PORTfolio Analytics is a Big Data problem

is it?

All and only my personal views here

2/5/2019
Luca S.V. Spampinato – Bloomberg L.P.
lspampinato1@bloomberg.net

Bloomberg

Always start with a quotation

Yngve Slyngstad, since 2008 CEO of the world's biggest sovereign wealth fund, Norges Bank Investment Management (> $1Trillion AUM)
Bloomberg News Interview 02/02/2019

BM: What do you think is different today in terms of managing money vs. how it was when you started off at the fund in 1998 or even earlier at Storebrand?
YS: It is still about information processing, but the amount of information that is available is of course increasing every year, and the frequency of that information is just getting faster and faster. You have to cut through that and find what is essential. With this kind of a skill set it is very difficult to see who has got it and who hasn't got that ability, but I think it's one way of distinguishing.

BM: How is the advance of technology and AI changing investing?
YS: My own guess is that it's going to dramatically change quantitative investing and particularly risk-factor investing, probably more so than traditional active management.

AI in portfolio analytics is running along mainstream

Optimization in various flavours is a best hit

But there are actual, production ready, "side" results:

- x Liquidity analysis
- x Forecast of cash flows
  (subscriptions, and redemptions in Mutual Funds)

One of the pros* of wide spreading AI in finance
is that it brings focus on Data Processing

*(the only?)

