?

Third Nature

It’s not just the log data, it’s big data + small data

A product has a complex set of master data and metadata and this needs to be added back to the bare events.



Third Nature

Where does the reference data come from?

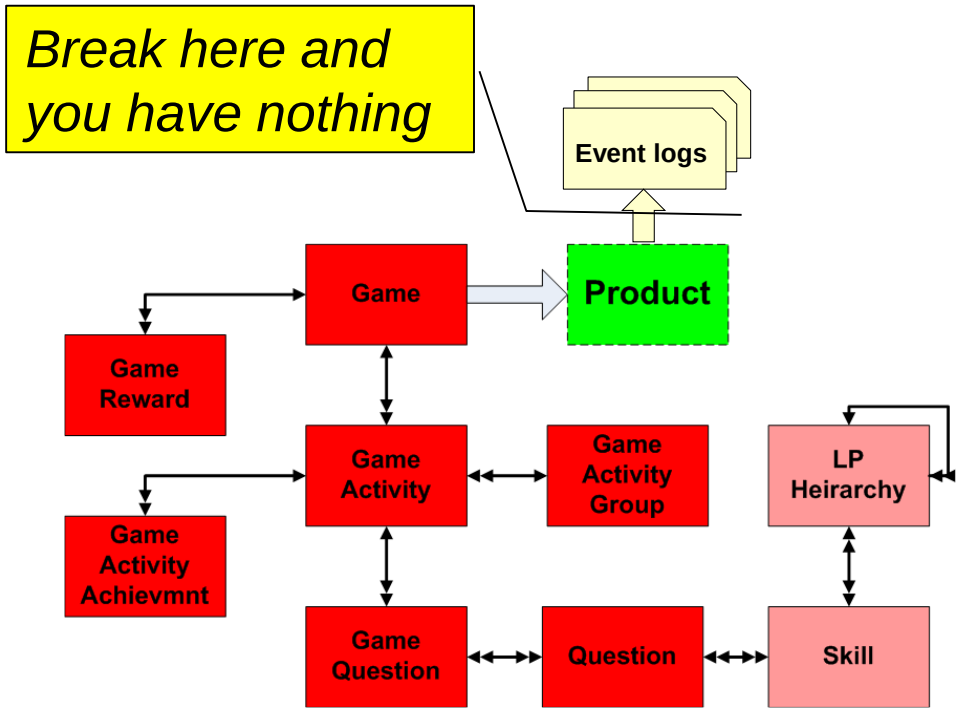
The keys come from somewhere. You don't just make up system-wide unique identifiers in your code.

The lack of local lookup data at event generation leads to development practices that lead to inconsistencies.

Problem we had: product identifiers that didn't match any known products

Had to fix by analyzing each log of bad-ID events.

Because developers used config files they copied from the PMS and put in the code

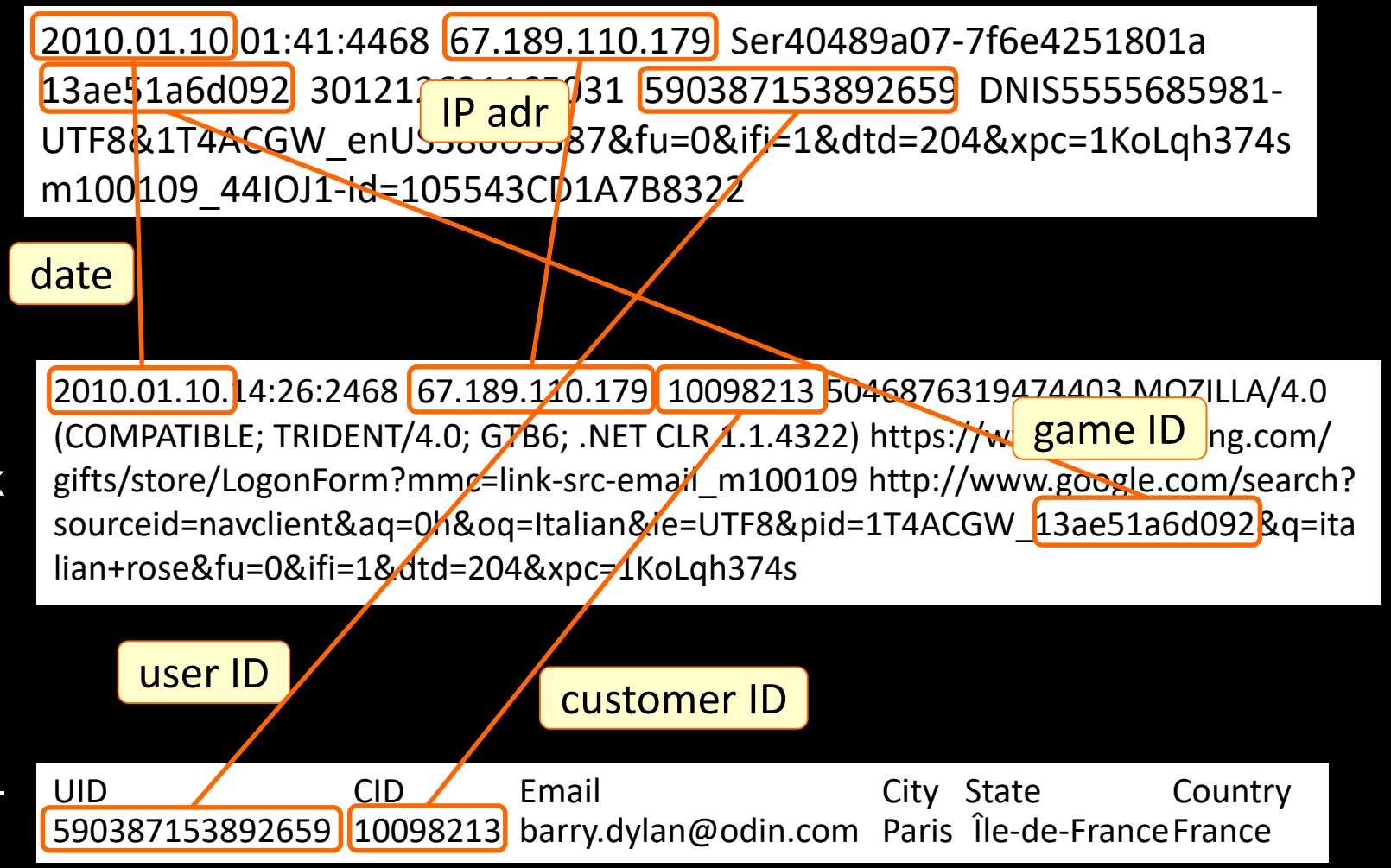


Third Nature

MDM again

If you want to link datasets then you must manage the keys

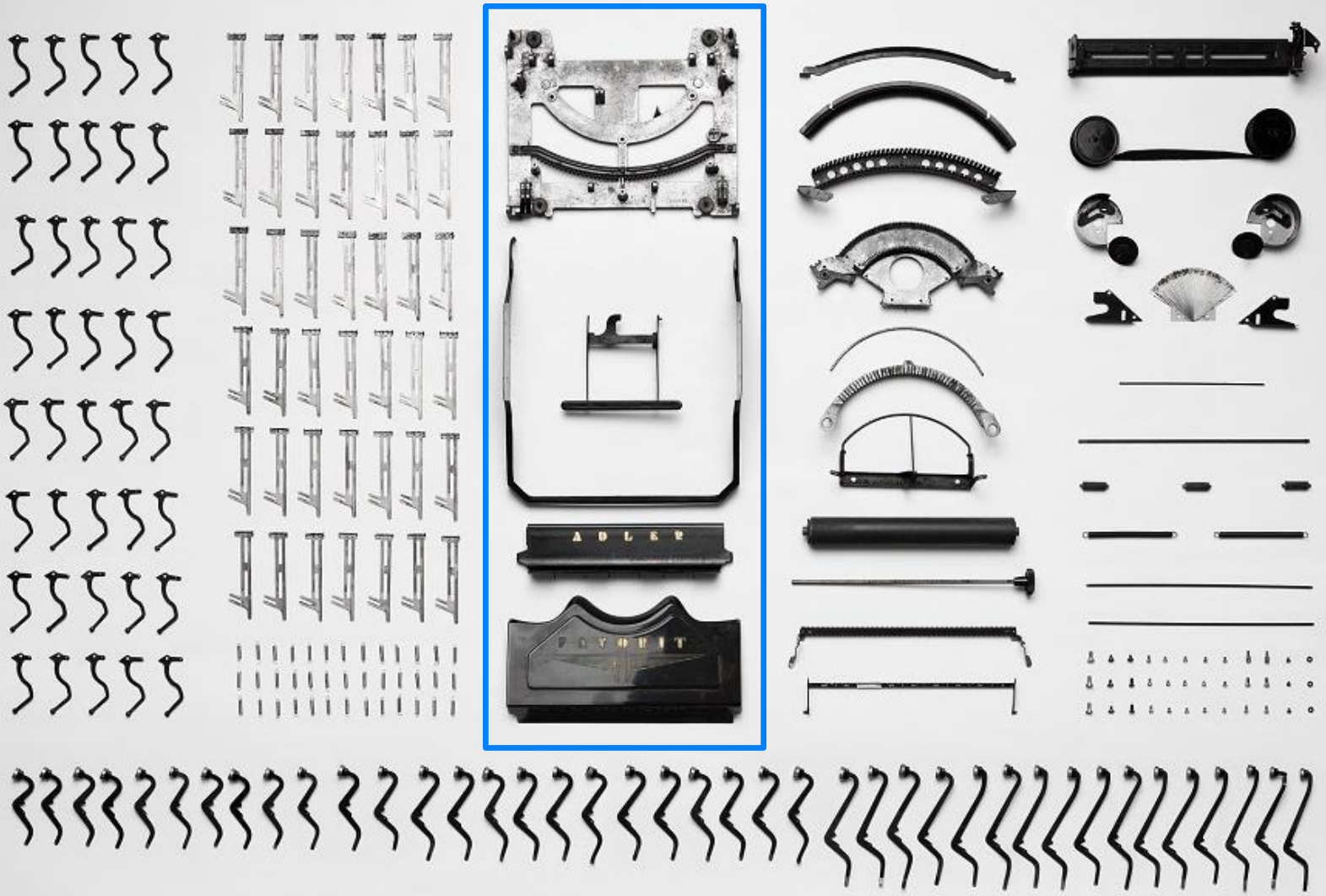
You need canonical forms for common data (in code too)



Third Nature

What you think you have

What you really have



This is called “self-service.” Analysts will love it…

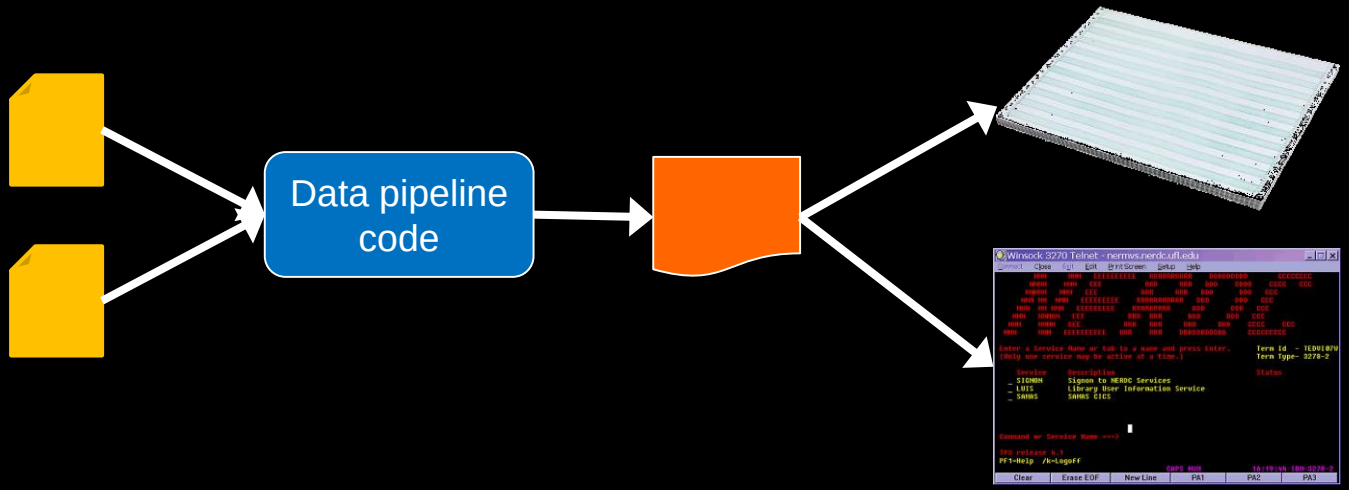
Third Nature

History: This is how BI was done through the 80s

First there were files and reporting programs.

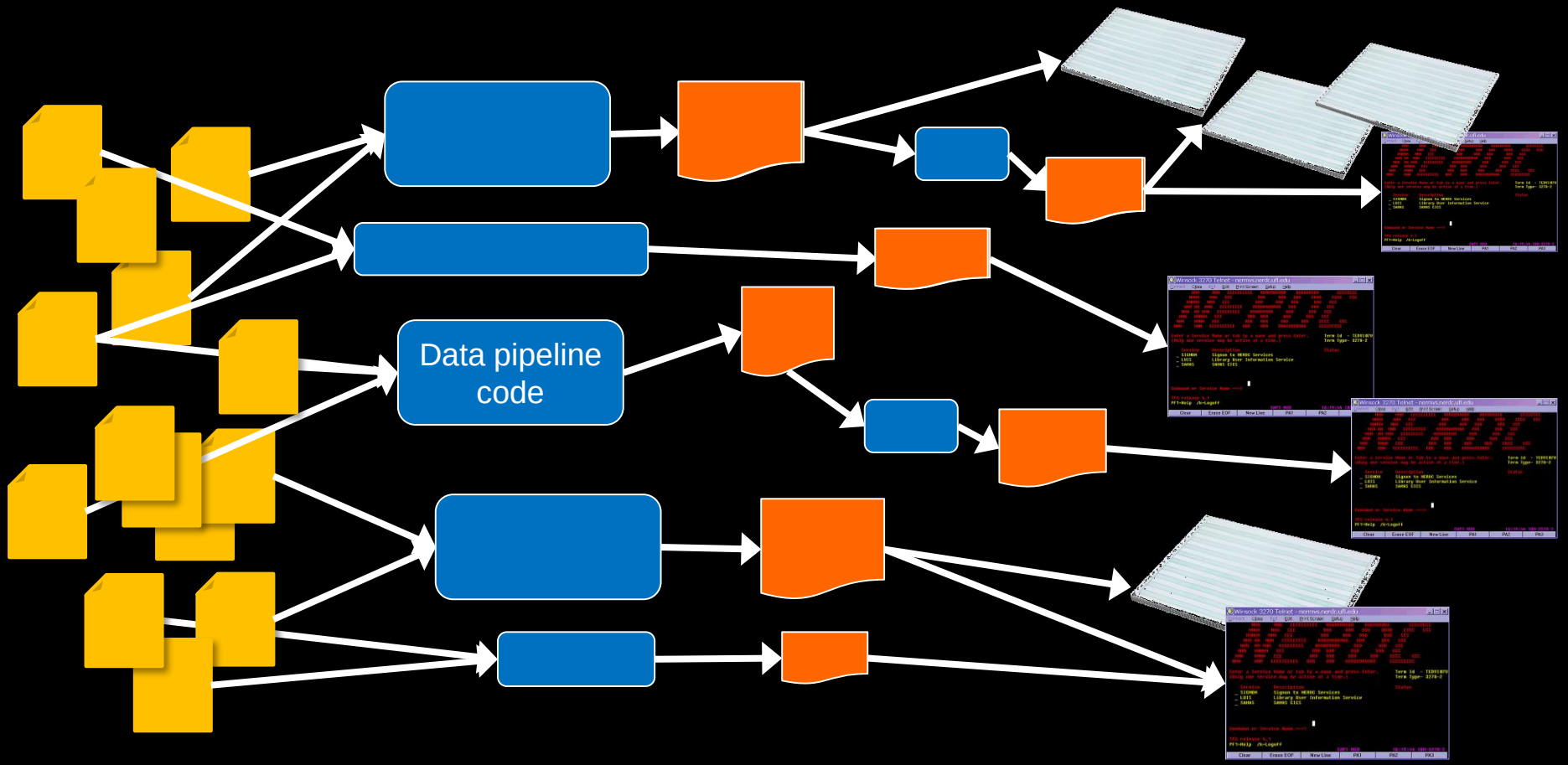
Application files feed through a data processing pipeline to generate an output file. The file is used by a report formatter for print/screen. Files are largely single-purpose use.

