The Triumph of Simplicity

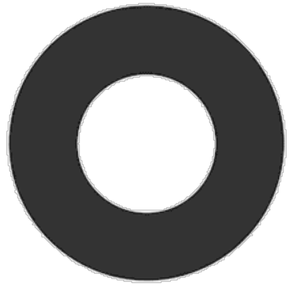
How simple services are reshaping databases and the enterprise

"Life is really simple, but we insist on making it complicated."
-Confucius

Who am I?


HELLO,
MY NAME
IS
IAN PLOSKER

Co-founder & CTO



ORCHESTRATE

@dstroyallmodels

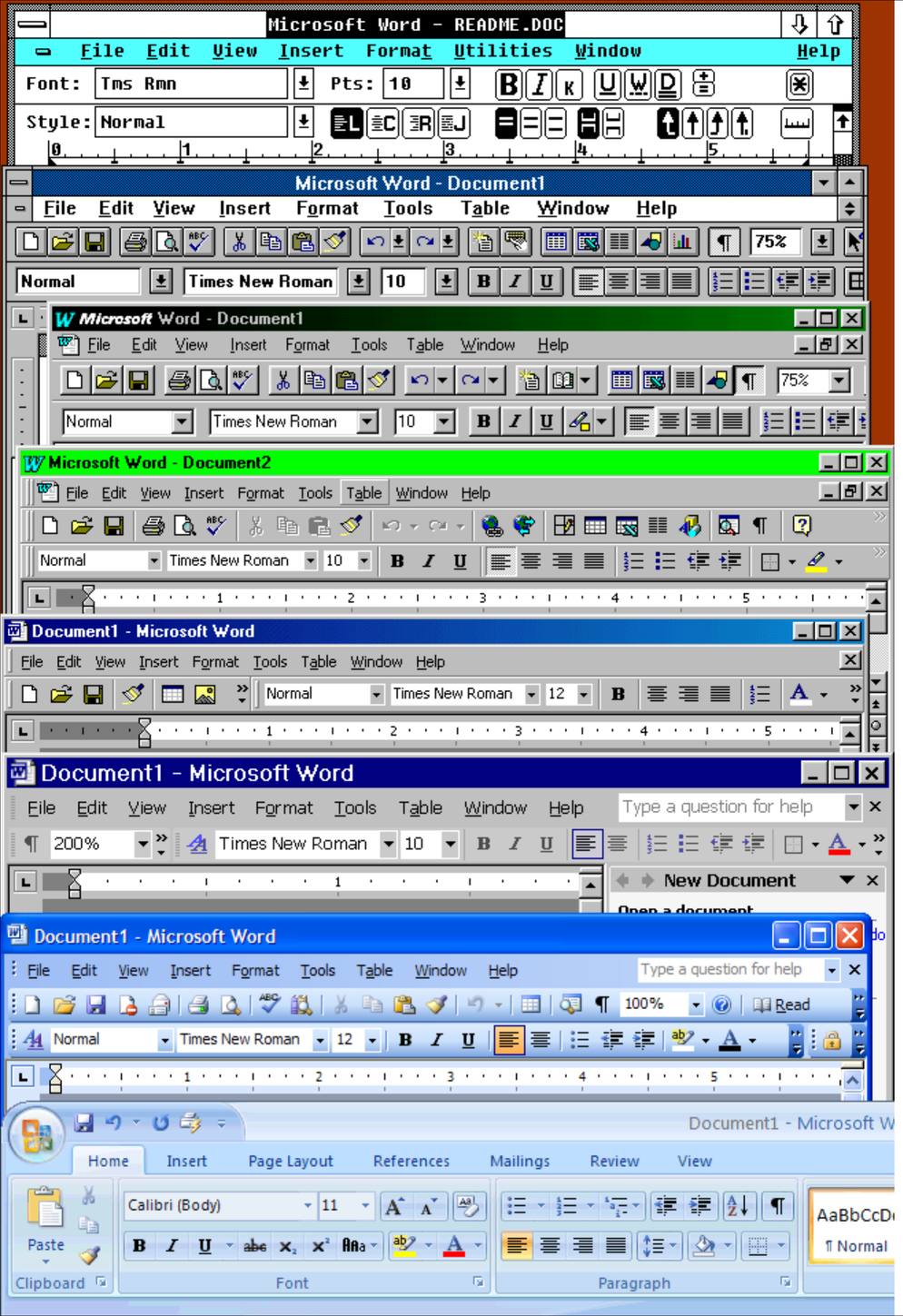
about.me/ian.plosker

Our goal is to make storing and querying so easy, you don't need databases

E.W. Dijkstra

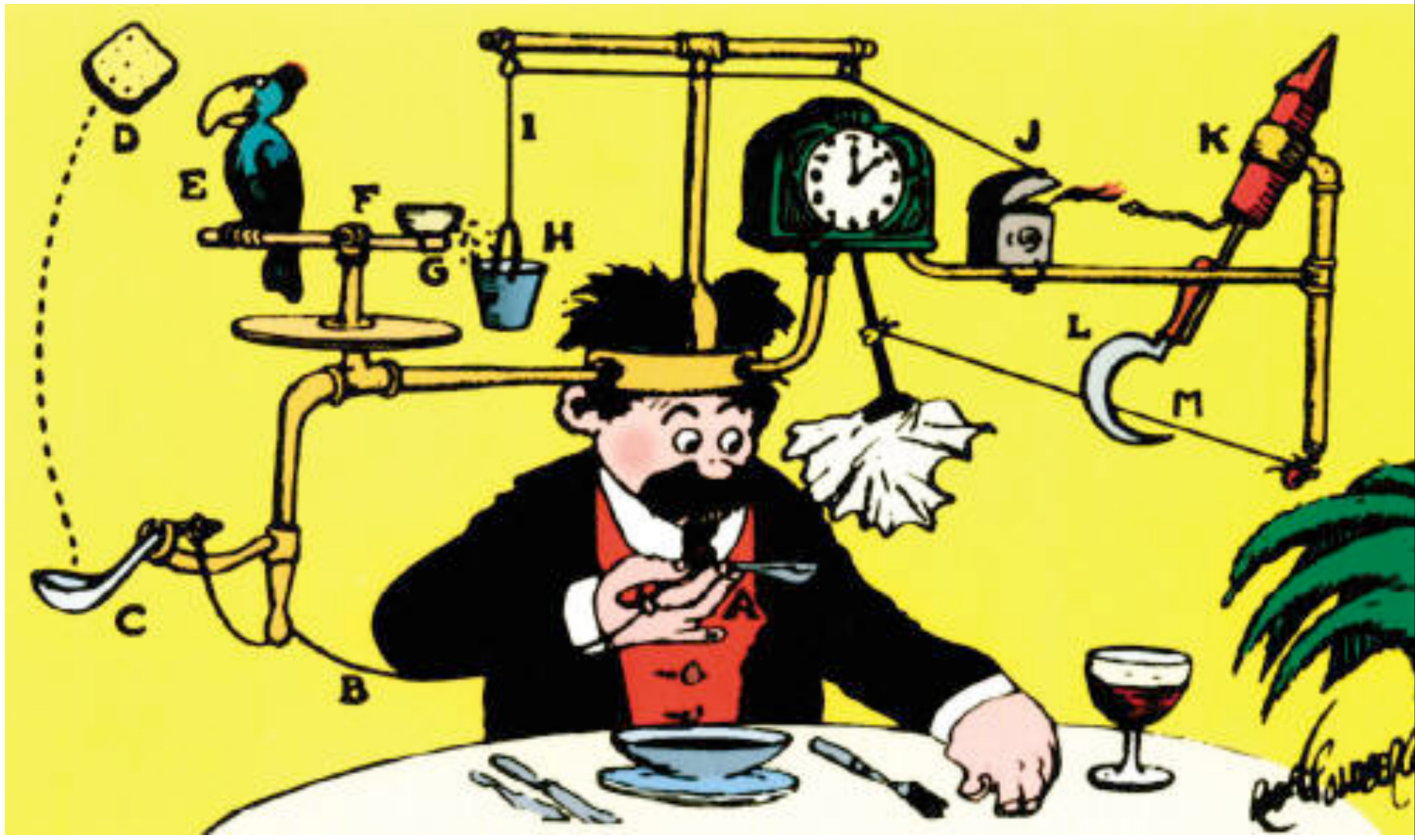
"Simplicity is a great virtue but it requires hard work to achieve it and education to appreciate it. And to make matters worse: complexity sells better."