The Analytics Pyramid

AI

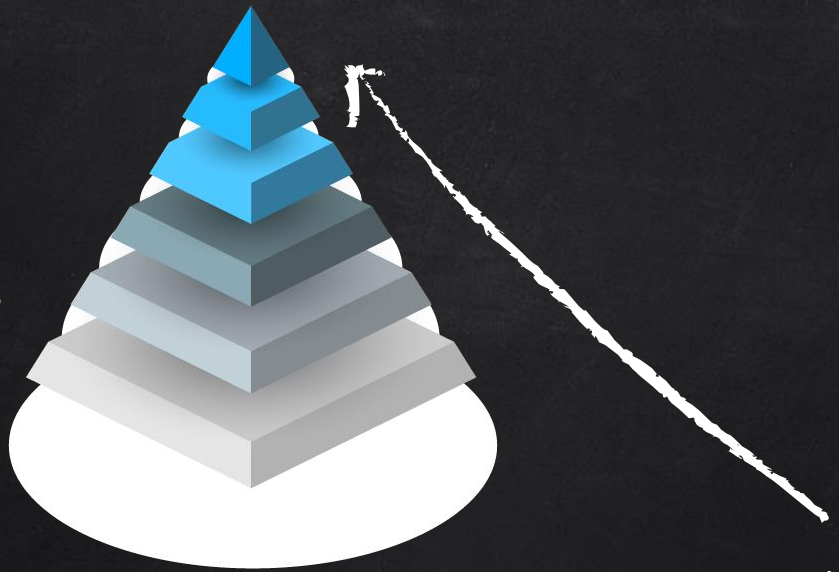
ML

Data Science

Analytics

Data Engineer

Big Data



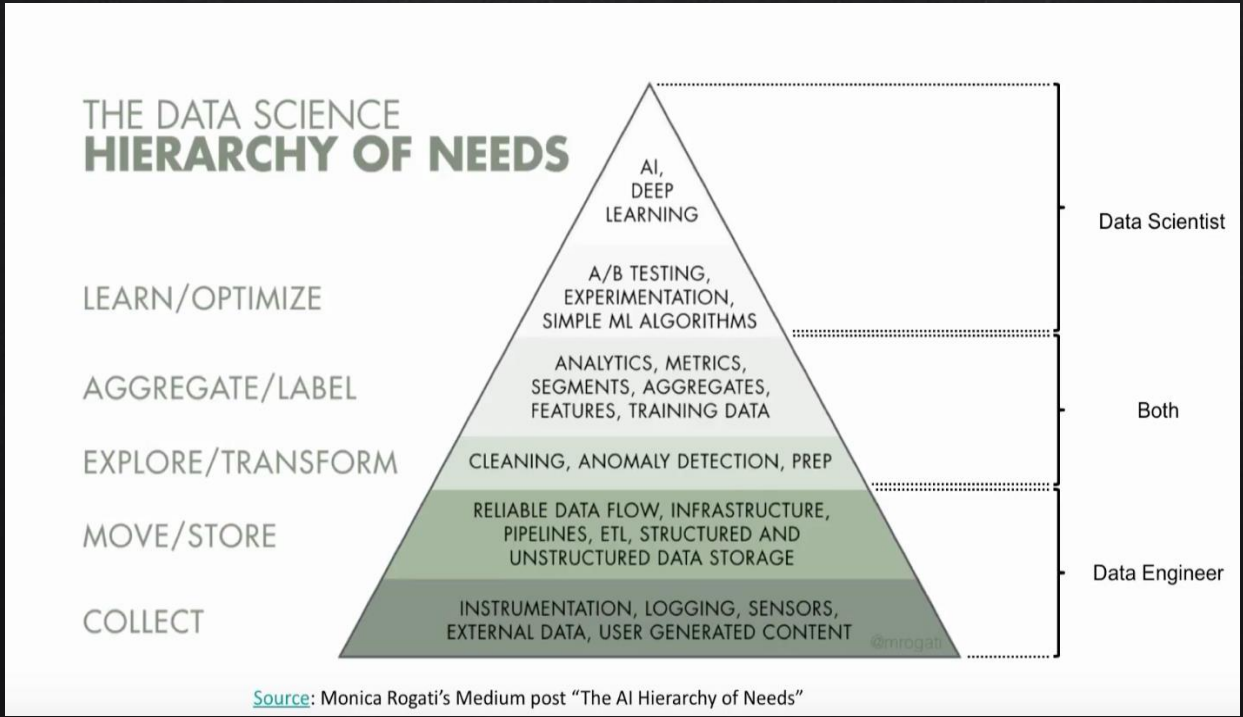
Portfolio Analytics is a Data Management problem anyway:

- ✗ Heterogeneous Data Sources
- ✗ Size of Portfolios
- ✗ Analytics demand

...and climbing up the Pyramid cannot be avoided

New roles

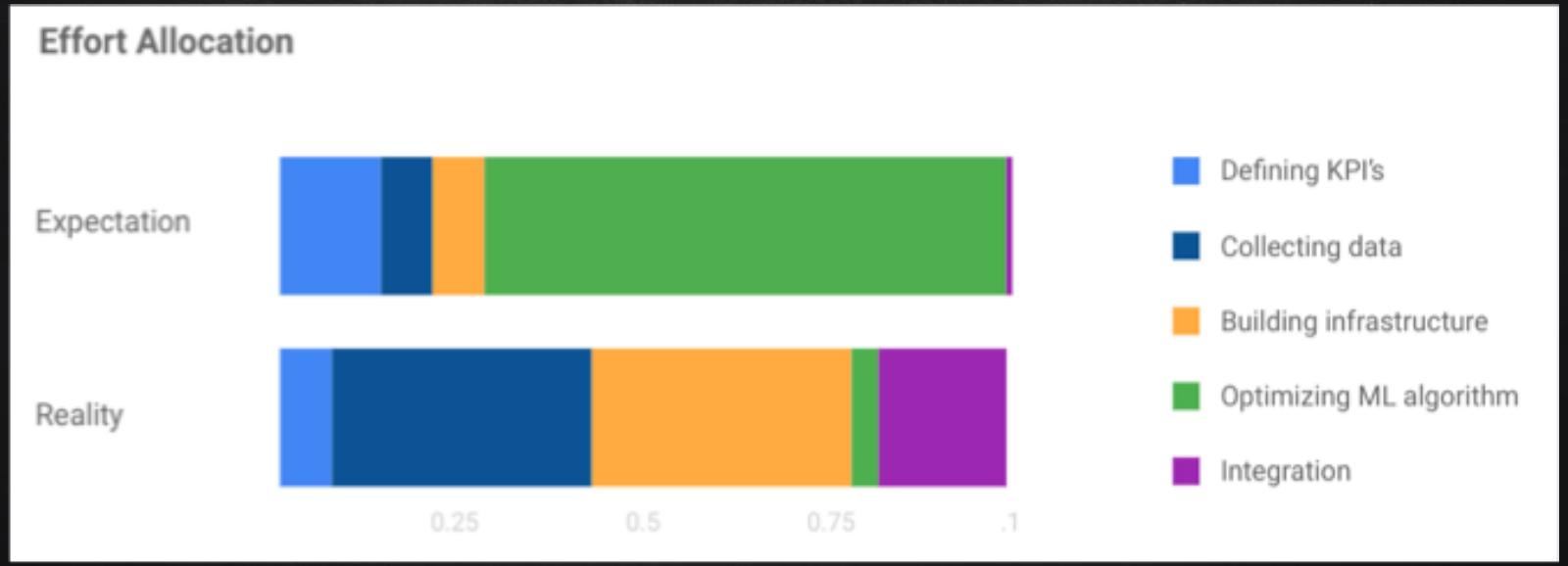
and, by the way, Data Scientists cannot be left alone