Every report is a program written by a developer.



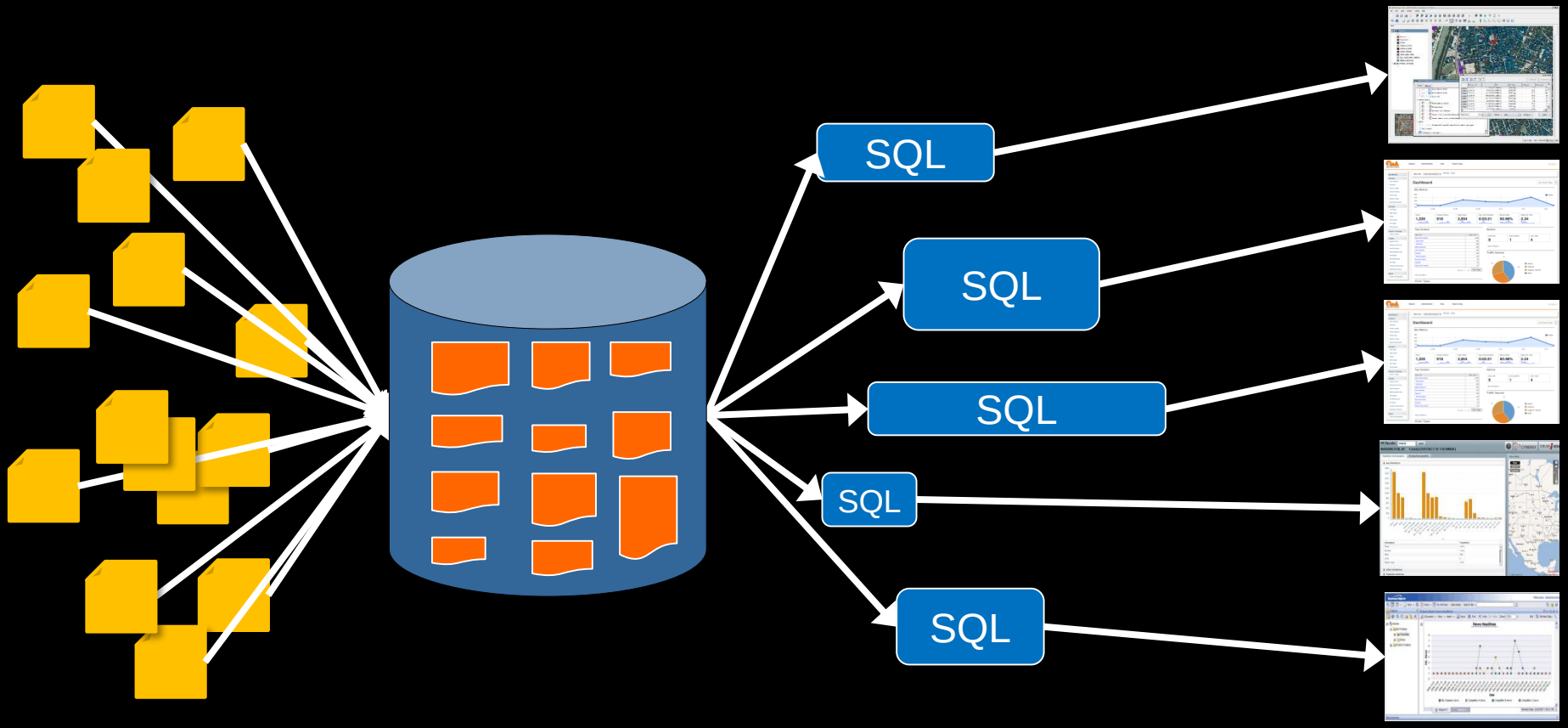
History: This is how BI ended the 80s

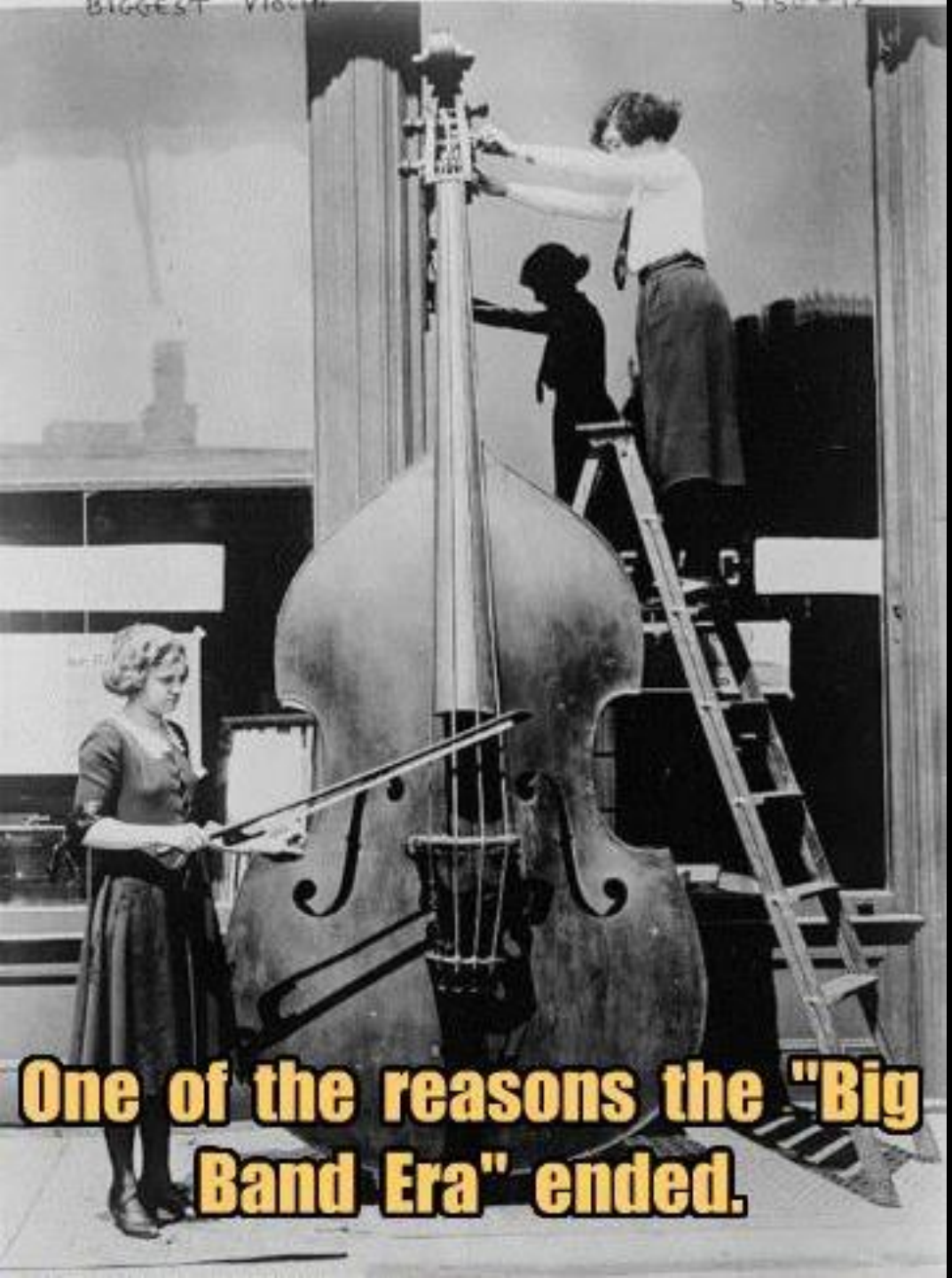
The inevitable situation was...



History: This is how we started the 90s

Collect data in a database. Queries replaced a *LOT* of application code because much was just joins. We learned about “dead code”





Pragmatism and Data

Lessons learned during the ad-hoc SQL era of the DW market:

- When the technology is awkward for the users, the users will stop trying to use it.
- Even “simple” schemas weren’t enough for anyone other than analysts and their Brio…

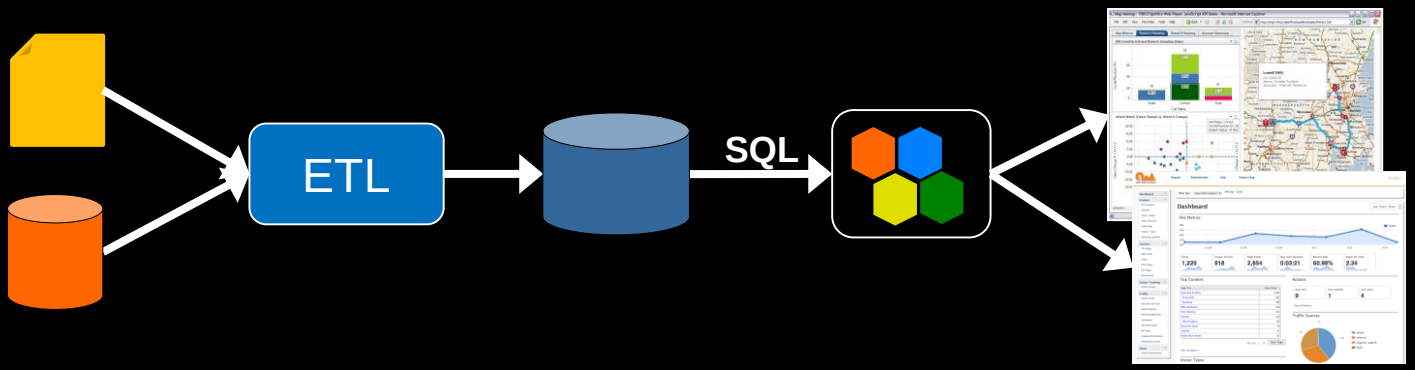
Led to the evolution of metadata-driven SQL-generating BI tools, ETL tools.

BI evolved to hiding query generation for end users

With more regular schema models, in particular dimensional models that didn't contain cyclic join paths, it was possible to automate SQL generation via semantic mapping layers.

We developed data pipeline building tools (ETL).

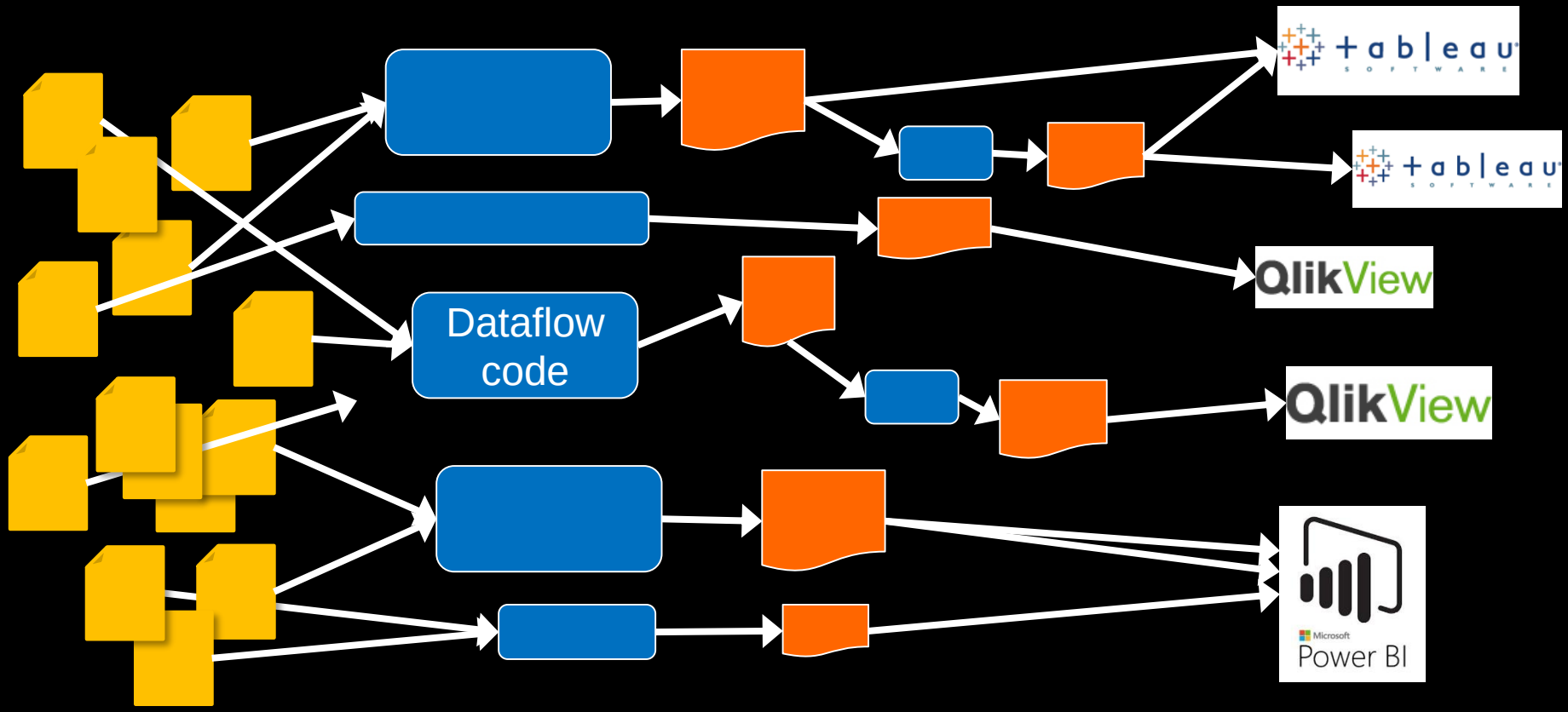
Query via business terms made BI usable by non-technical people.



Life got much easier…for a while

Today's model: Lake + self-service, looks familiar…

The Lake with data pipelines to files or Hive tables
is exactly the same pattern as the COBOL batch..