http://www.cs.utexas.edu/users/EWD/transcriptions/EWD08xx/EWD896.html



Some people believe that complexity is how value is added

We fetishize complexity



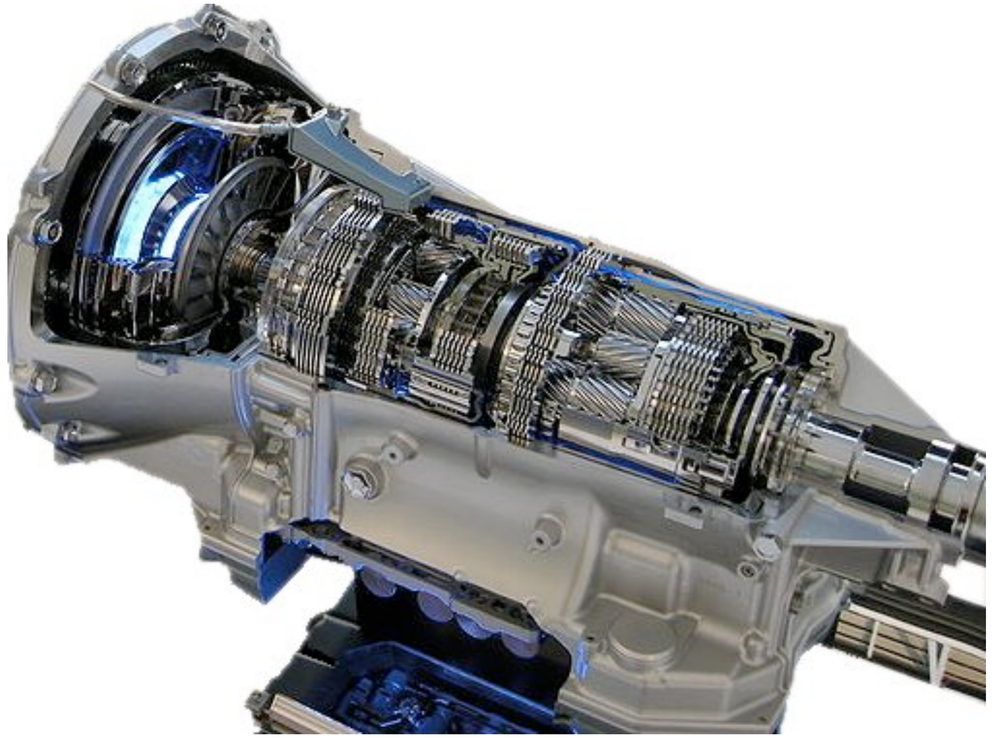
http://www.cs.utexas.edu/users/EWD/transcriptions/EWD08xx/EWD898.html

```
SELECT SUM(offerTotal) as theOfferTotal, SUM(lienTotal) AS theLienTotal, SUM(CLVtotal) AS theCLVtotal,
SUM(estGrossProfitTotal) AS theESTGPtotal FROM (( SELECT COALESCE(SUM(COALESCE(offerAmount, 0)), 0) AS
offerTotal, COALESCE(SUM(COALESCE(amount, 0) + COALESCE(legalFees, 0) + COALESCE(costs, 0)), 0) AS
lienTotal, COALESCE(SUM(((amount + legalFees + costs) * (1 + (rateOfInterest / 100) *
(FLOOR((UNIX_TIMESTAMP(NOW()) - UNIX_TIMESTAMP(dateOfAttachment)) / 86400) / 365)))), 0) AS CLVtotal,
COALESCE(SUM((((amount + legalFees + costs) * (1 + (rateOfInterest / 100) * (FLOOR((UNIX_TIMESTAMP(NOW()) -
UNIX_TIMESTAMP(dateOfAttachment)) / 86400) / 365))) - COALESCE(offerAmount, 0))), 0) AS estGrossProfitTotal
FROM lienTable AS theLienTable, propertyTable, property_lien, stateInterestTable, data, judgementLienTable
WHERE theLienTable.lienID = property_lien.lienID AND propertyTable.propertyID = property_lien.propertyID AND
propertyTable.state = stateInterestTable.state AND theLienTable.lienID = judgementLienTable.lienID AND
theLienTable.lienStatusID IN (65, 70, 75) AND data.id = (SELECT data.id FROM lienTable, data, data_lien
WHERE lienTable.lienID = data_lien.lienID AND data_lien.id = data.id AND category = 15 AND lienTable.lienID
= theLienTable.lienID ORDER BY data.id DESC LIMIT 1) AND dateOfAttachment != 0 AND UNIX_TIMESTAMP(NOW()) >
UNIX_TIMESTAMP(dateOfAttachment) AND FLOOR((UNIX_TIMESTAMP(NOW()) - UNIX_TIMESTAMP(dateOfAttachment)) /
86400) > 0 AND rateOfInterest > 0 ) UNION ( SELECT COALESCE(SUM(COALESCE(offerAmount, 0)), 0) AS offerTotal,
COALESCE(SUM(COALESCE(amount, 0) + COALESCE(legalFees, 0) + COALESCE(costs, 0)), 0) AS lienTotal,
COALESCE(SUM(((amount + legalFees + costs) * (1 + (rateOfInterest / 100) * (FLOOR((UNIX_TIMESTAMP(NOW()) -
UNIX_TIMESTAMP(judgementDate)) / 86400) / 365)))), 0) AS CLVtotal, COALESCE(SUM((((amount + legalFees +
costs) * (1 + (rateOfInterest / 100) * (FLOOR((UNIX_TIMESTAMP(NOW()) - UNIX_TIMESTAMP(dateOfAttachment)) /
86400) / 365))) - COALESCE(offerAmount, 0))), 0) AS estGrossProfitTotal FROM lienTable AS theLienTable,
propertyTable, property_lien, stateInterestTable, data, judgementLienTable WHERE theLienTable.lienID =
property_lien.lienID AND propertyTable.propertyID = property_lien.propertyID AND propertyTable.state =
stateInterestTable.state AND theLienTable.lienID = judgementLienTable.lienID AND theLienTable.lienStatusID
IN (65, 70, 75) AND data.id = (SELECT data.id FROM lienTable, data, data_lien WHERE lienTable.lienID =
data_lien.lienID AND data_lien.id = data.id AND category = 15 AND lienTable.lienID = theLienTable.lienID
ORDER BY data.id DESC LIMIT 1) AND COALESCE(dateOfAttachment, 0) = 0 AND judgementDate != 0 AND
UNIX_TIMESTAMP(NOW()) > UNIX_TIMESTAMP(judgementDate) AND FLOOR((UNIX_TIMESTAMP(NOW()) -
UNIX_TIMESTAMP(judgementDate)) / 86400) > 0 AND rateOfInterest > 0 ) ) AS theBigTable;
```