Source: Monica Rogati's Medium post "The AI Hierarchy of Needs"

The Surprise

ML projects' pain, in retrospective



... does this sound familiar?

Expanding Data Universe 1 – Analytics demand

- Asset owners sophistication, at any level (e.g. Risk Monitoring is the word in wealth management)
- Regulation

- Scenario Analysis (hundreds of thousands of scenarios, full repricing, for VaR , Asset Liability Modeling, etc.
- Factor based analysis (thousands of factors)
- Multi-factor, multi-dimension Performance&Risk Attribution

Expanding Universe 2 – Number, Size and Complexity of Portfolios

Number

- ✗ Wealth management
- ✗ Proliferation of funds

Size

- ✗ Asset owners aggregation
- ✗ Custom, structured Benchmarks

Complexity

- ✗ Pervasive multi-asset
- ✗ Custom classifications vs. Strategies

Expanding Universe 3 – Data Quantity and Complexity

Holdings
- ✗ Decades
- ✗ daily

Prices:
- ✗ Many sources and formats
- ✗ different frequencies
- ✗ "round-the-clock" alignment

Reference:
- ✗ Terms& Conditions
- ✗ All asset classes & types
- ✗ UDIs

Auxiliary:
- ✗ Rates/Curves
- ✗ Classifications
- ✗ Indexes and benchmarks

Transform:
- ✗ Currency rates
- ✗ C.A.s
- ✗ Distributions &Accruals

Bench.
- ✗ Decades
- ✗ Daily
- ✗ Multi-family
- ✗ Prices
- ✗ 100,00s

Risk Factors Fuel:
- ✗ Fundamentals & Estimates (Fiscal alignment)
- ✗ Ratings
- ✗ Credit Data

Beyond:
- ✗ News & Events
- ✗ Supply Chain
- ✗ Sector Trends

Numbers (nowadays in the PORT ecosystem)