We already know that people don't scale…

We’re so focused on the light switch that we’re not talking about the light

DATA ARCHITECTURE

Decouple the Architecture

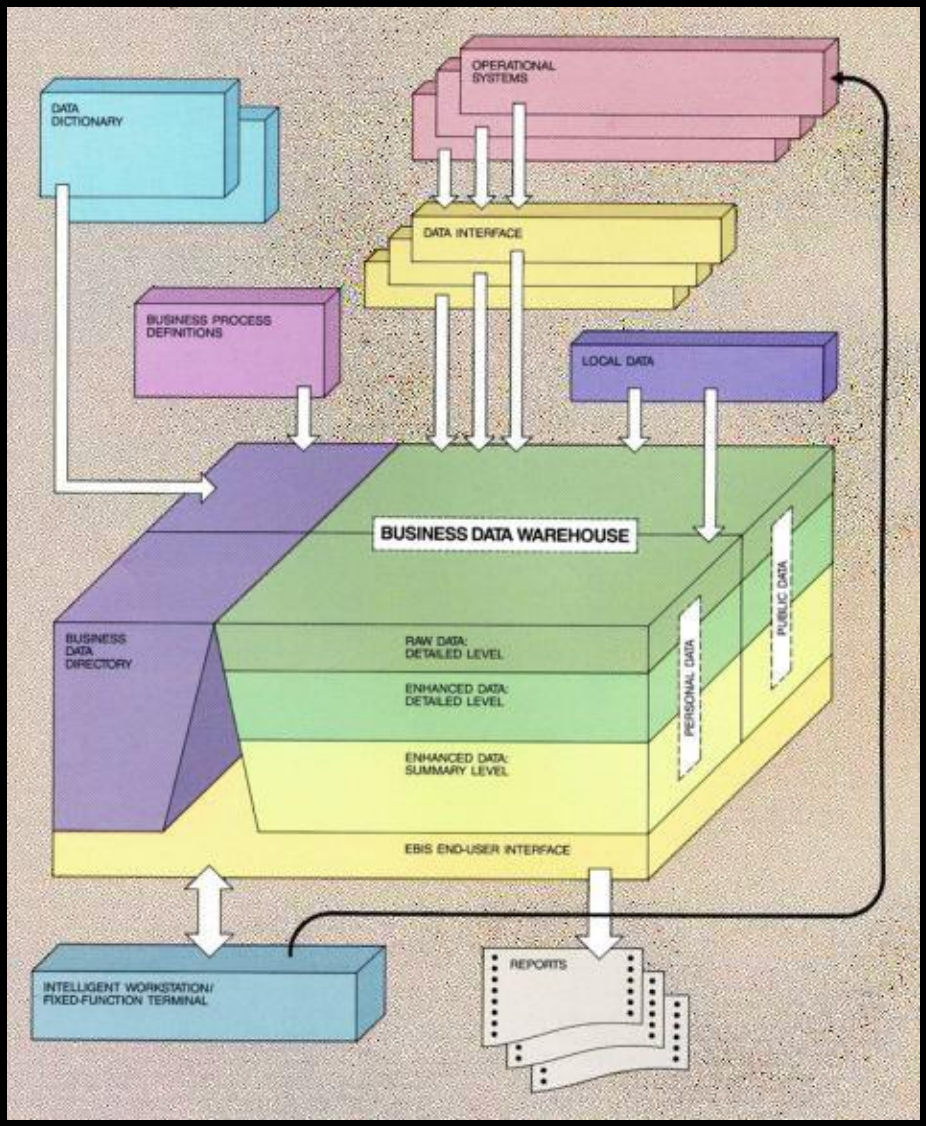
The core of the data warehouse isn't the database, it's the data architecture that the database and tools implement.

We need a data architecture that is not limiting:

- Deals with change more easily and at scale
- Does not enforce requirements and models up front
- Does not limit the format or structure of data
- Assumes the range of data latencies in and out, from streaming to one-time bulk
- Allows both reading and writing of data from outside

The architecture from 1988 we SHOULD HAVE BEEN USING

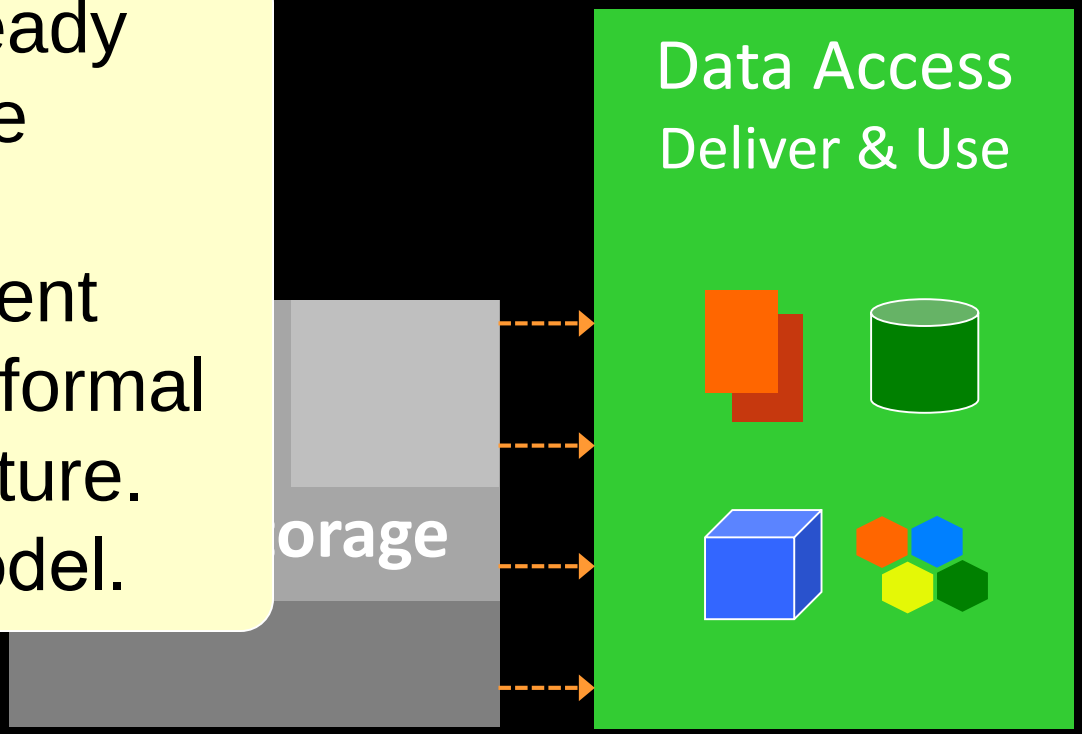
The general concept of a separate architecture for BI has been around longer, but this paper by Devlin and Murphy is the first formal data warehouse architecture and definition published.



"An architecture for a business and information system", B. A. Devlin, P. T. Murphy, IBM Systems Journal, Vol.27, No. 1, (1988)

The goal is to decouple: solve the application and infrastructure problems separately

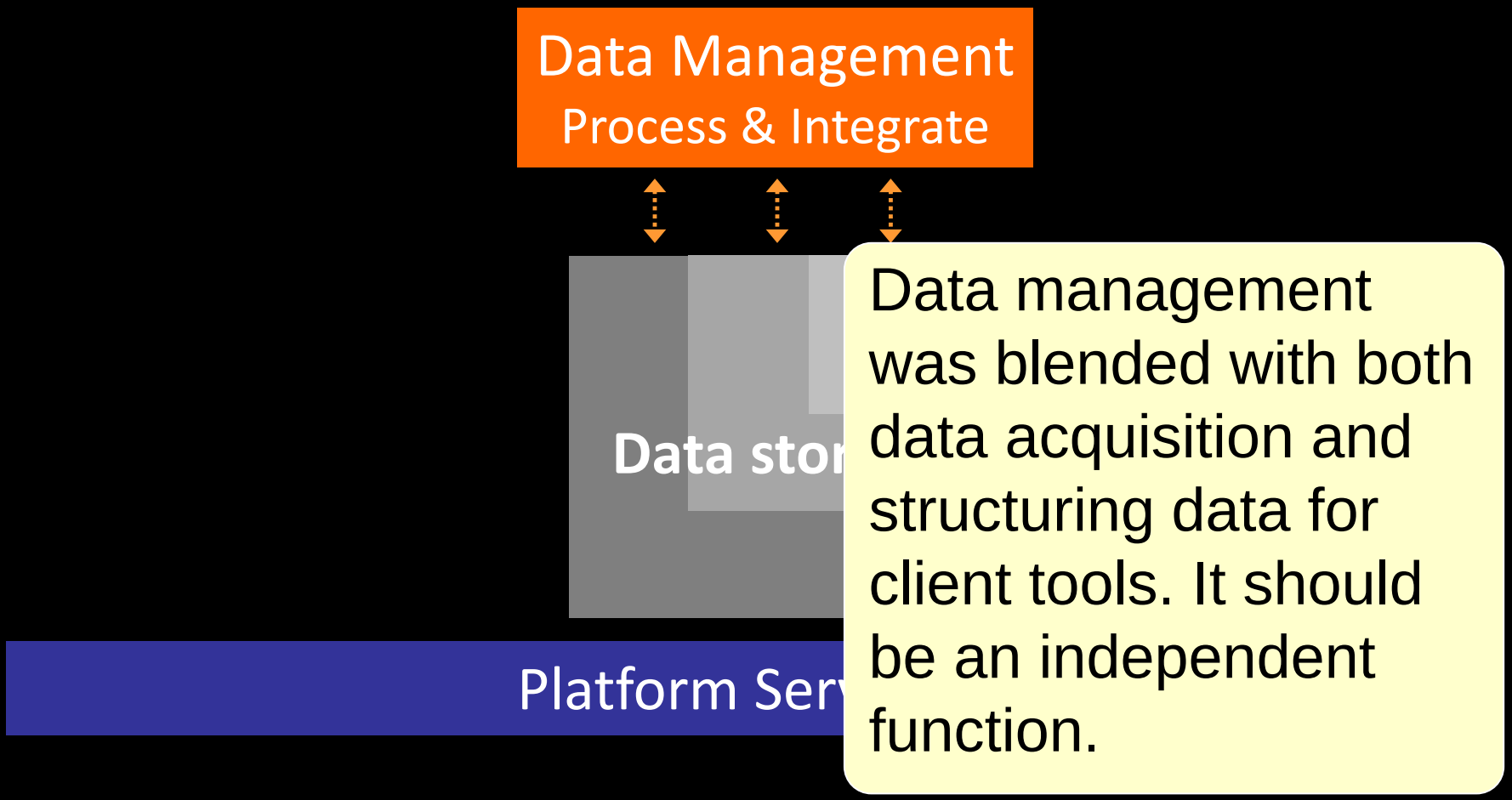
Data access is already somewhat separate today. Make the separation of different access methods a formal part of the architecture. Don't force one model.



Platform Services

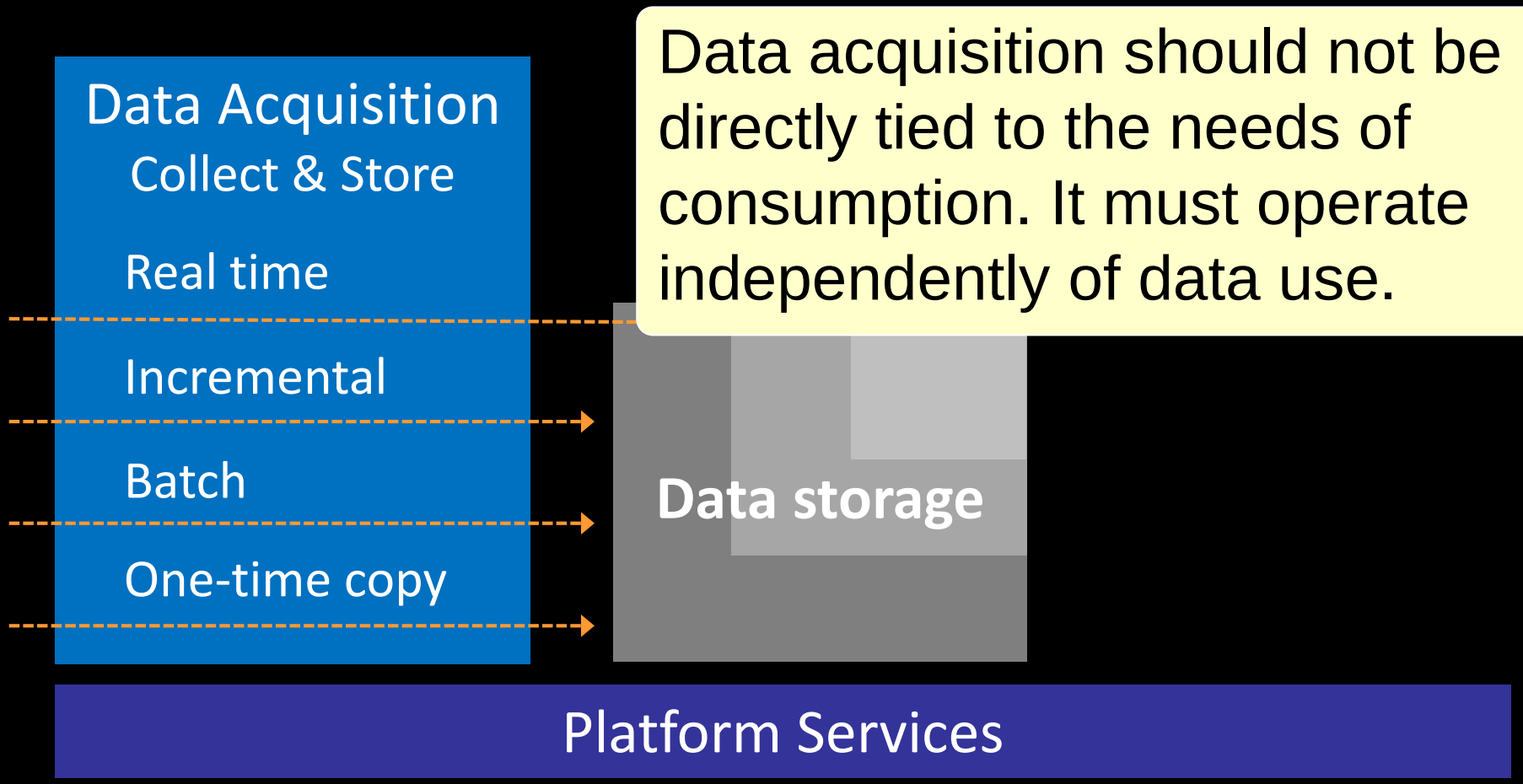
This separates BI from other uses of data, allowing each type of use to structure the data specific to its own requirements.