Who Owns Complexity?

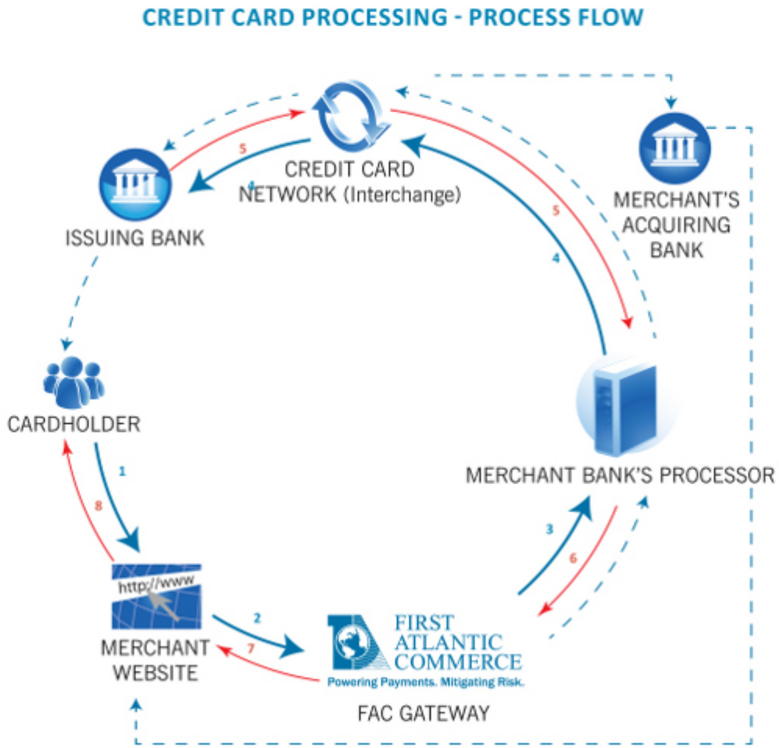
Simple

Complex



Simple

Complex

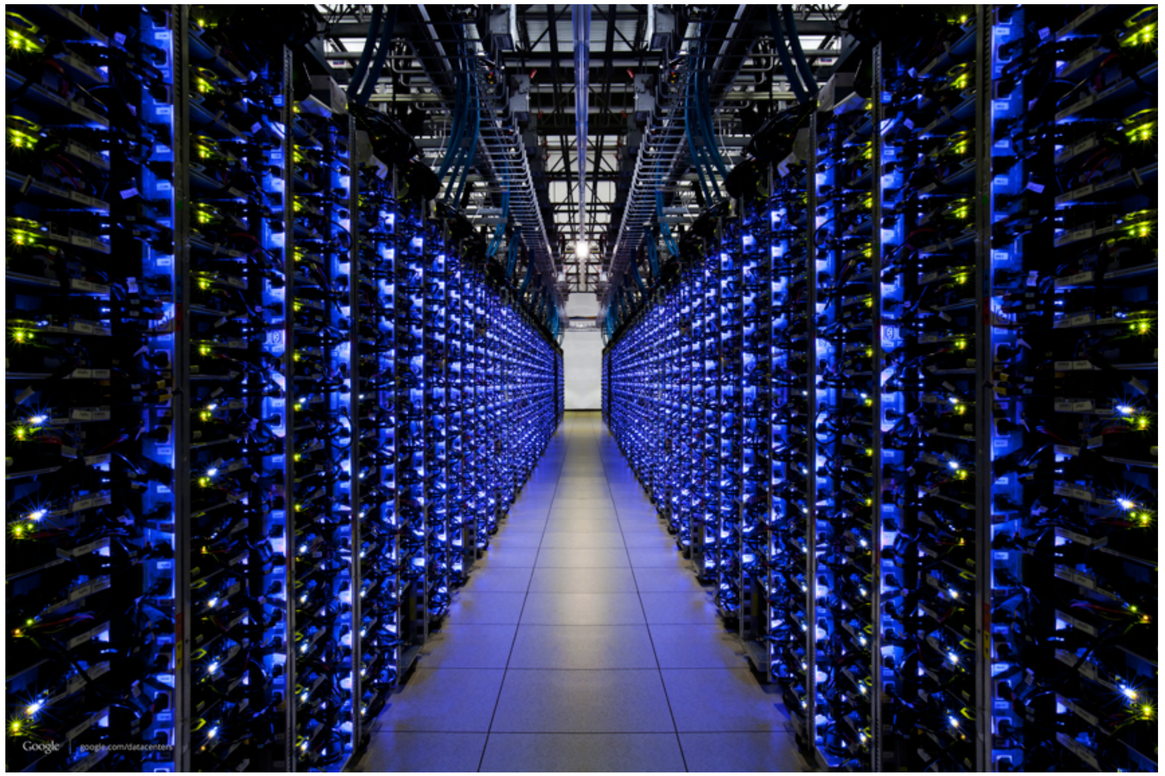


Simple

Google

Google Search I'm Feeling Lucky

Complex



Simple

```
aws ec2 run-instances --image-id ami-
c3b8d6aa --count 2 --instance-type
t1.micro --key-name MyKeyPair --
security-groups MySecurityGroup
```