>1Million securities

Not including UDIs

>12,000 fields

Not including Custom data

~800 TB Daily Data Flow

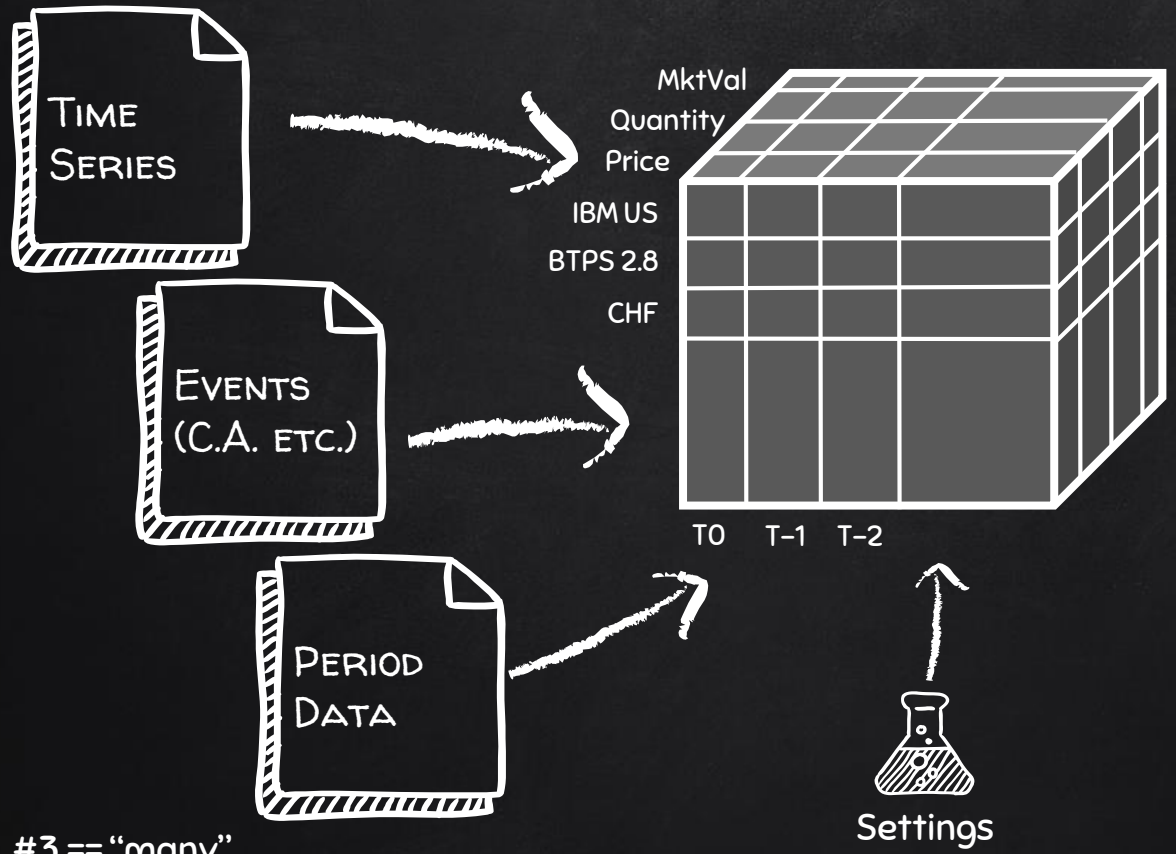
i.e. ~24 PB/month

16 updates/day

~80,000 Avg. Query per Hour

160,000 Peak

Portfolio Analytics Data Requirements
boil down to a (big) cube



Key factors are:

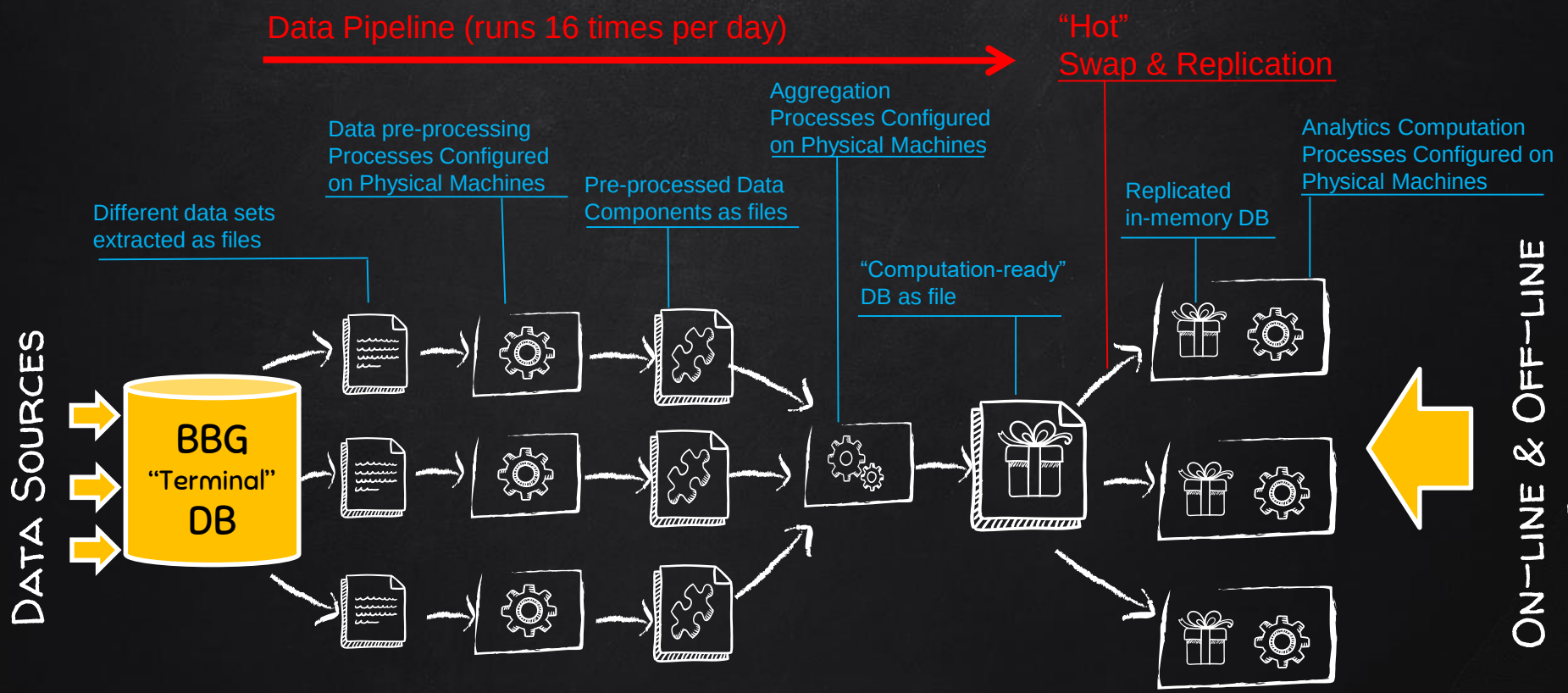
- ✗ Bring the data in the cube
- ✗ Compute in the cube