The goal is to decouple: solve the application and infrastructure problems separately



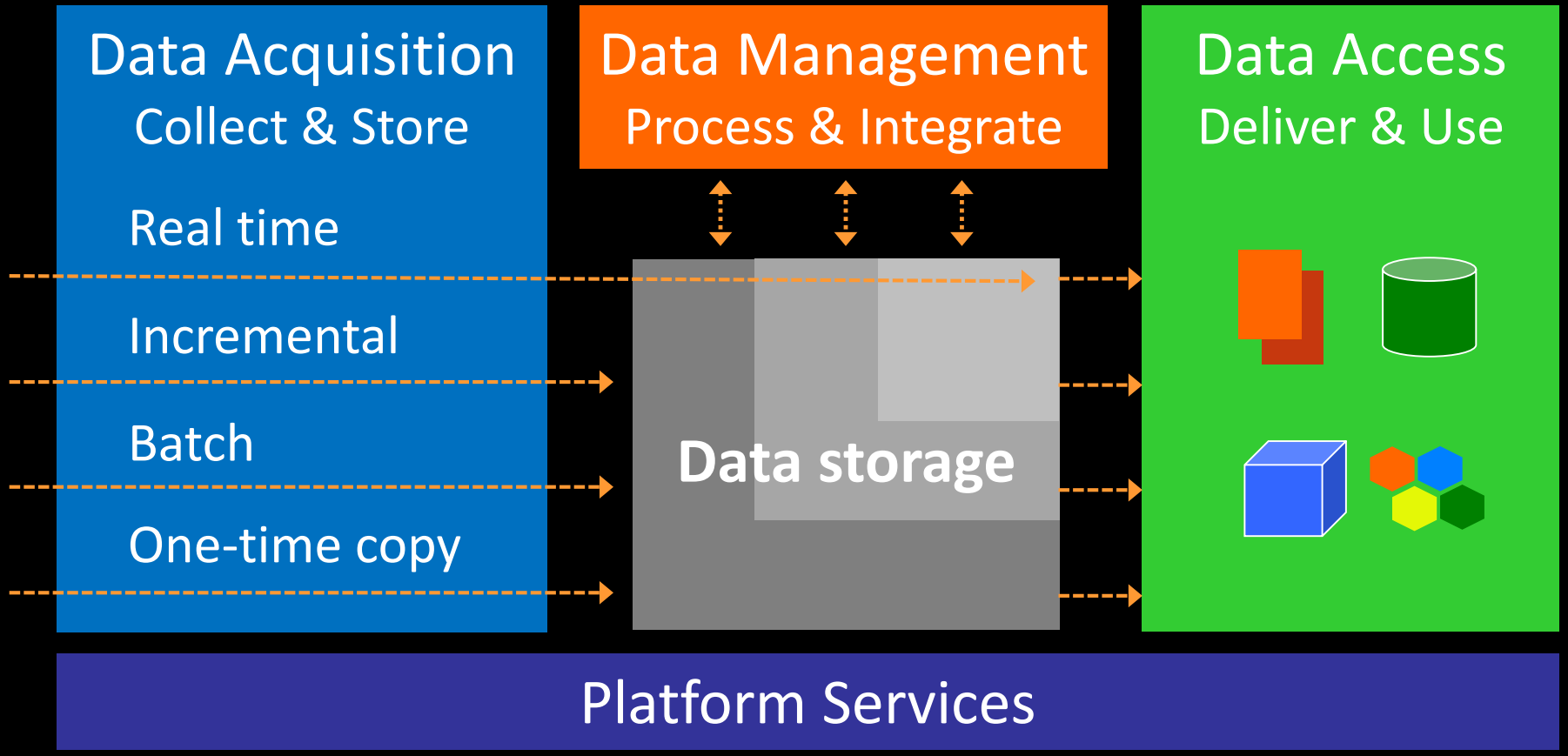
Data management should not be subject to the constraints of a single use

The goal is to decouple: solve the application and infrastructure problems separately



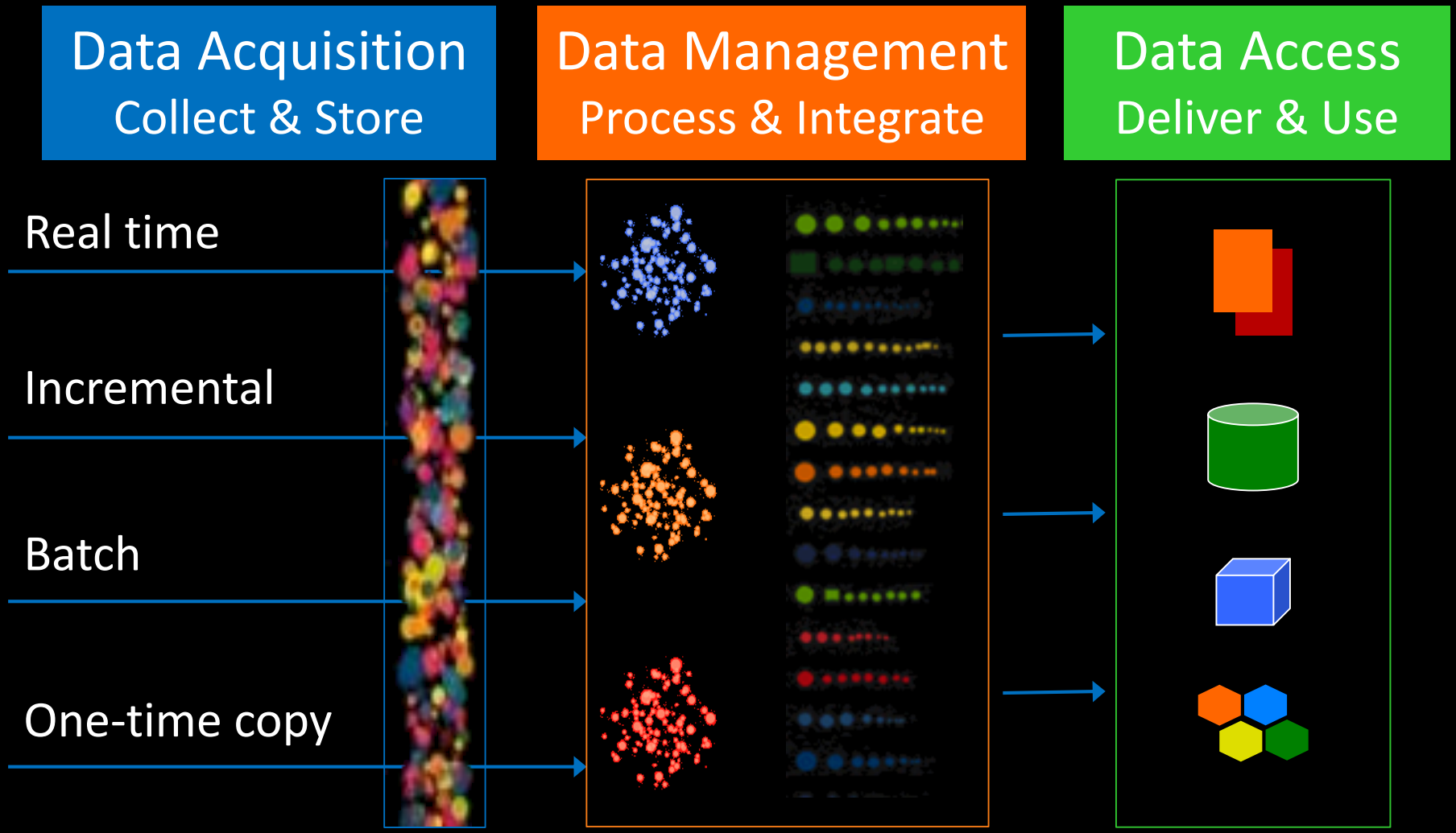
Data arrives in many latencies, from real-time to one-time. Acquisition can't be limited by the management or consumption layers.

The full analytic environment subsumes all the functions of the data warehouse, and extends them



The platform has to do more than serve queries; it has to be read-write.

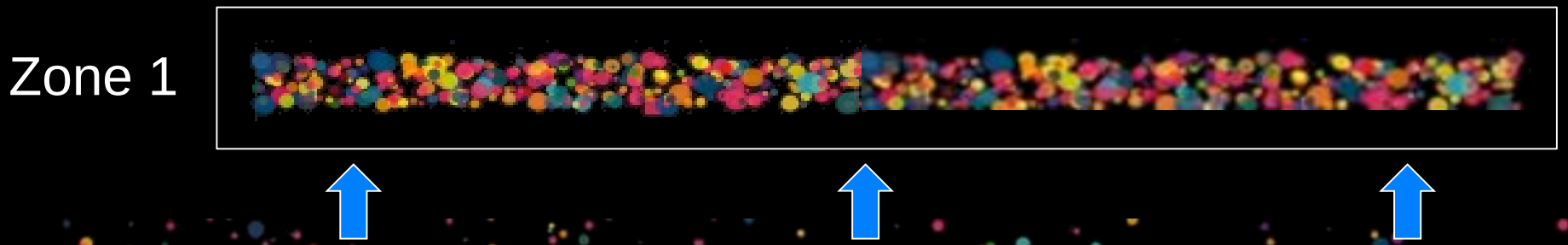
The data architecture must align with system components because each of them addresses different data needs



Separating concerns is part of the mechanism for change isolation

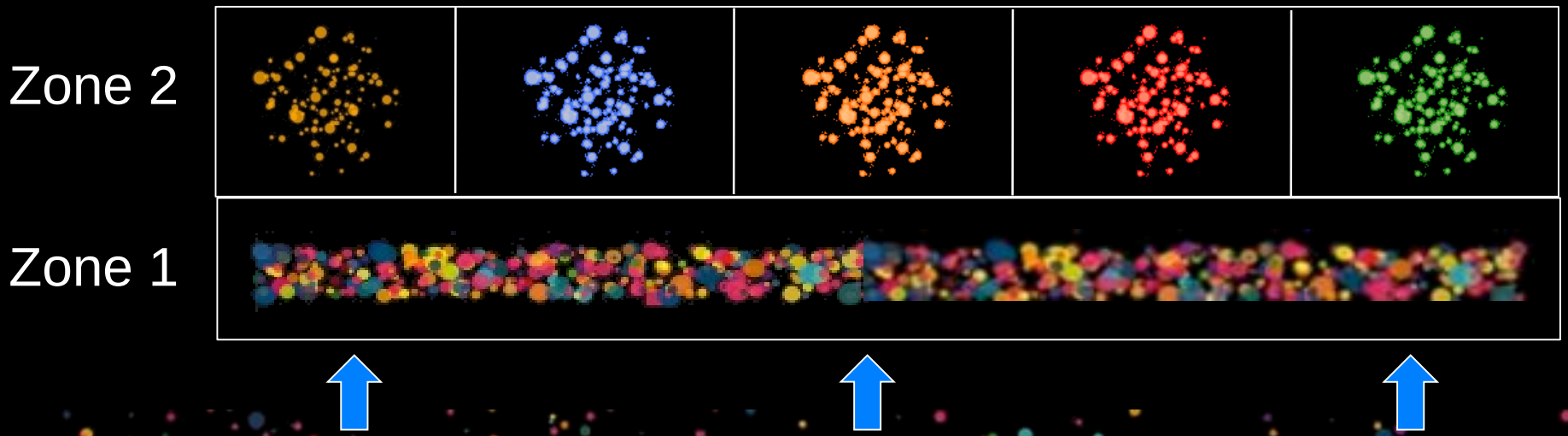
Zone 1: Acquistion

- Focus is only on collecting and tracking data
- Direct recording of data from sources
- Each dataset has as much schema as the source provides, could be explicit or implicit or none
- Foundation layer for all subsequent use