Complex

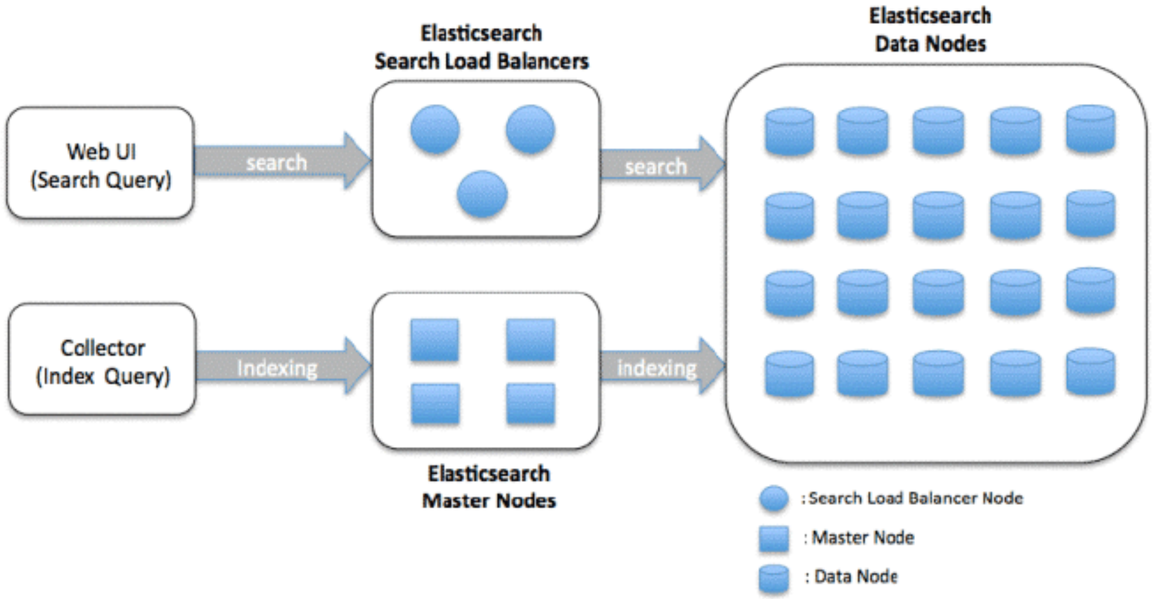
Simple

Complex

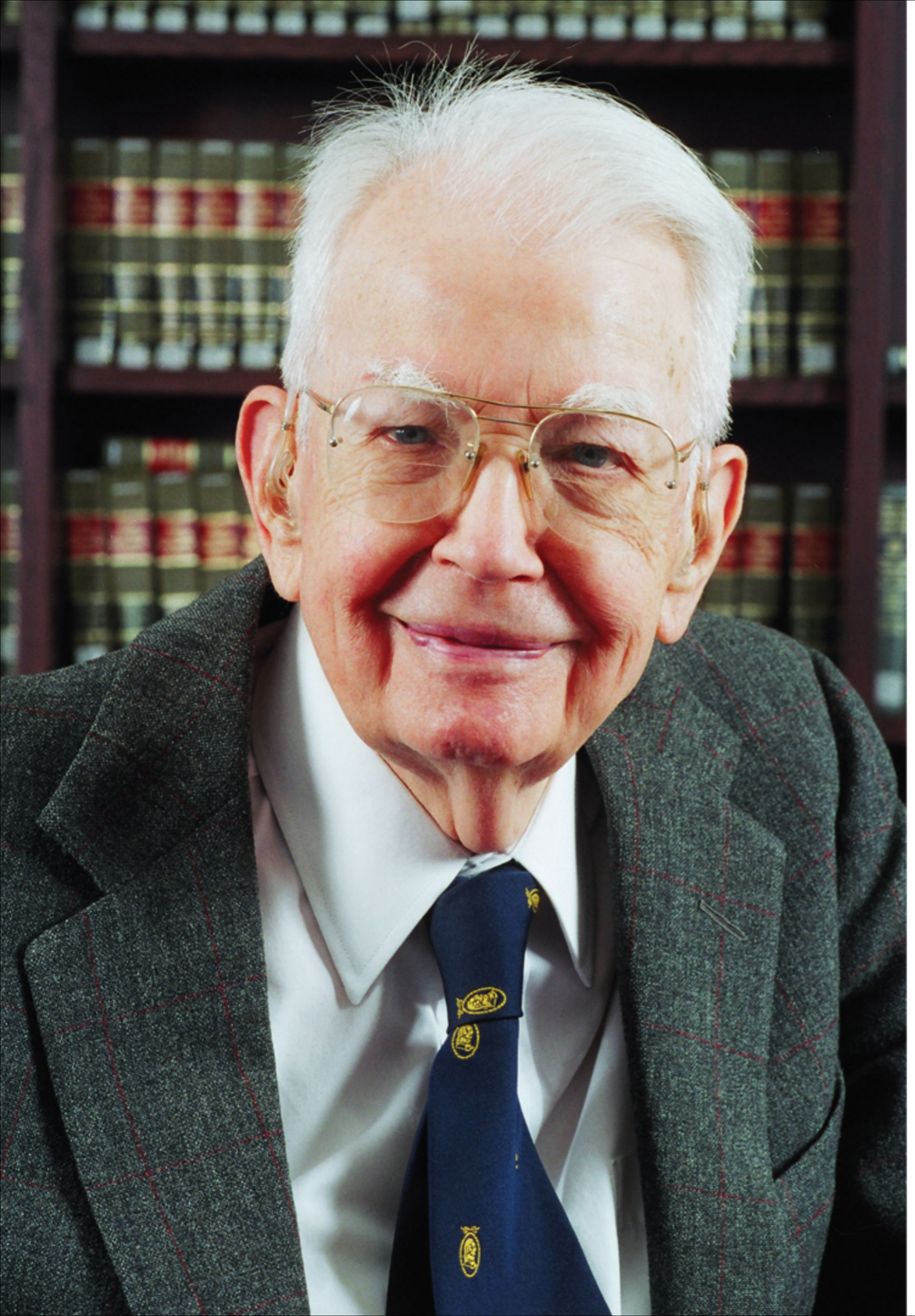
Simple

```
author:Dijkstra AND year:[1970 TO 1985]
```

Complex



There is a quiet revolution in progress



Ronald Coase

"... a firm will tend to expand until the costs of organizing an extra transaction within the firm become equal to the costs of carrying out the same transaction by means of an exchange on the open market or the costs of organizing in another firm."

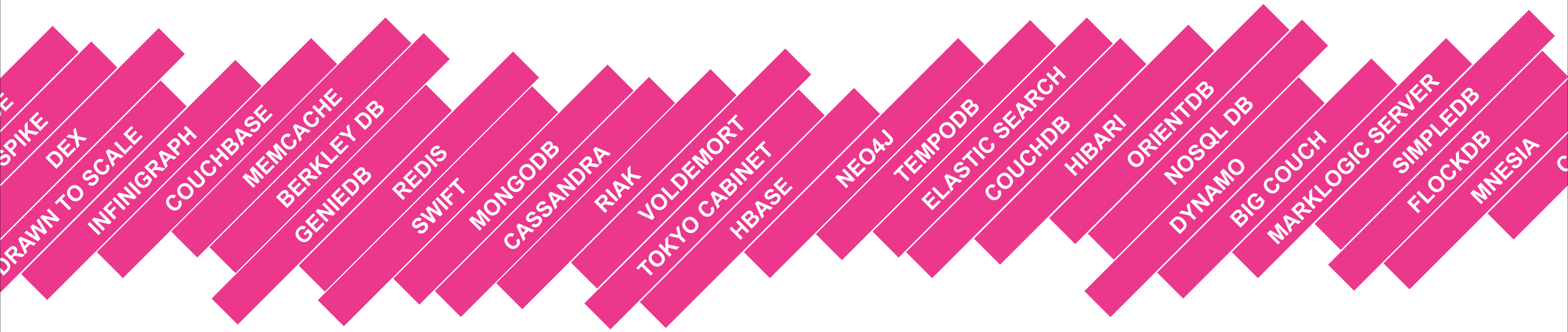
http://www3.nccu.edu.tw/~jsfeng/CPEC11.pdf