(easy to say, isn't it?)

#3 == "many"

What We Do Now ('90s on steroids)

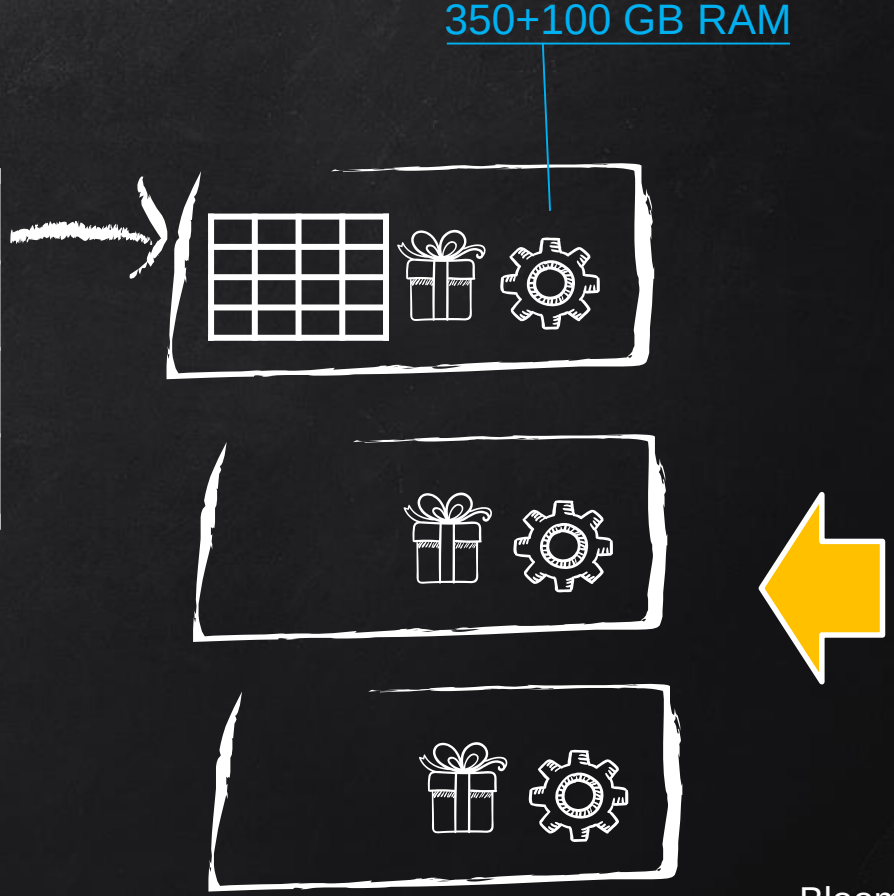
A Traditional large scale architecture: scale-up means re-configuration