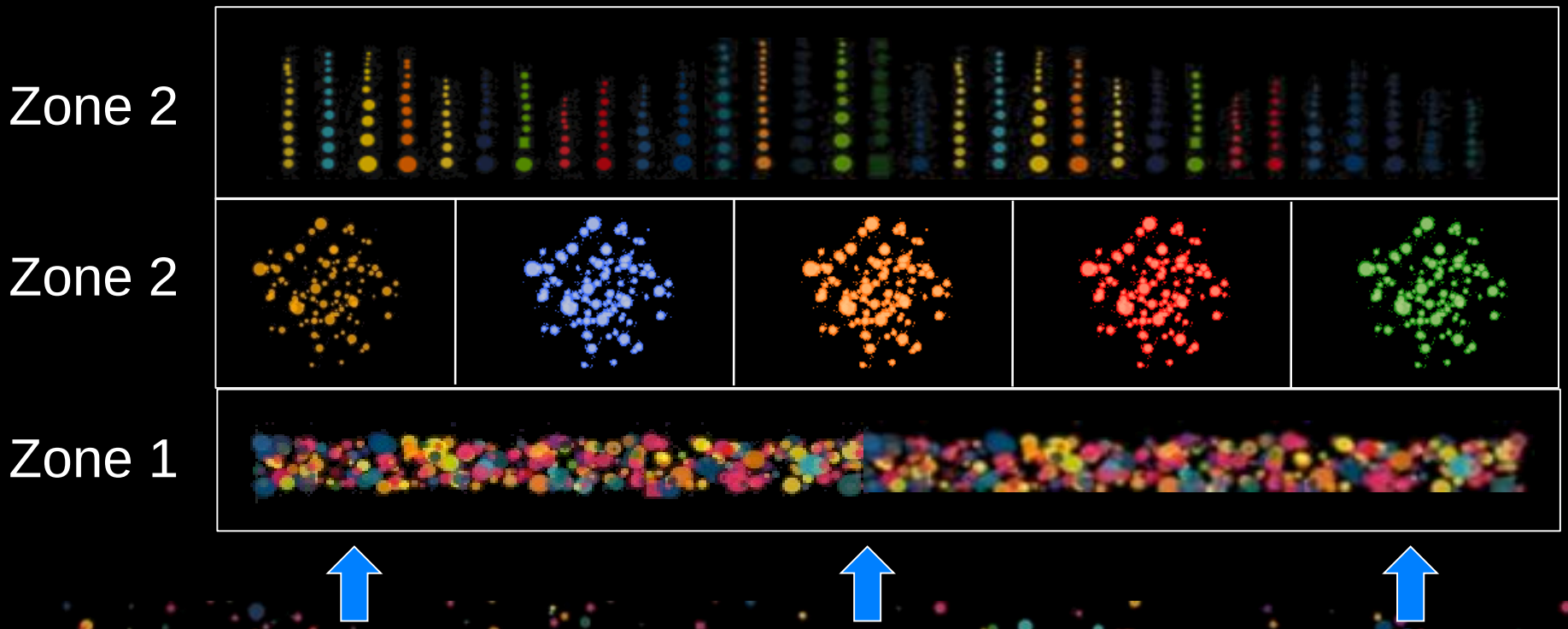
Zone 2: Integration - Standardized

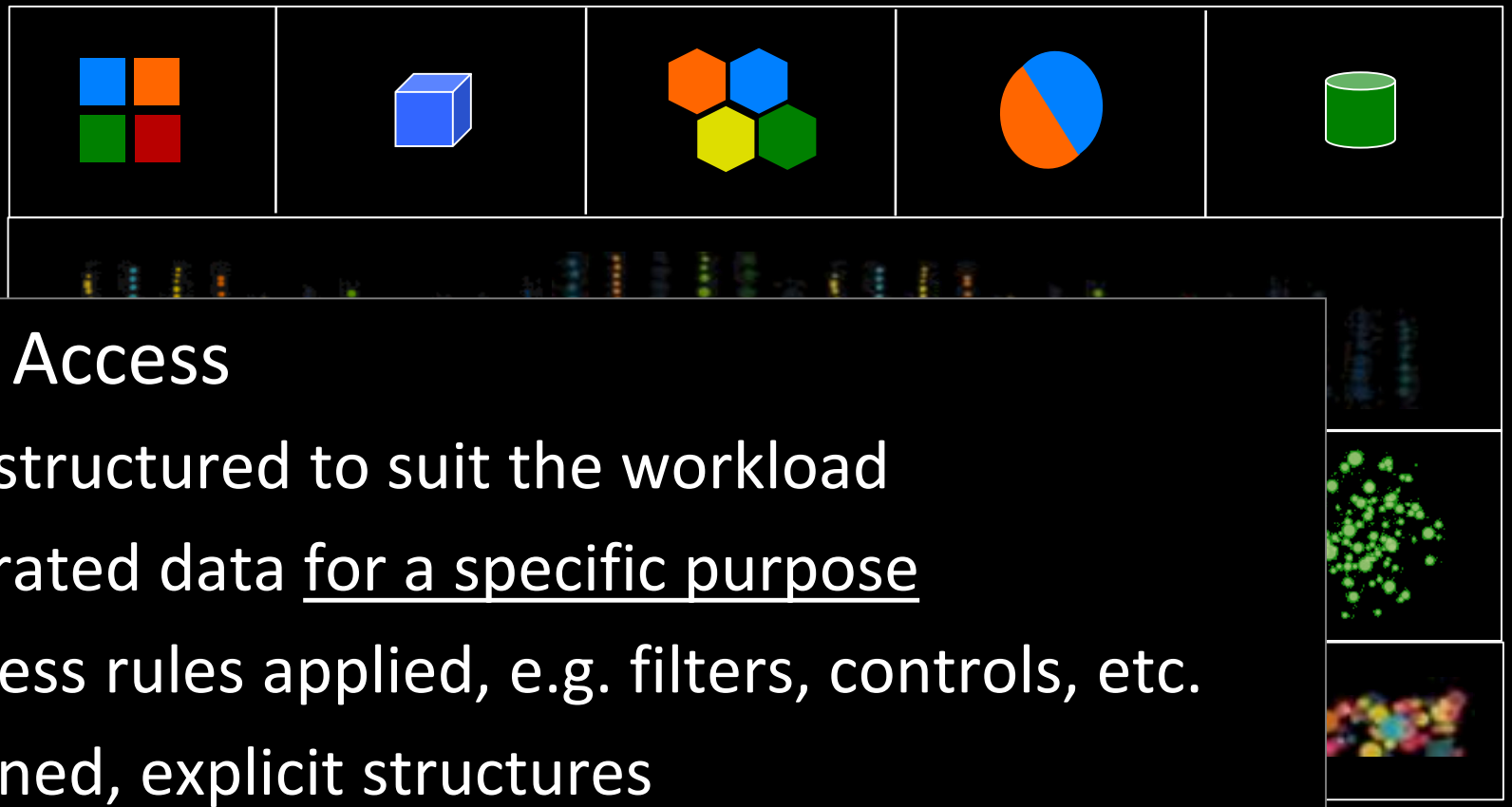
- Parsed and cataloged – all structure and form are made explicit so data can be accessed
- Namespace (e.g. keys) managed
- Common elements are standardized
- Datasets are profiled, indexed, available



Zone 2: Integration - Enhanced

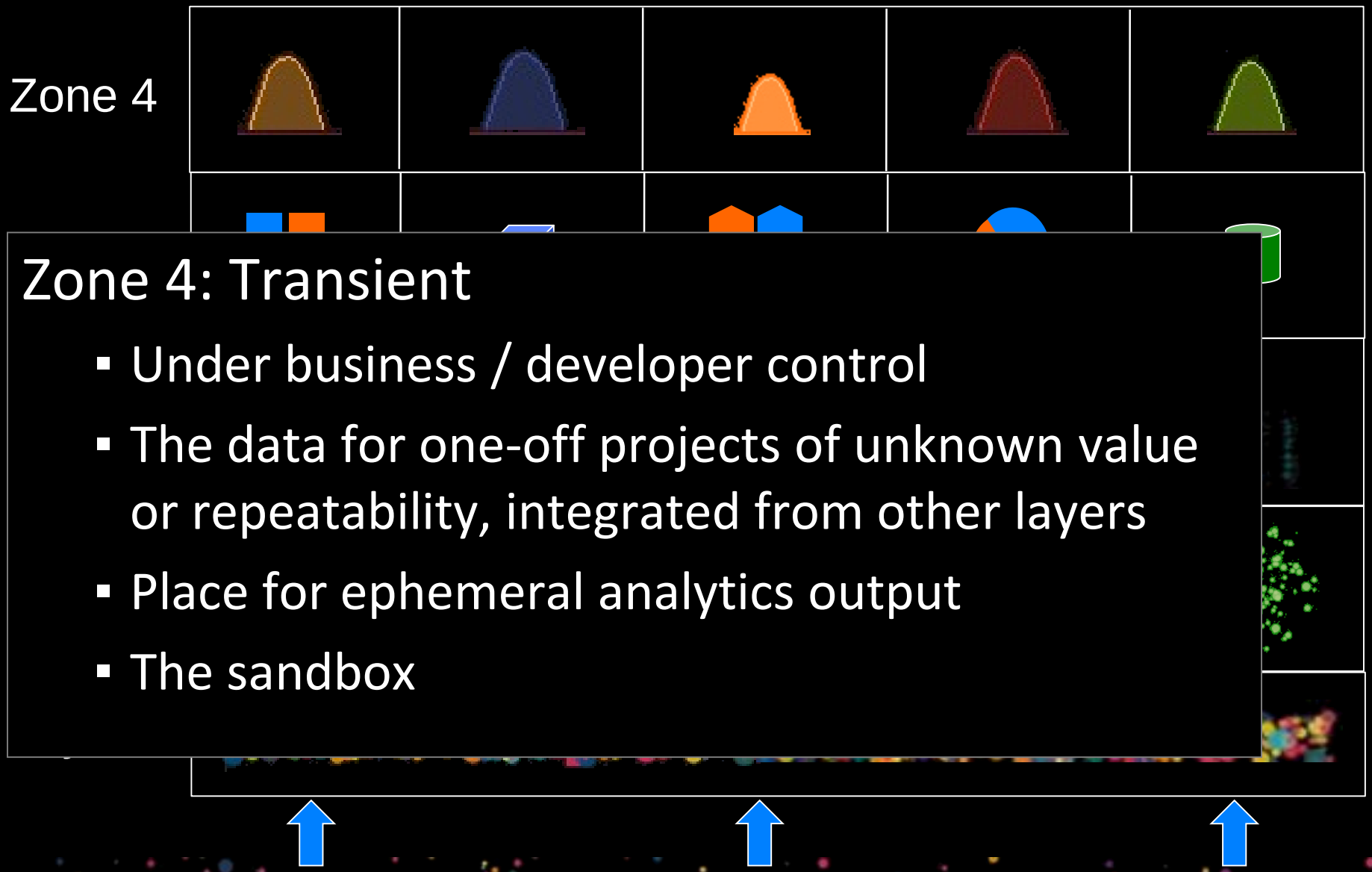
- Common shared KPIs, master datasets
- Extracted and derived data available, e.g. NEE output
- Data is linkable, labeled, possibly cleaned





Zone 3: Access

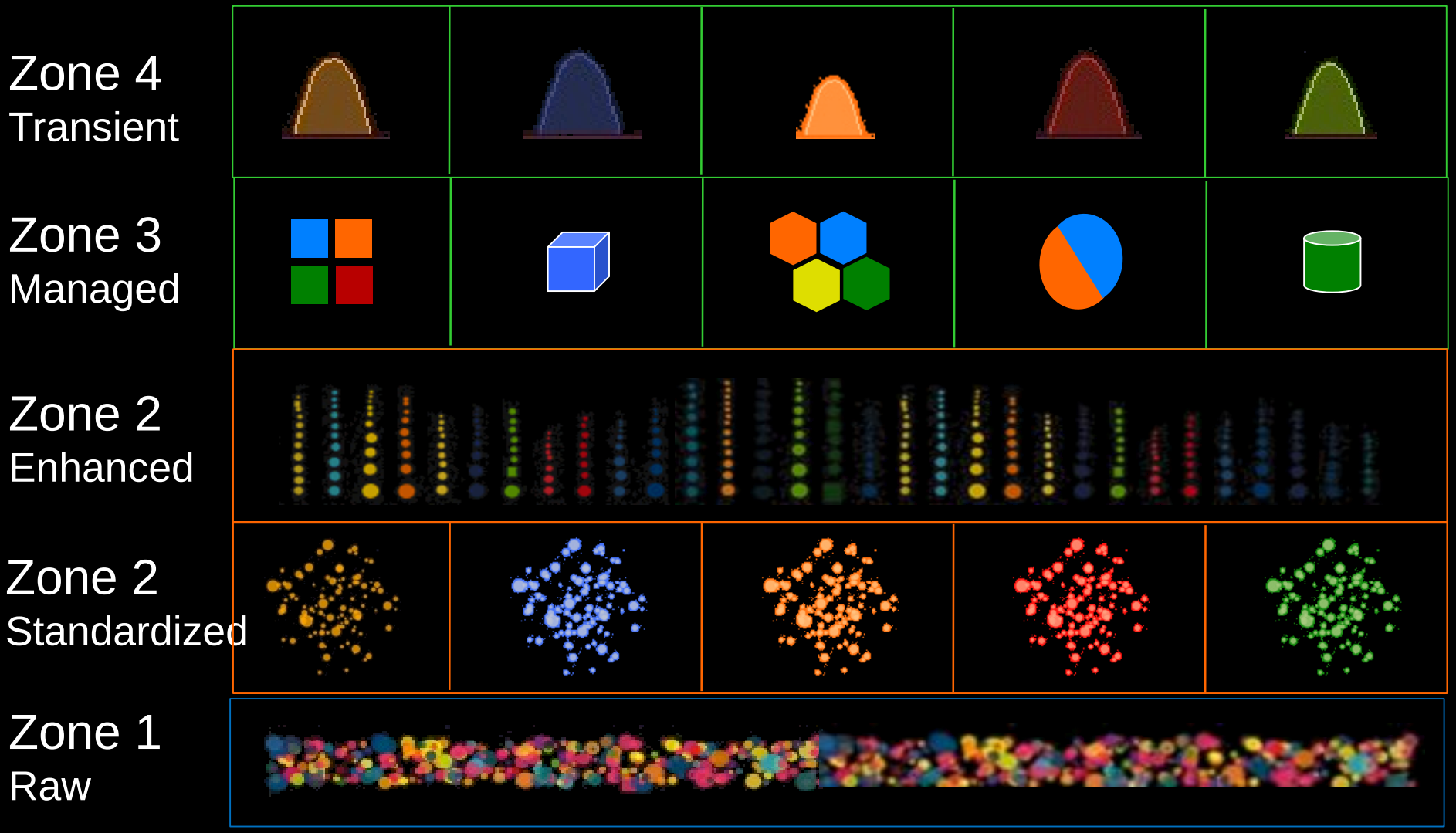
- Data structured to suit the workload
- Integrated data for a specific purpose
- Business rules applied, e.g. filters, controls, etc.
- Designed, explicit structures
- Generally repeatable use



Zone 4: Transient

- Under business / developer control
- The data for one-off projects of unknown value or repeatability, integrated from other layers
- Place for ephemeral analytics output
- The sandbox

Decoupled data architecture



Food supply chain: an analogy for analytic data

Multiple contexts of use, differing quality levels

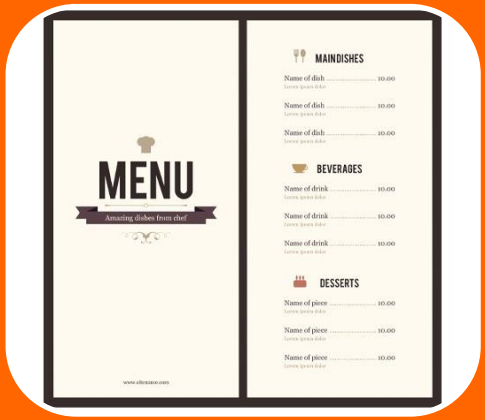
You need to keep the original because just like baking, you can't unmake dough once it's mixed.

The design focus is different in each area

Ingredients

Goal: available

User needs a recipe in order to make use of the data.



Pre-mixed

Goal: discoverable and integrateable

User needs a menu to choose from the data available

Meals

Goal: usable

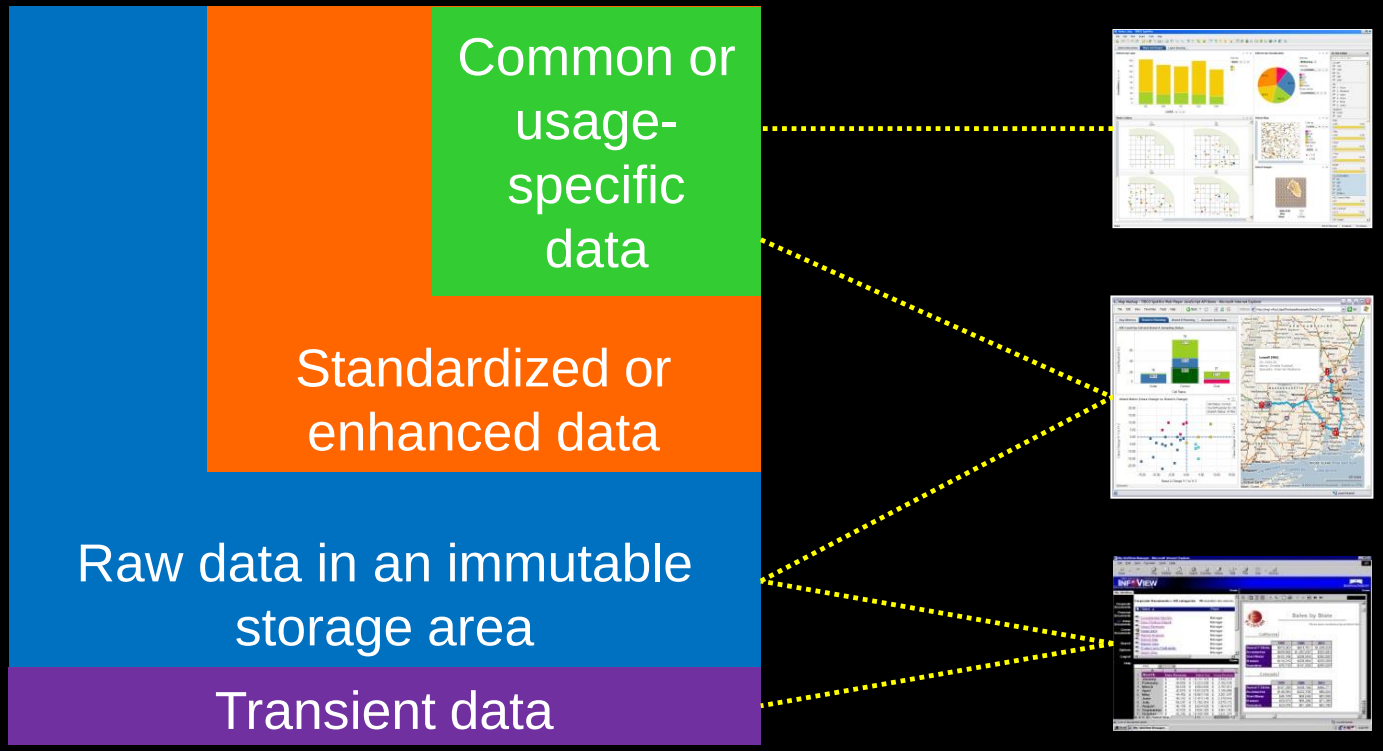
User needs utensils but is given a finished meal

The data is in zones of management, *not* isolating layers

Relax control to enable self-service while avoiding a mess.