Functions that organizations formerly did themselves are now sourced externally

Let’s talk databases

Databases in 2005

Databases in 2013



25 databases in production today that didn't exist 7 years ago

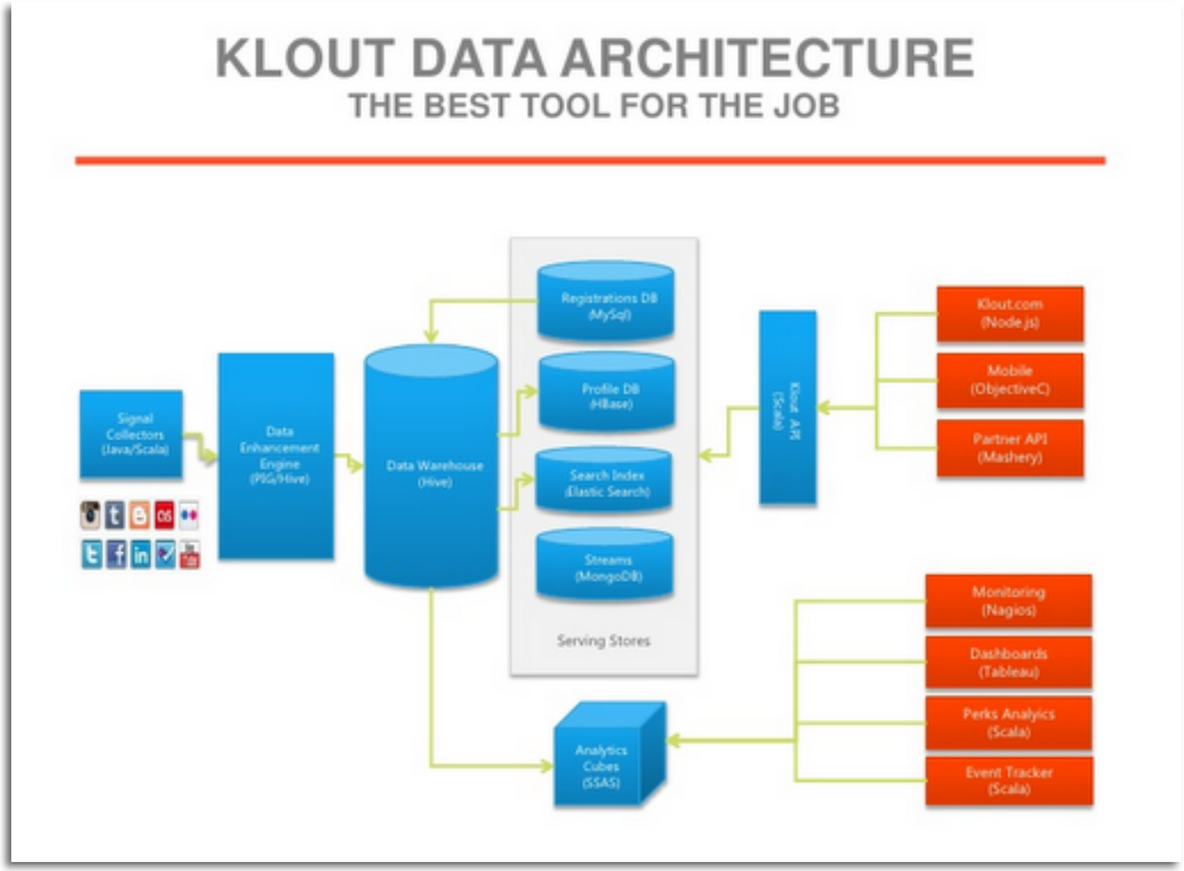
ONLINE QUERY TYPES

	Key-Value	**Search**	**Geo**	**Graph/ Relation**	**Event**
Scale-up	BerkleyDB CouchDB MongoDB MySQL	SOLR Sphinx	PostGIS MongoDB SOLR	neo4j	MySQL
Scale-out	Riak Cassandra	elasticsearch	elasticsearch	titan	HBase

The database paradox of choice:

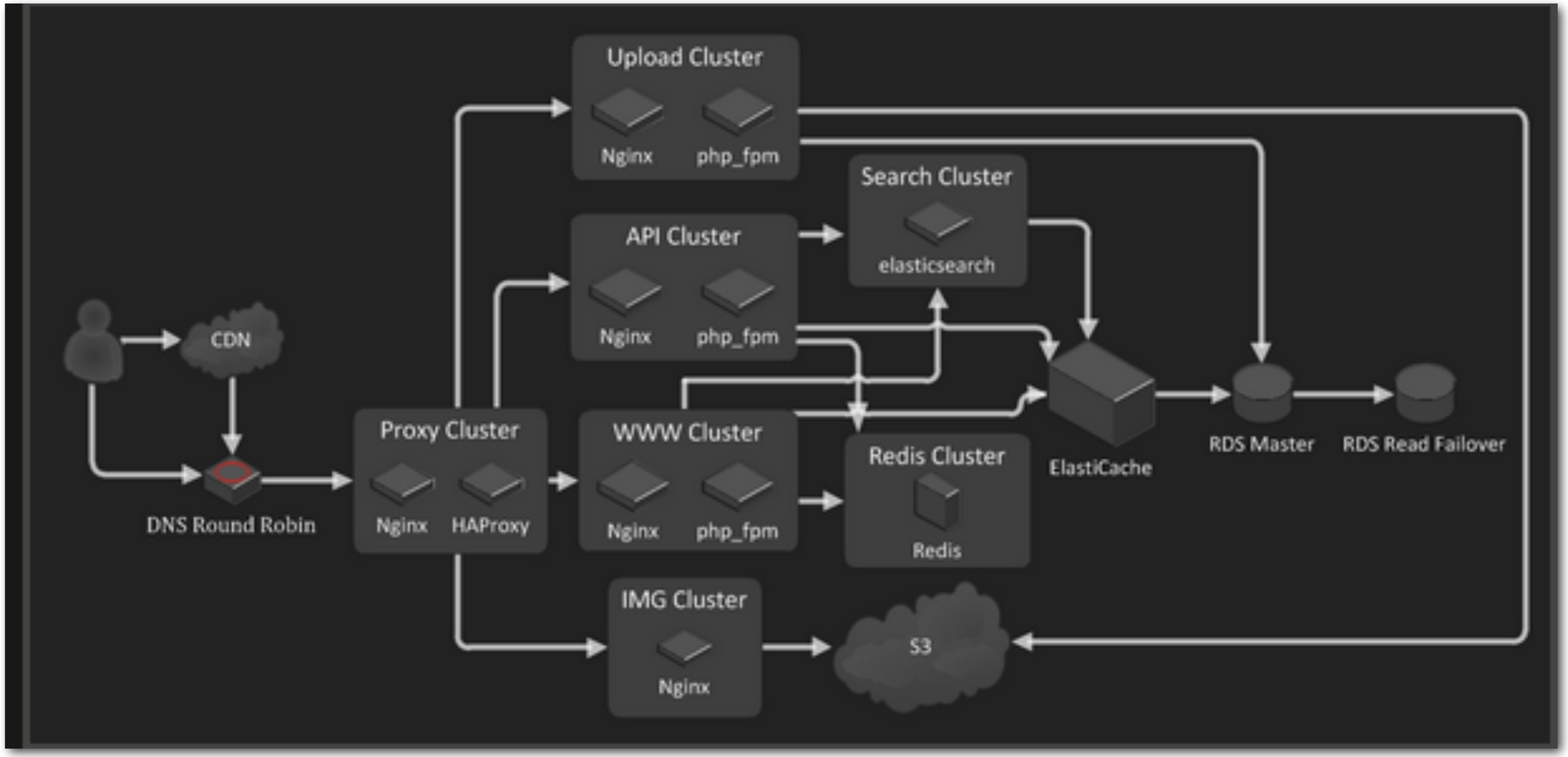
Choice has brought complexity

Klout's Data Architecture



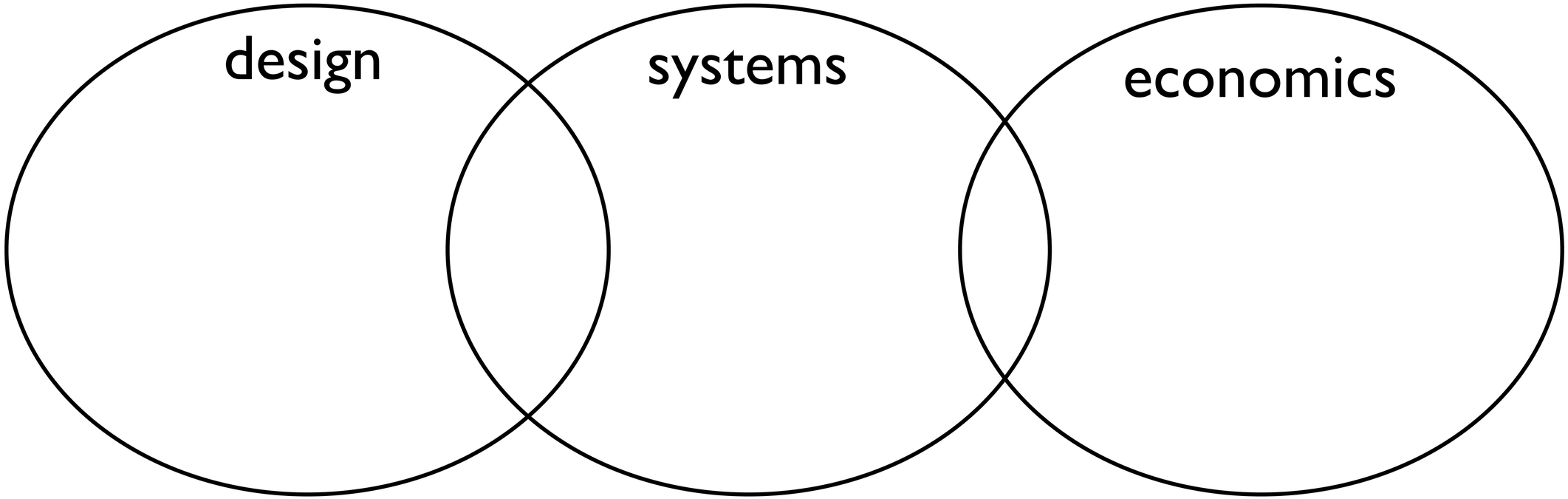
1. Hbase
2. MySQL
3. ElasticSearch
4. MongoDB

Imgur's Data Architecture

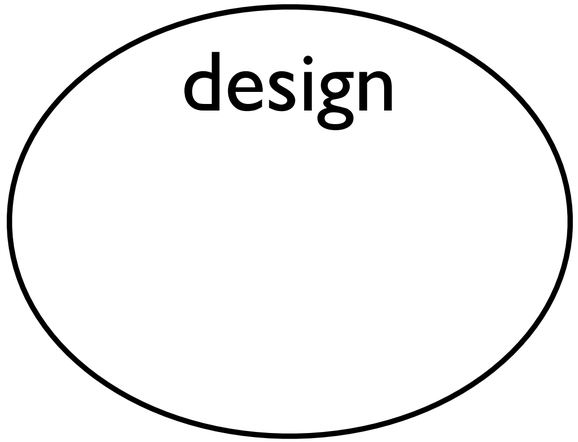


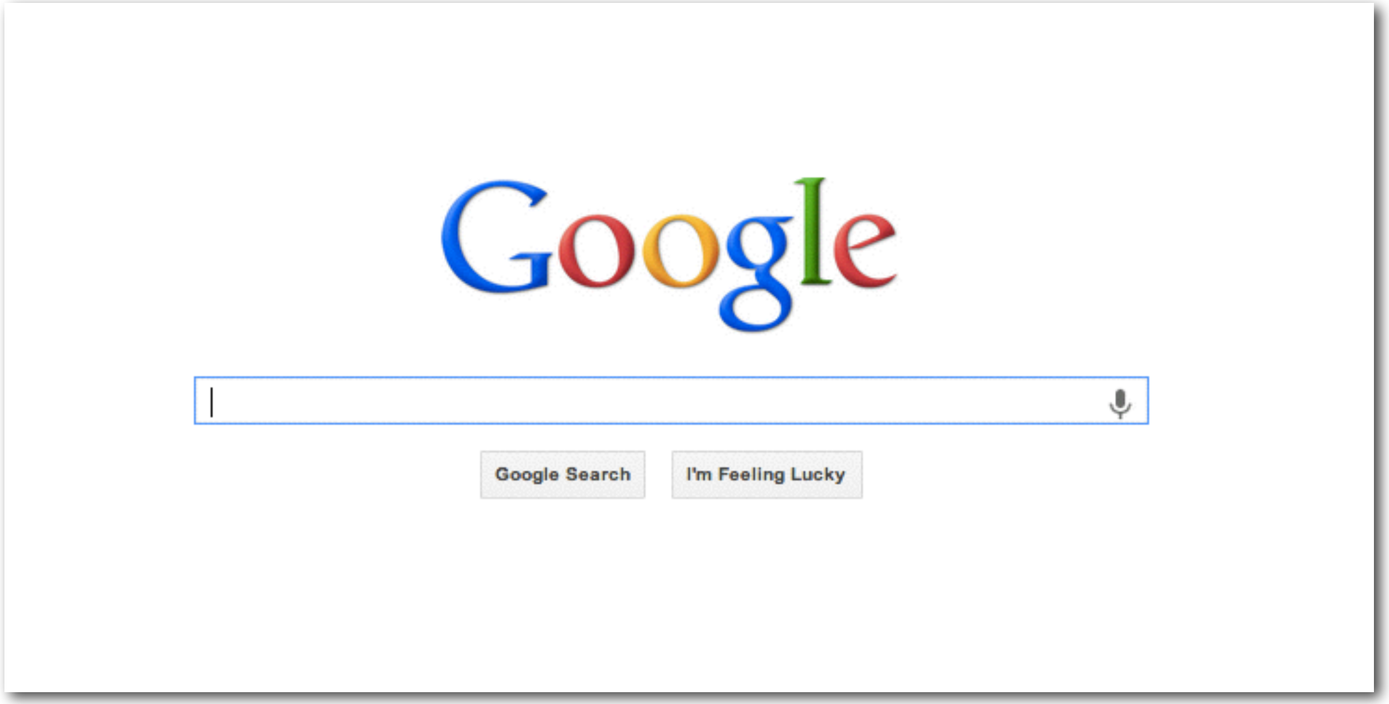
1. Hbase 2. ElasticSearch 3. MySQL 4. Memcache 5. Redis 6. HAproxy