#3 == "many"

"Uno Vale Uno" i.e. "Uno alla Volta, per Carita'"

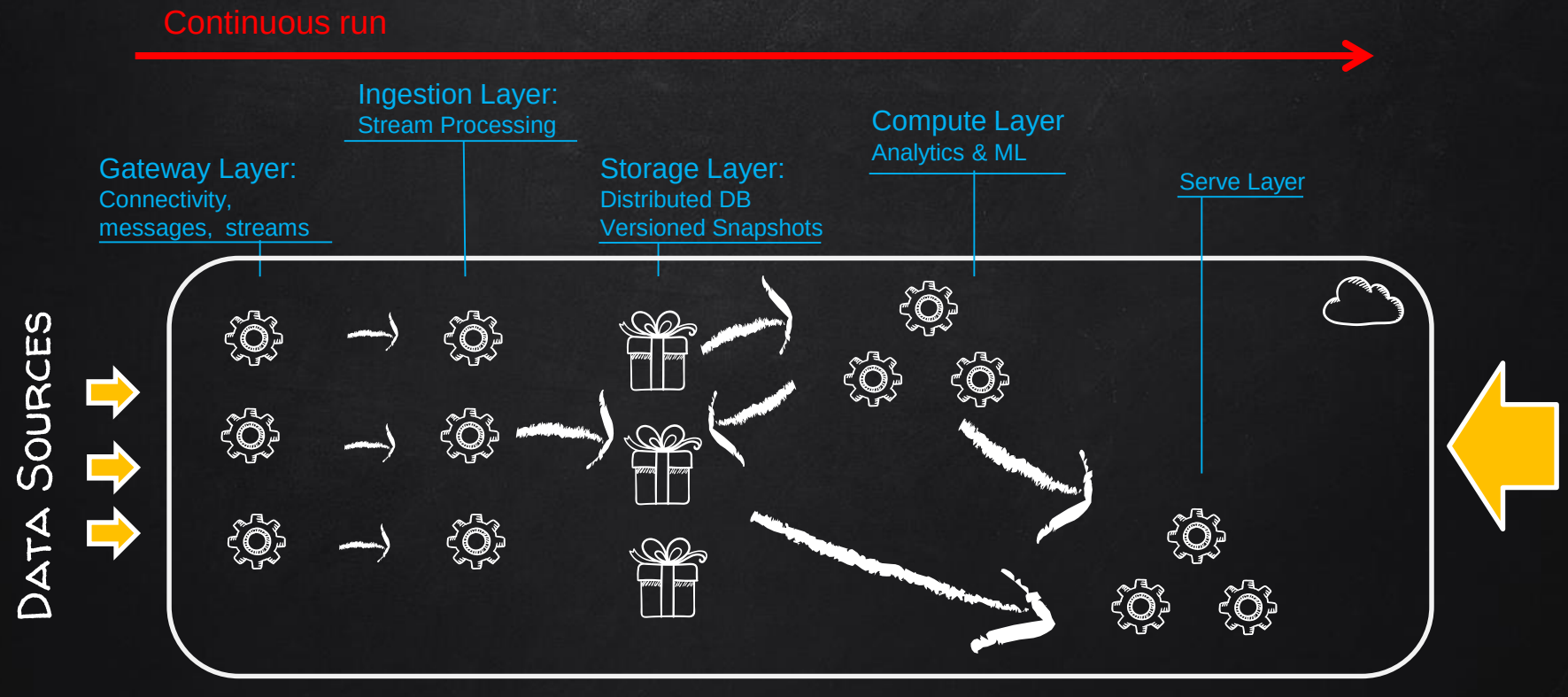
Portfolio	T0	T-1	T-2
IBM US			
BTPS 2.8			
CHF			



#3 == "many"