Do not constrain access to one zone or to a single tool.

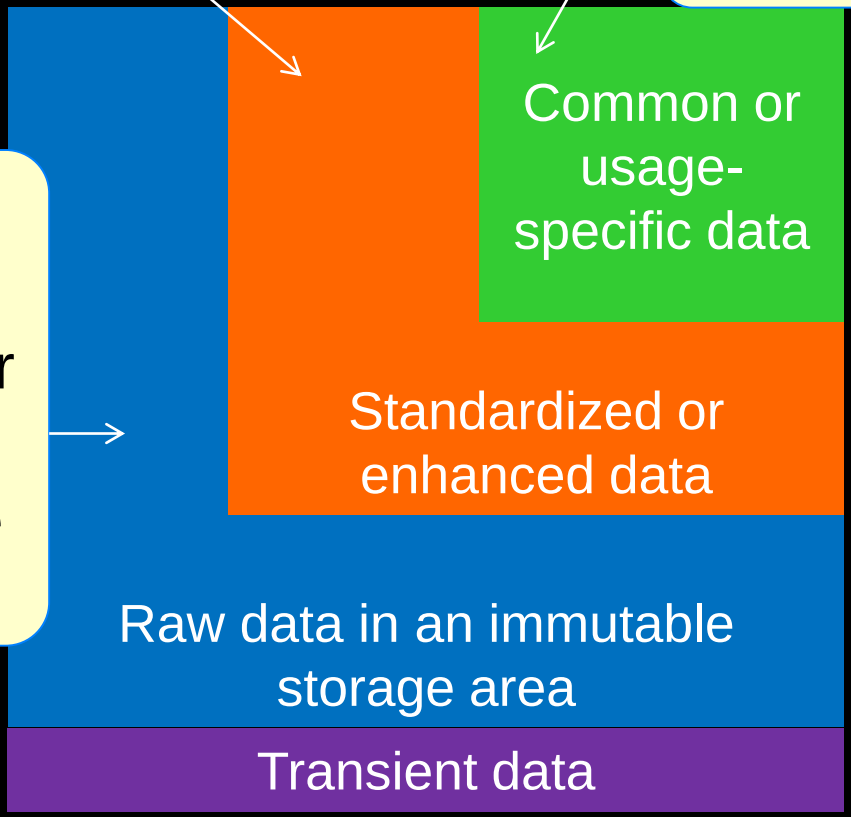
Focus on visibility of data use, not control of data.



This data architecture resolves rate of change problems

More effort applied to management, slower.

Optimized for specific uses / workloads. Generally the slowest change.



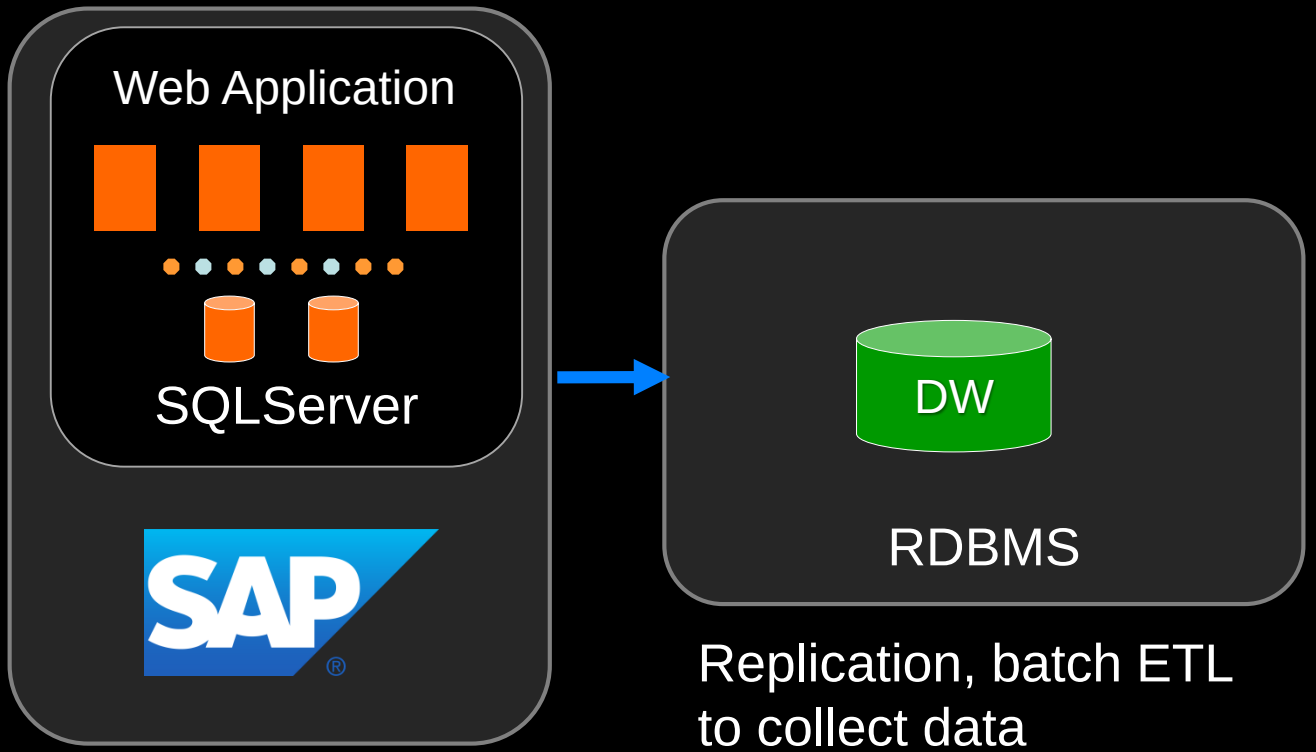
New data of unknown value, simple requests for new data can land here first, with little work by IT.

Not fast vs slow: *fast vs right*

Not flexibility vs control: *flexibility vs repeatability*

Agile for structure change vs agile for questions / use

Example: data environment, mid-size retailer



Replication, batch ETL to collect data

Standard retail / wholesale data warehouse model.

Nightly batch model from stage, contains ~ 1 TB of data

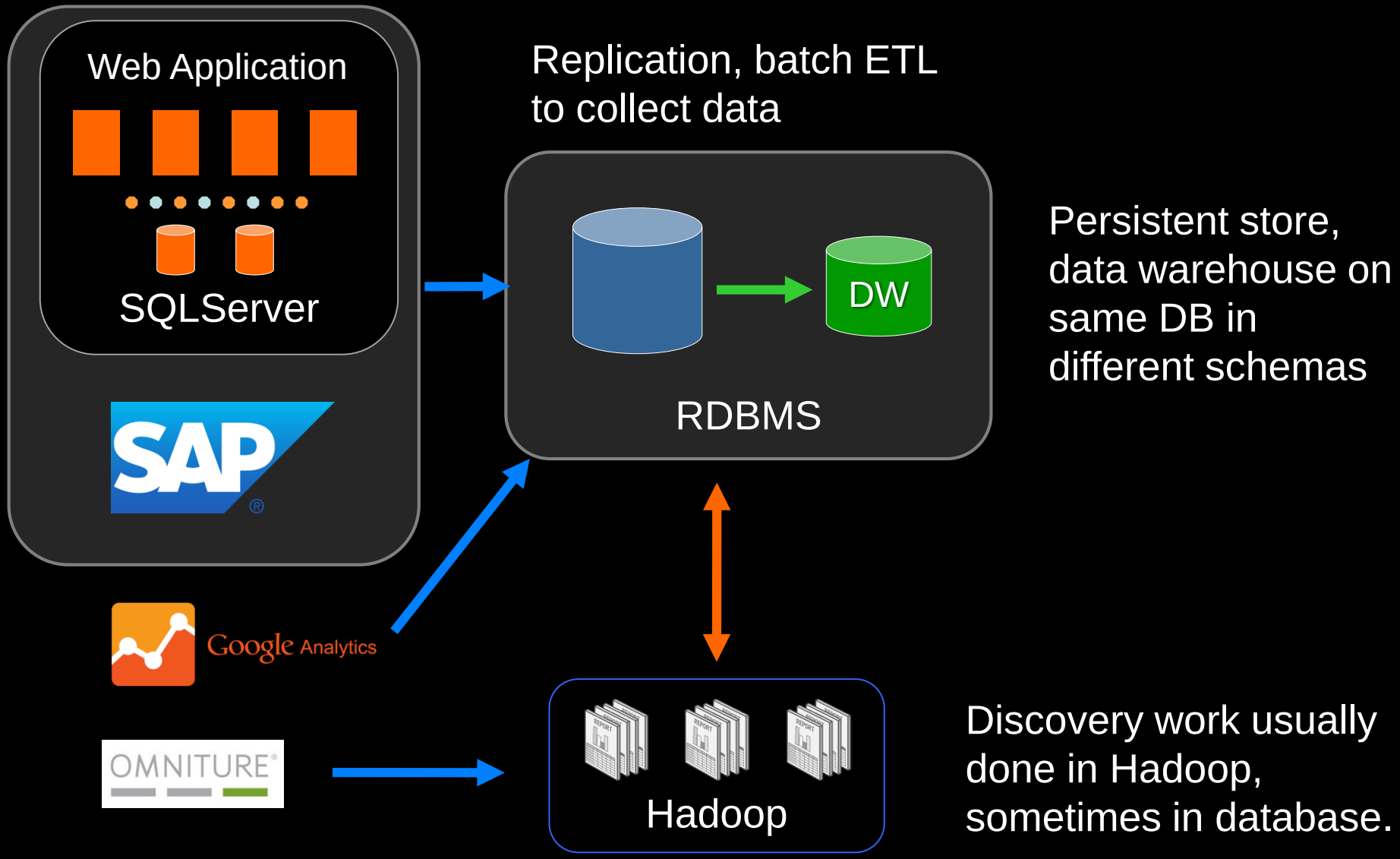
New requirement:

OMNITURE

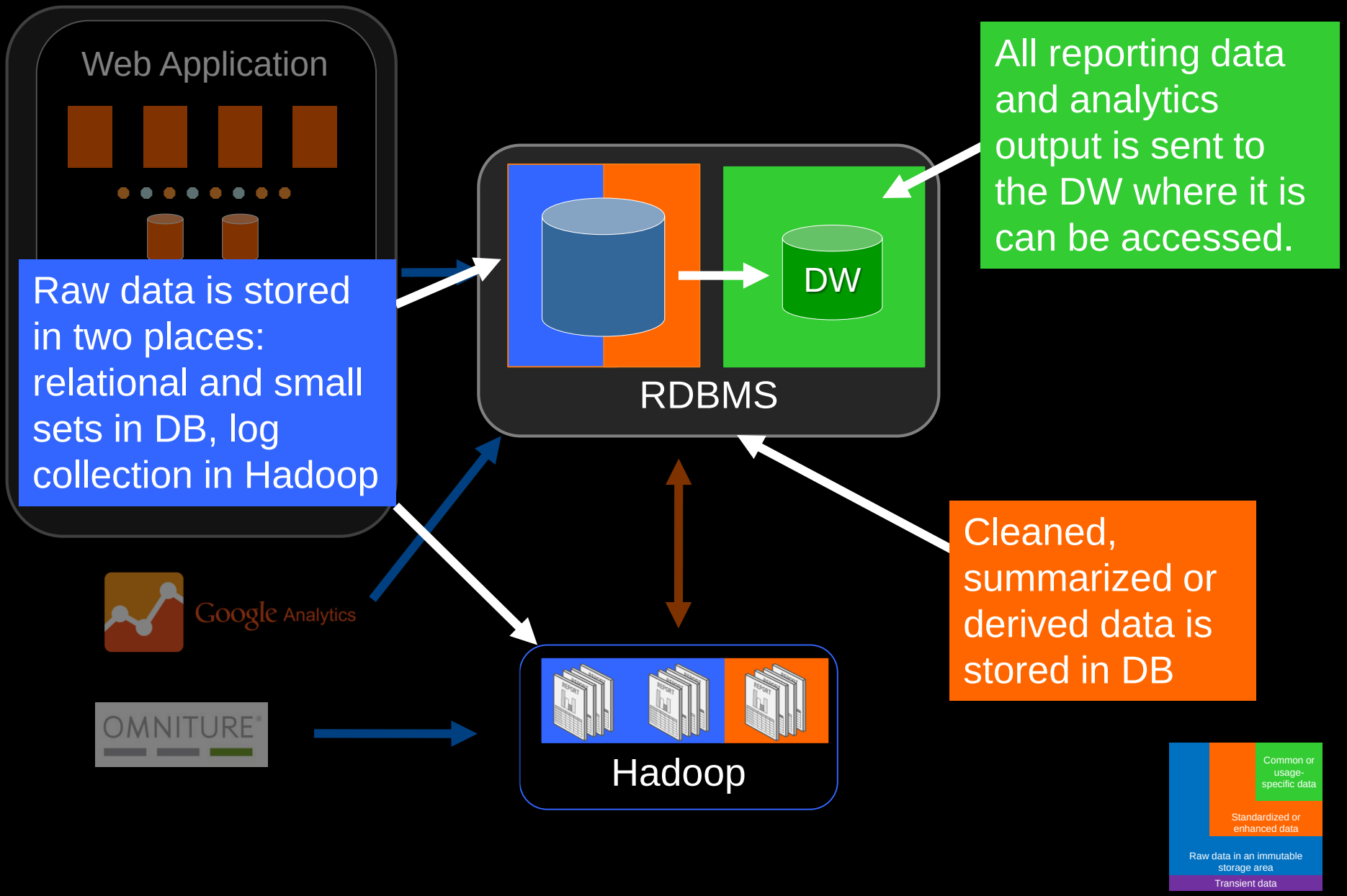
Load all the online marketing and web activity for use in analytic models (not simple web analytics)