The Value of Simplicity Varies Across Domains

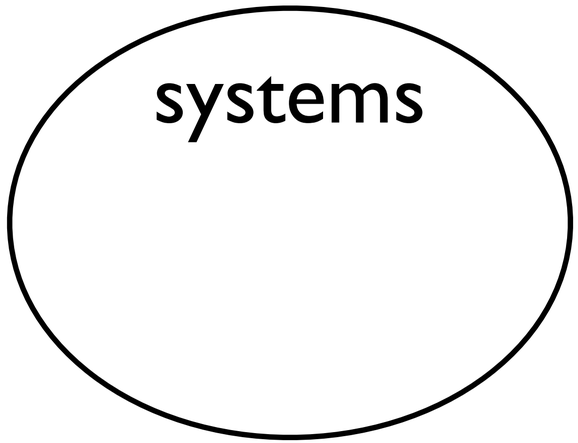


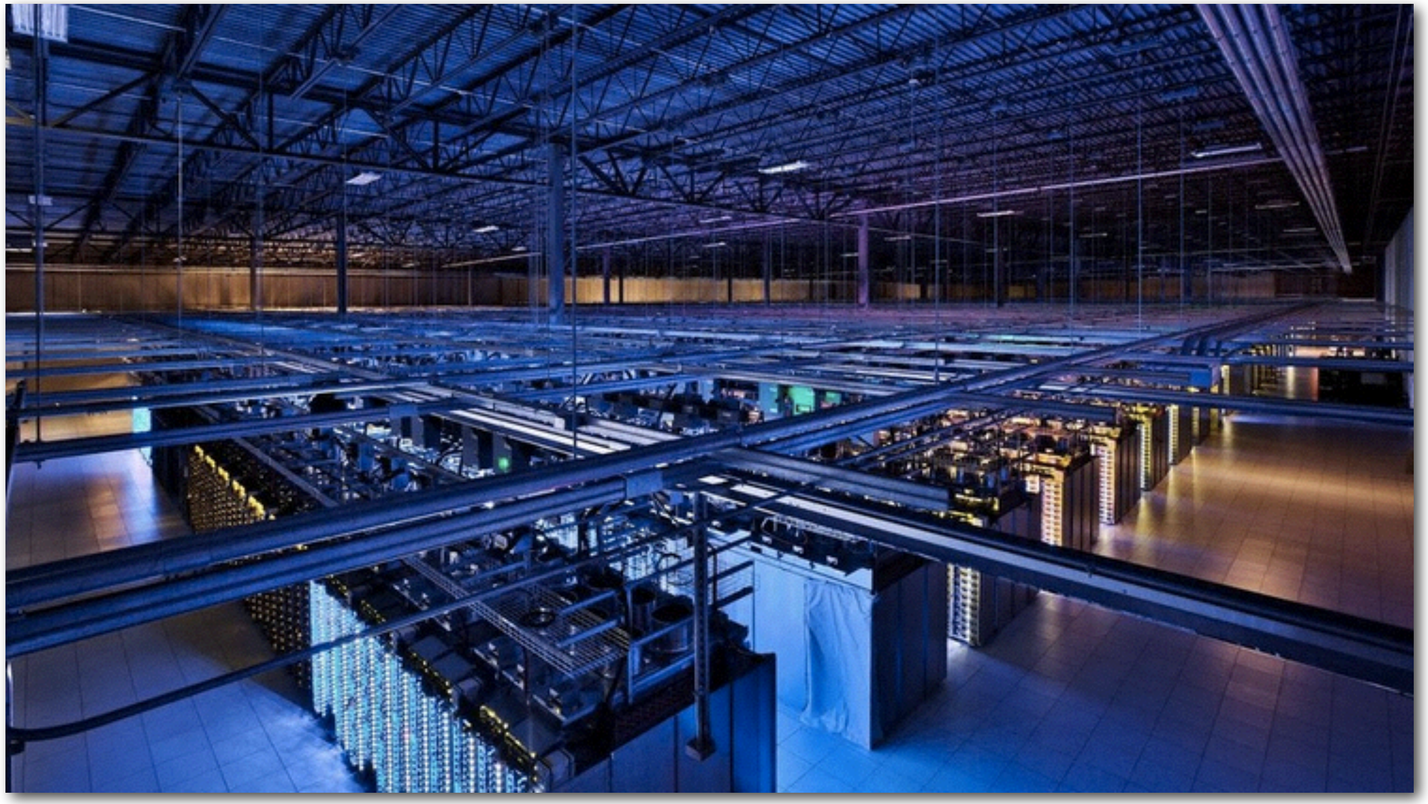
Simple or complex?





Simple or complex?