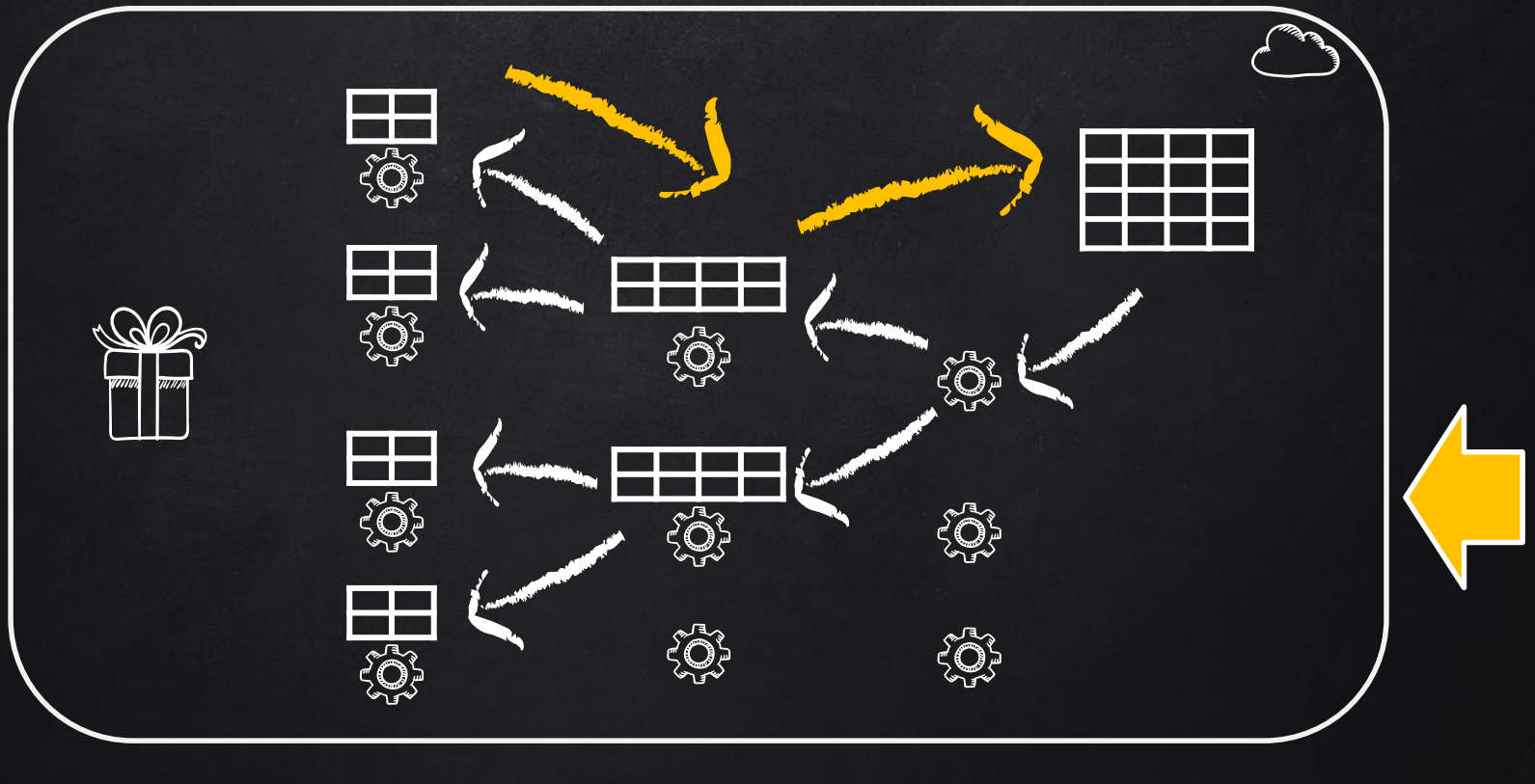
The Target

Virtualized Applications is the key: scale-up is automatic and self-adapting



kafka · Apache Airflow · Spark Streaming · hadoop · Apache Spark · python · Java · Grafana · elastic