Will add ~10 TB of data, continuous or hourly loads

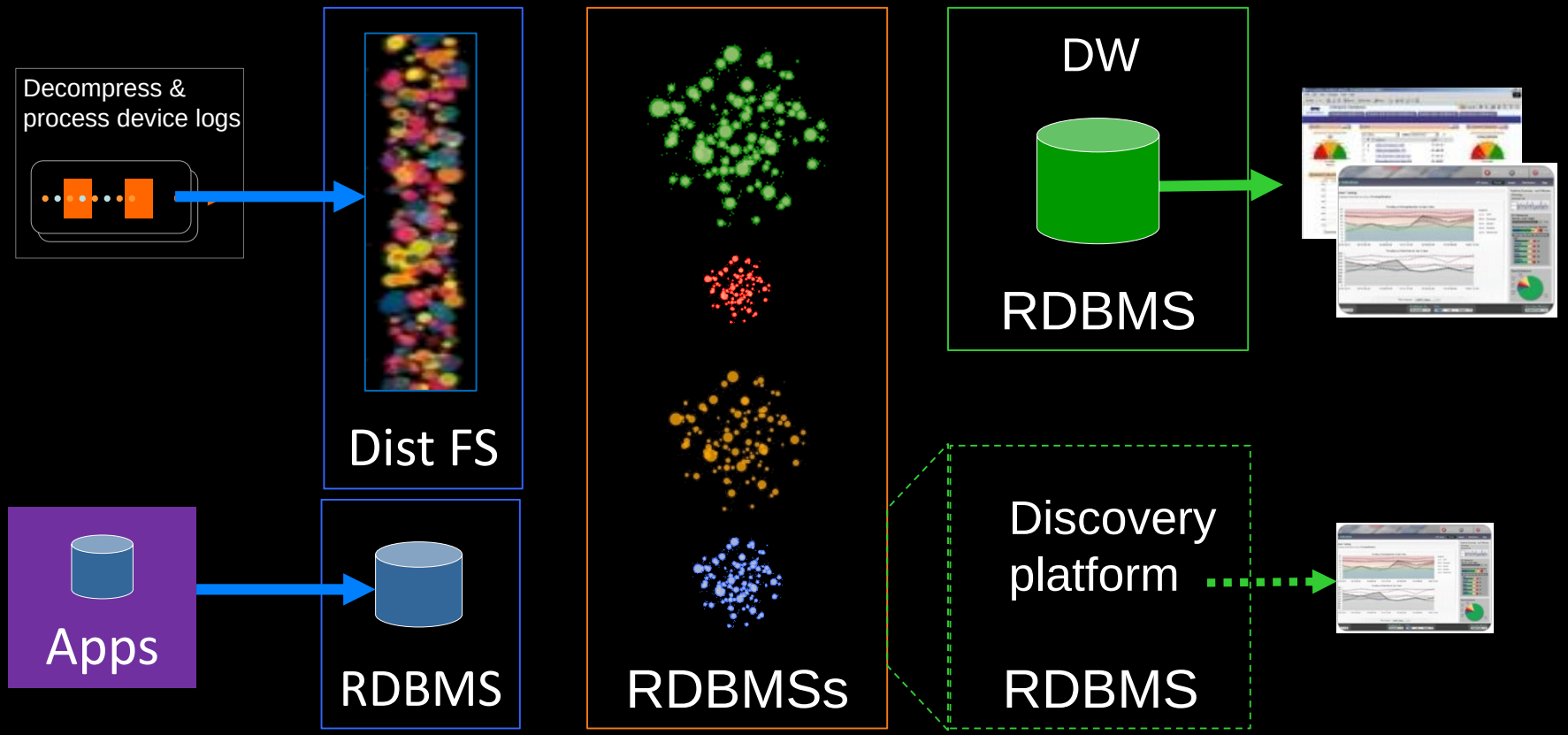
Example: data environment, mid-size retailer



This data architecture uses the 3 zone pattern



The concept of a zone is not a physical system. It's data architecture

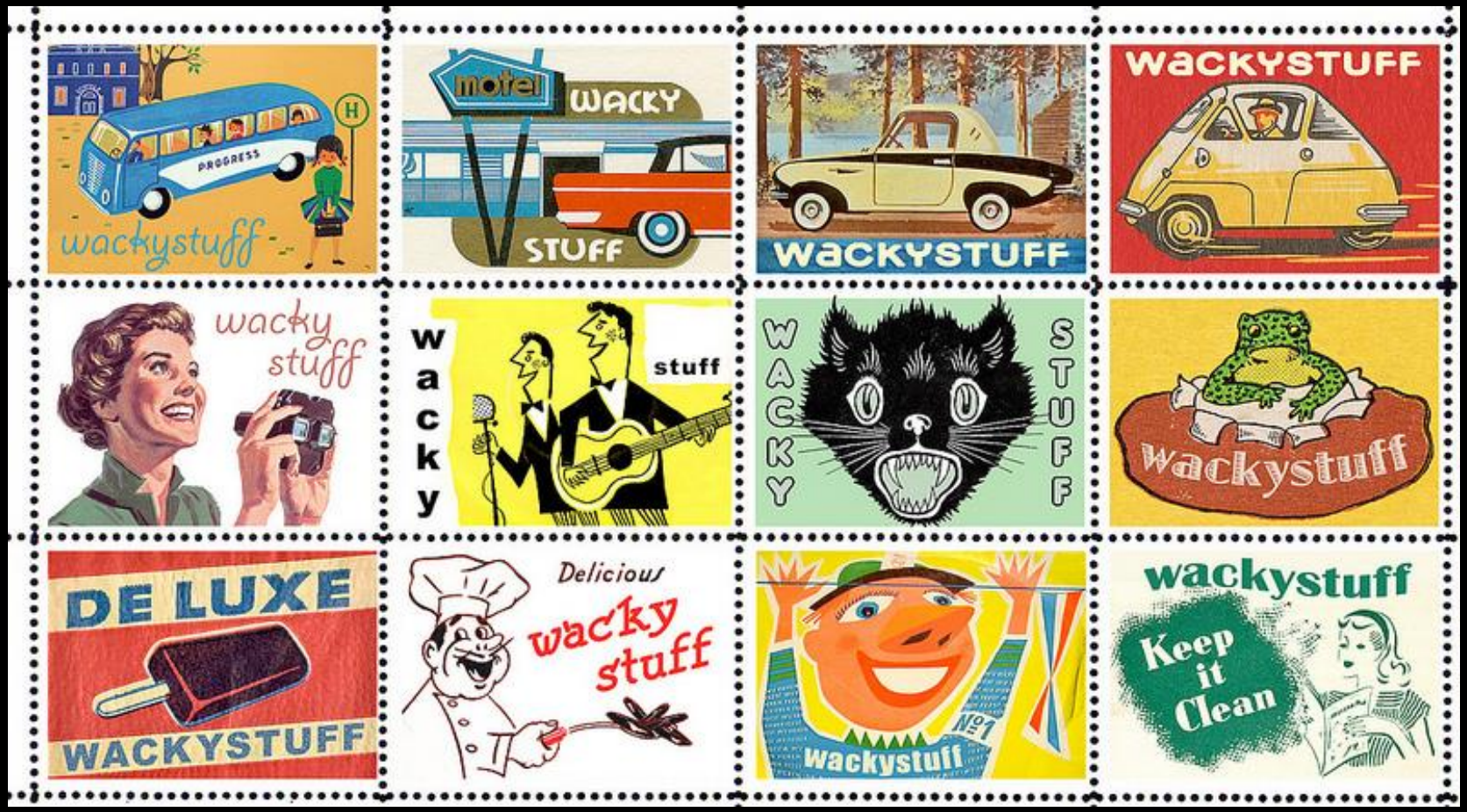


The biggest decision is to separate all data collection from the data integration from consumption.

Physical system/technology overlays are separate, depend on the specific use cases and needs of the organization.

Third Nature

What about the technology? Do I need an <X>?

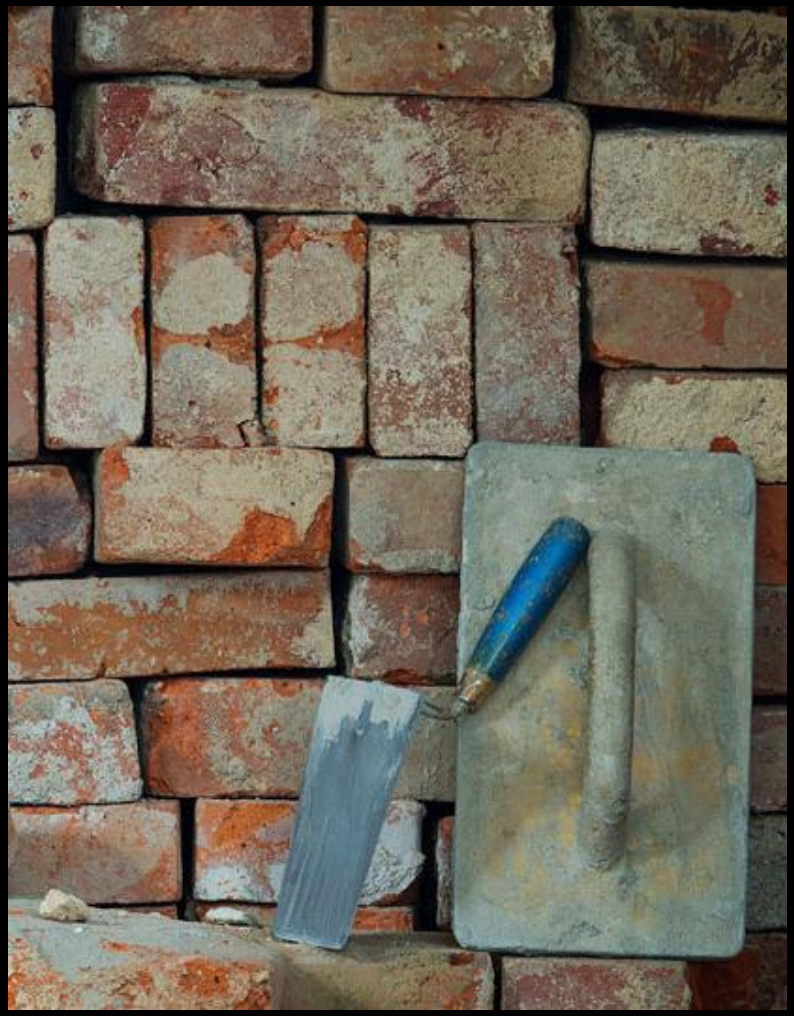


Third Nature

Blended Architectures Are a Requirement, Not an Option

- Data Warehouse + Data Lake
- On Premise + Cloud
- RDBMS + S3 + HDFS
- Commercial + Open Source

Bricks are not buildings



We don’t think this is equivalent to this

Architecture is not technology. It’s not a product you can buy.

Third Nature

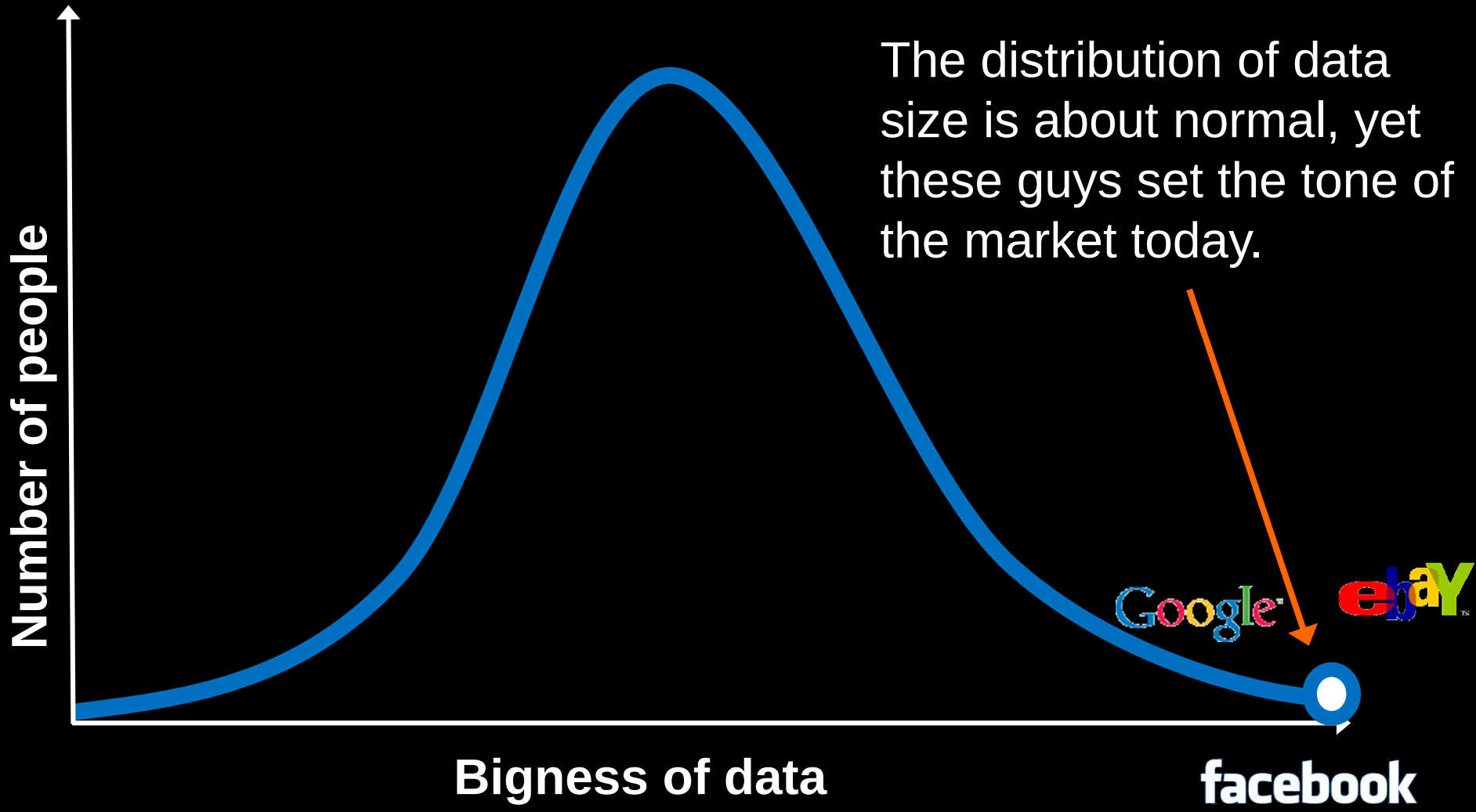
New tools and tool architectures are required

One tool for all jobs doesn't work any more*

** it never did, we just had fewer jobs*

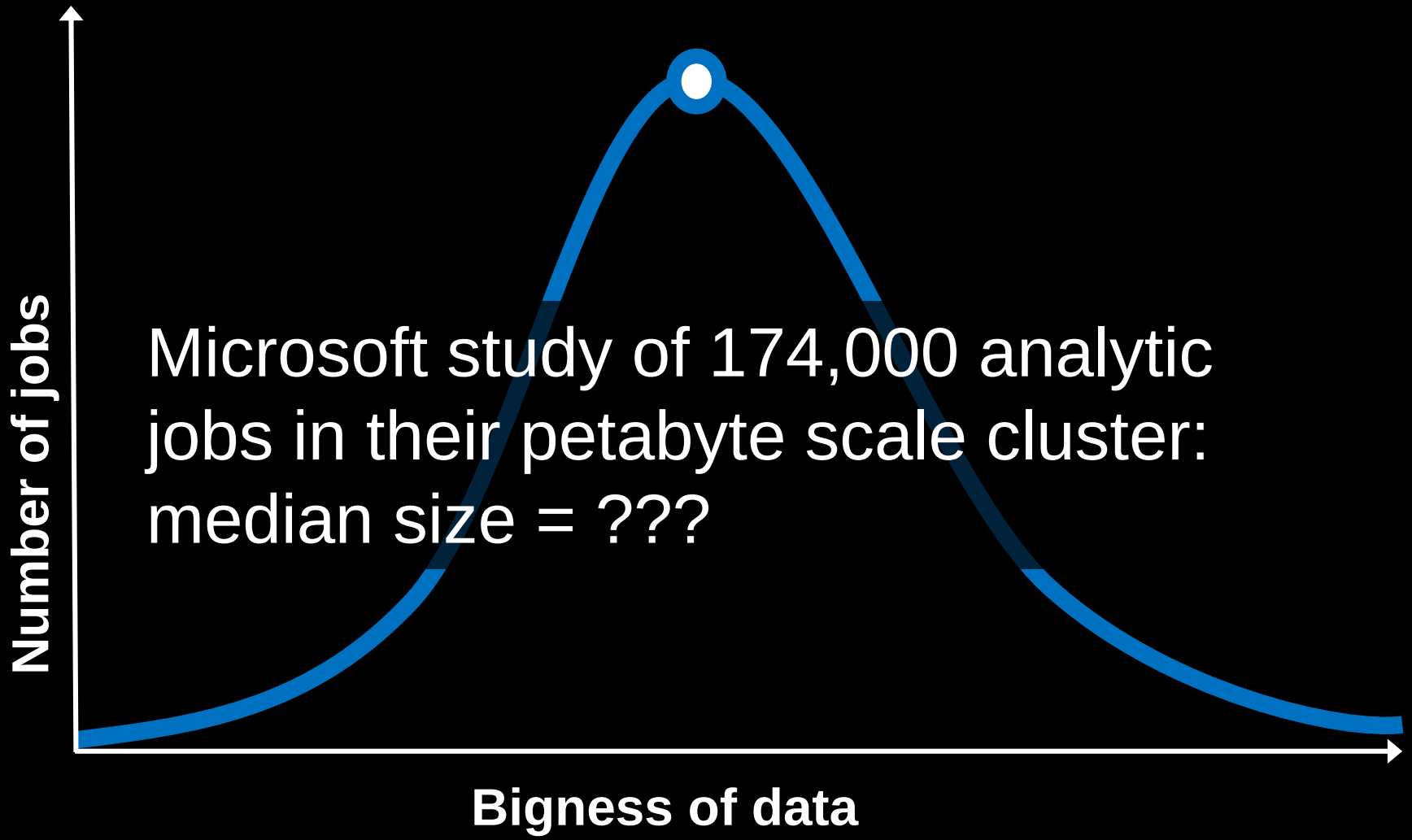
Third Nature

Beware overengineering: do you really need special new technology infrastructure?