Distributed Computing I: Paradigmatic Map-Reduce



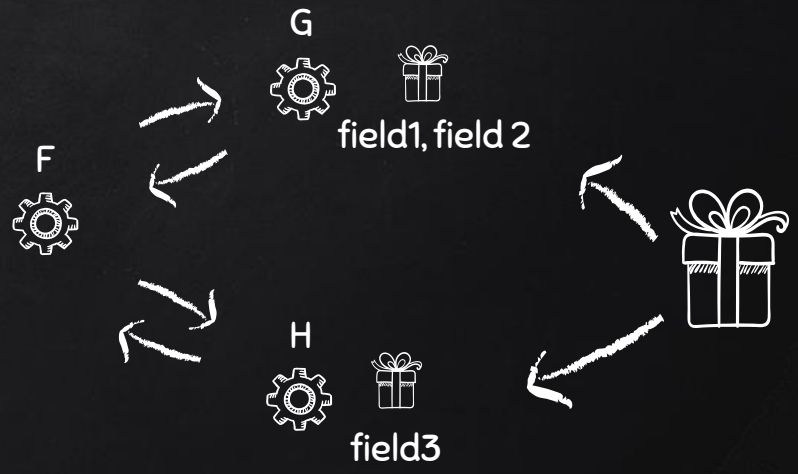
Distributed Computing II: Transparent Distribution

Computation Tree is meta-described:

- ✗ Distribution is transparent (independent branches are executed in parallel)
- ✗ Lazy evaluation brings the right data to the right computation nodes

(e.g. Python code with DASK)

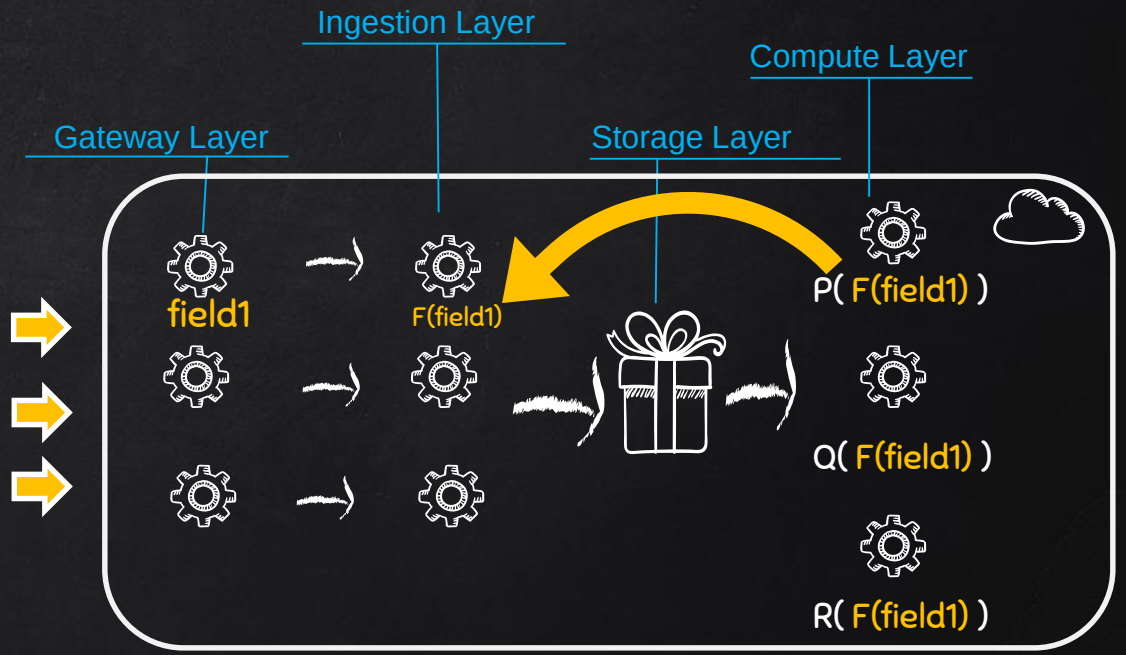
F(G(field1, field 2),H(field3)) ==



Distributed Computing III: Greedy Pipeline

Data-bound computation modules can be easily* pushed backward in the Data Pipeline

* well, yes, they have to be properly packaged



Thanks & Questions

lspampinato1@bloomberg.net

Bloomberg