Third Nature

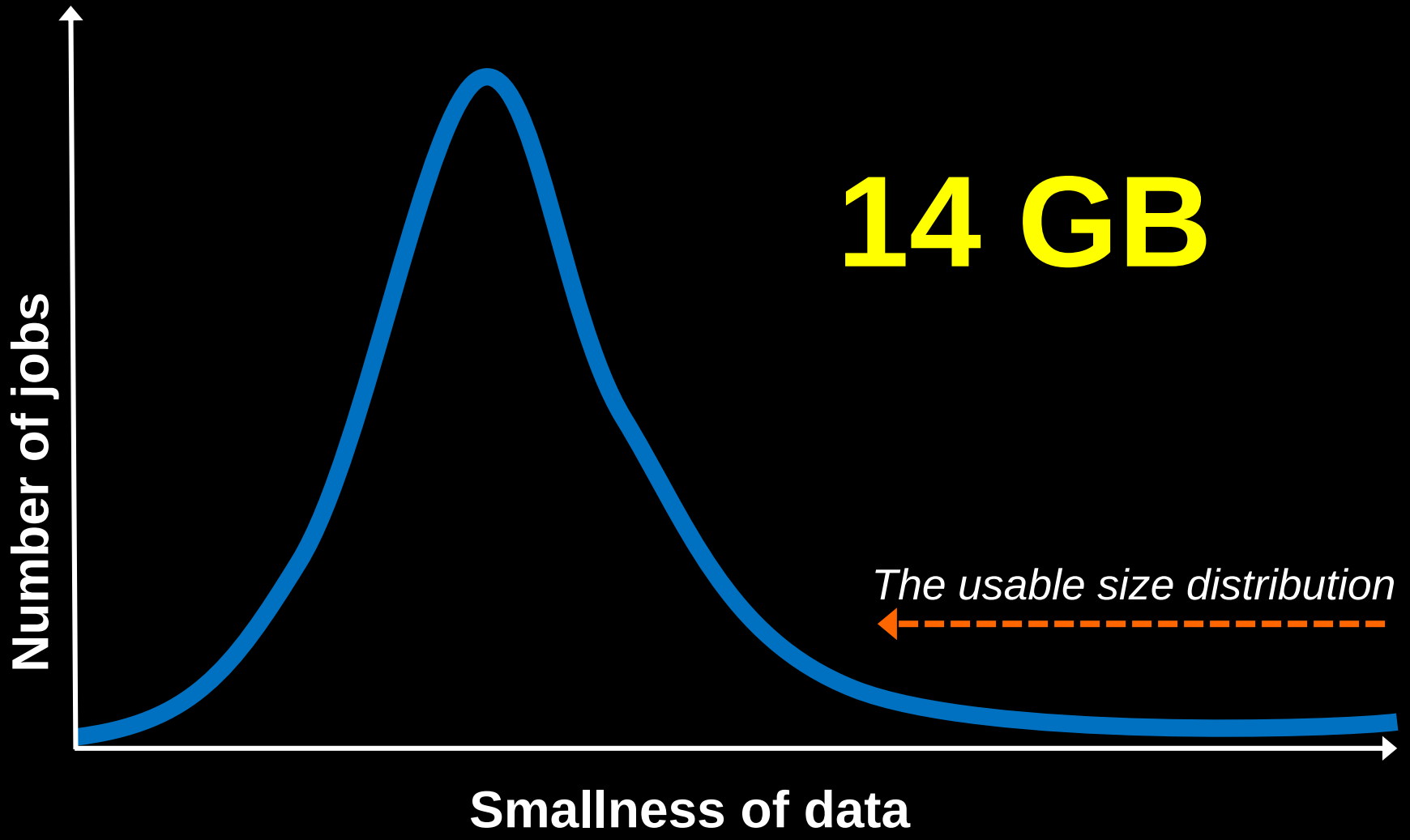
Analytics: This is really *raw data under storage*



Copyright Third Nature, Inc.

Third Nature

Working data for analytics most often not big



Third Nature

TANSTAAFL

When replacing the old with the new (or ignoring the new over the old) you always make tradeoffs, and usually you won't see them for a long time.

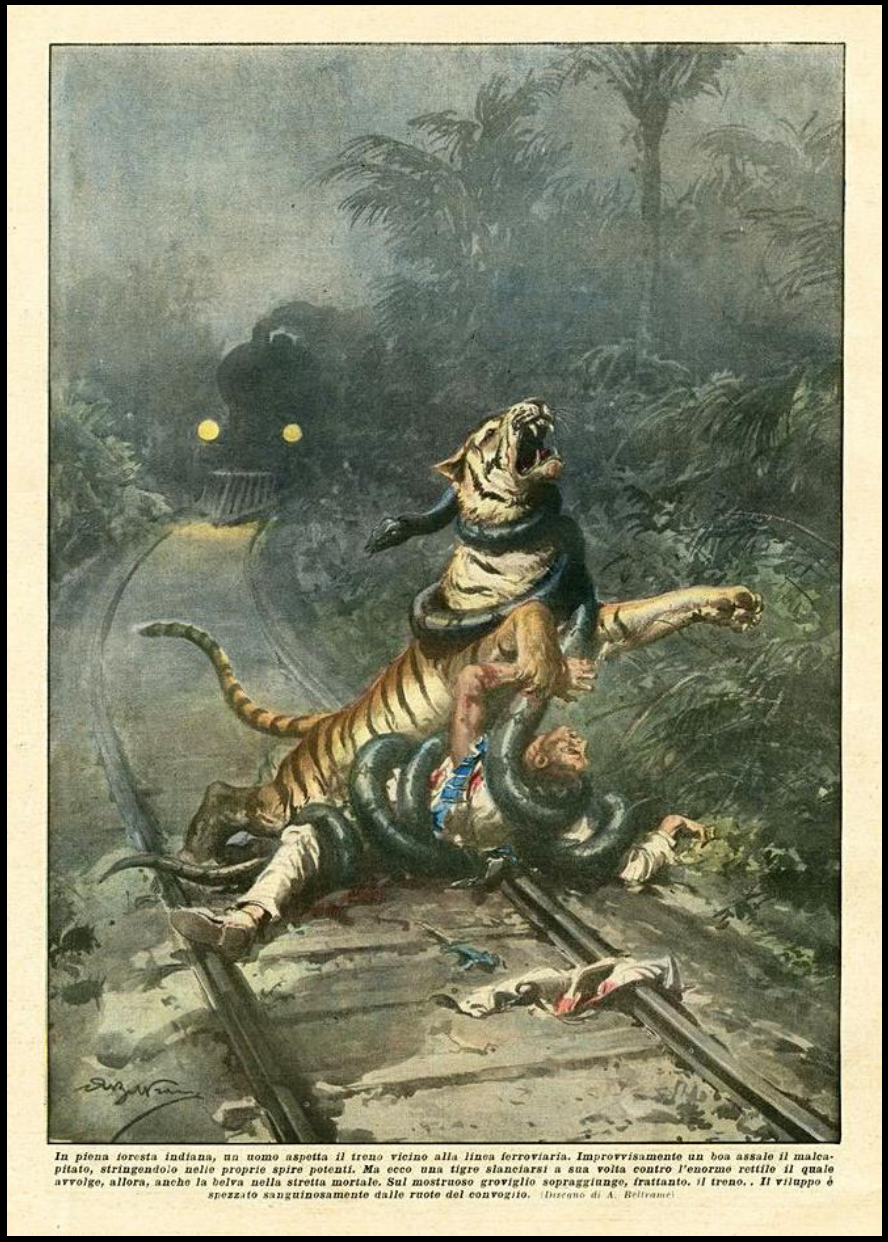
Technologies are not perfect replacements for one another. Often not better, only different.

Third Nature

Conclusion

1. It’s not about storing data, it’s about using it
2. Use drives architecture. Understand the uses, what you are designing for, to drive decisions.
3. Put data at the center, not technology. Don’t let the tech define what you can do or how you do it.
4. The death star is not the answer. The data model is not a flat earth. You are not building a monolith.
5. Know your history. Avoiding wheel reinvention saves time, money, careers.

Third Nature



In piena foresta indiana, un uomo aspetta il treno vicino alla linea ferroviaria. Improvvisamente un boa assale il malcapitato, stringendolo nelle proprie spire potenti. Ma ecco una tigre slanciarsi a sua volta contro l'enorme rettile il quale avvolge, allora, anche la belva nella stretta mortale. Sul mostruoso groviglio sopraggiunge, frattanto, il treno. Il viluppo è spezzato sanguinosamente dalle ruote del convoglio. (Disegno di A. Beltrame)

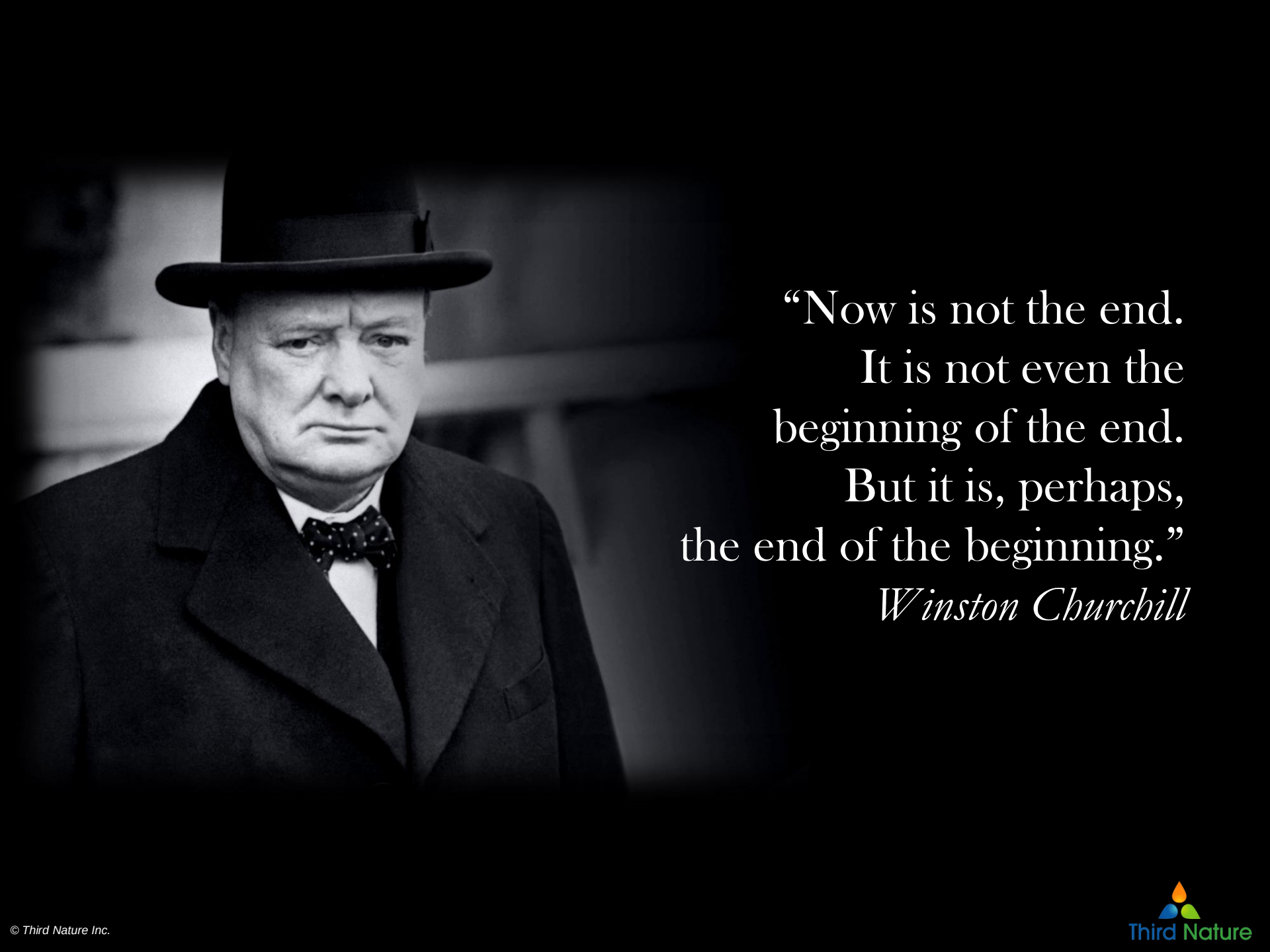
Manage your data (or it will manage you)

Data management is where developers are weakest.

Modern engineering practices are where data management is weakest.

You need to bridge these groups and practices in the organization if you want to do meaningful work with event stream data.

Third Nature



Third Nature

About the Presenter

Mark Madsen is the global head of architecture at Teradata, Prior to that he was president of Third Nature, a research and consulting firm focused on analytics, data integration and data management. Mark is an award-winning author, architect and CTO whose work has been featured in numerous industry publications. Over the past ten years Mark received awards for his work from the American Productivity & Quality Center, TDWI, and the Smithsonian Institute. He is an international speaker, chairs several conferences, and is on the O'Reilly Strata program committee. For more information or to contact Mark, follow @markmadsen on Twitter or visit http://ThirdNature.net

Third Nature

Todd Walter

Chief Technologist - Teradata

- Chief Technologist for Teradata
- A pragmatic visionary, Walter helps business leaders, analysts and technologists better understand all of the astonishing possibilities of big data and analytics
- Works with organizations of all sizes and levels of experience at the leading edge of adopting big data, data warehouse and analytics technologies
- With Teradata for more than 30 years and served for more than 10 years as CTO of Teradata Labs, contributing significantly to Teradata's unique design features and functionality
- Holds more than a dozen Teradata patents and is a Teradata Fellow in recognition of his long record of technical innovation and contribution to the company

TERADATA.