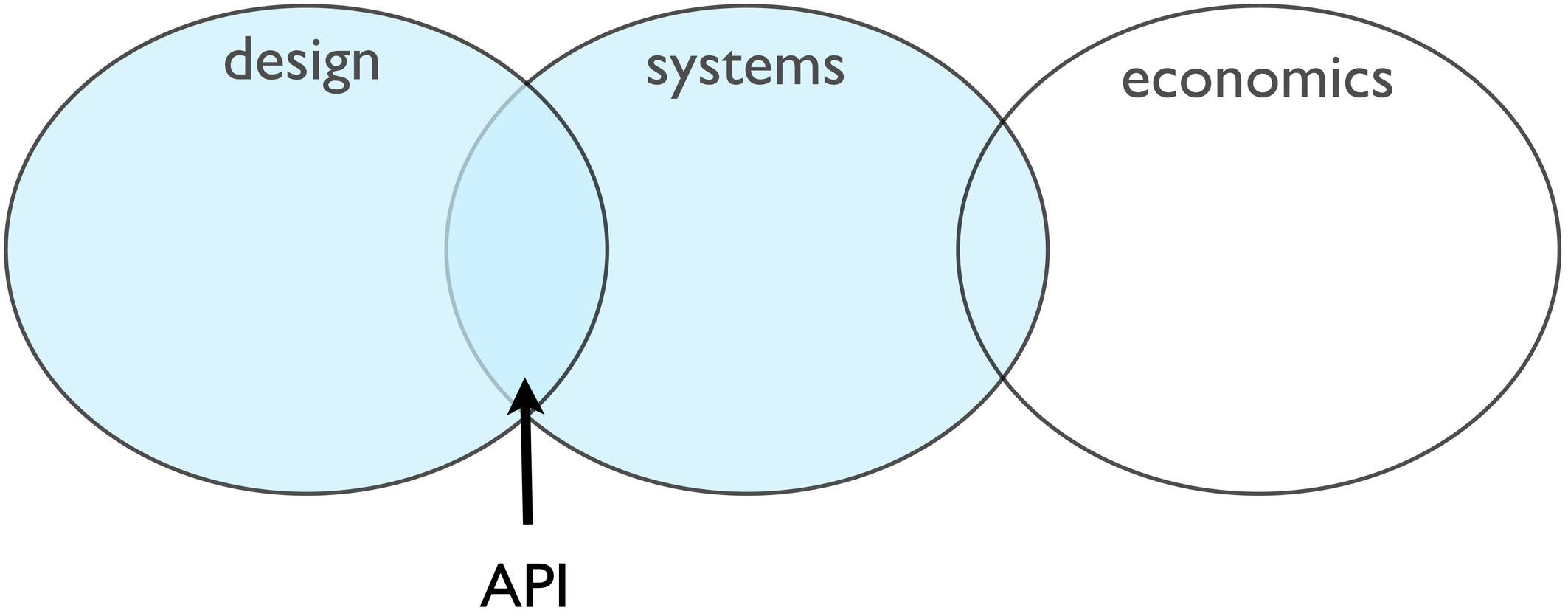


Simple or complex?

economics

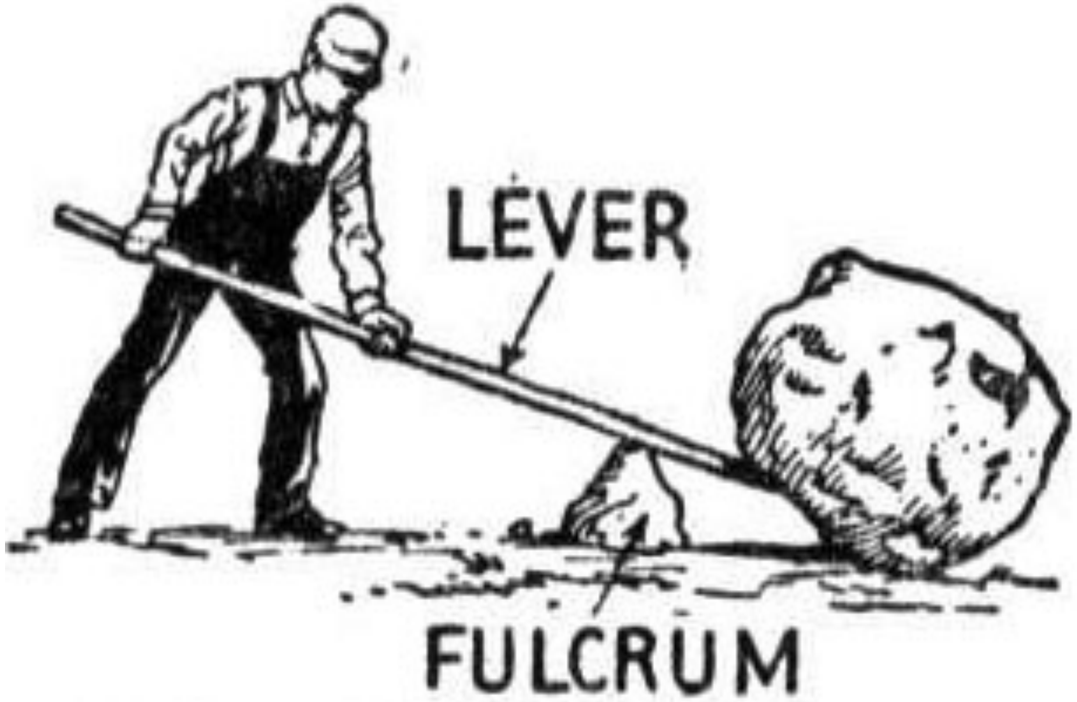
Net Digital Ad Revenue Share worldwide						
	2011		2012		2013	
	Share (percent)	Revenue ($ bn)	Share (percent)	Revenue ($ bn)	Share (percent)	Revenue ($ bn)
Google	32.08	27.72	31.46	32.73	32.84	38.62
Facebook	3.65	3.15	4.11	4.28	5.41	6.36
Yahoo!	3.95	3.41	3.37	3.51	2.97	3.50
Microsoft	2.60	2.25	2.46	2.56	2.49	2.92
IAC	1.01	0.87	1.26	1.32	1.37	1.62
AOL	1.17	1.01	1.02	1.06	0.94	1.11
Amazon	0.48	0.42	0.59	0.61	0.71	0.84
Twitter	0.16	0.24	0.28	0.38	0.50	0.58
Pandora	0.28	0.14	0.36	0.29	0.49	0.58
LinkedIn	0.18	0.16	0.25	0.26	0.32	0.38
Millennial Media	0.05	0.04	0.07	0.07	0.10	0.12
Other	54.40	47.02	54.77	56.98	51.85	60.97
Total (USD in billions)	**$86.43**		**$104.04**		**$117.60**	

Source: eMarketer



expose utility, hide complexity

Introducing: Leverage



Man lifting a stone
with a lever

Simple

Complex

Fulcrum (API)

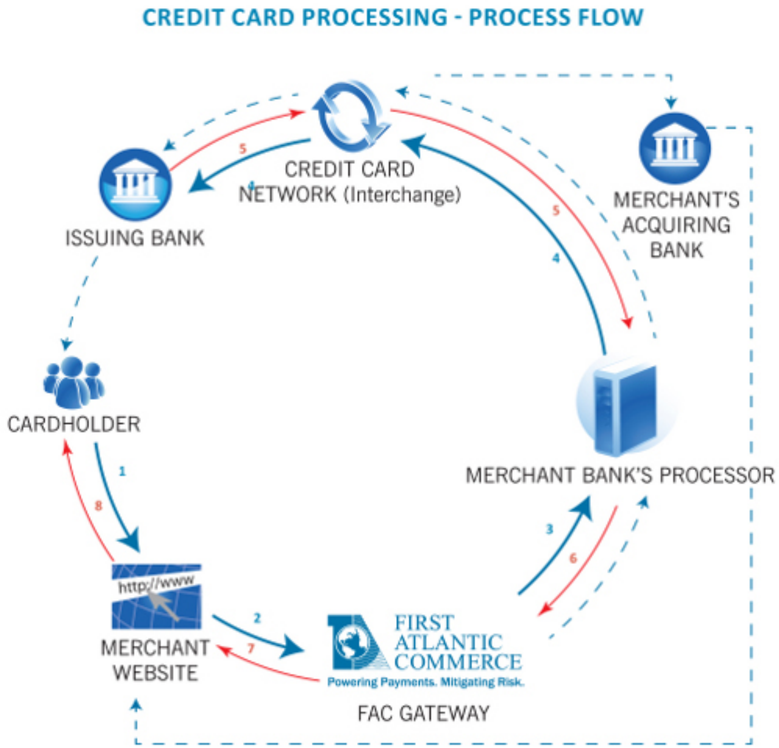
Simple

PayPal™

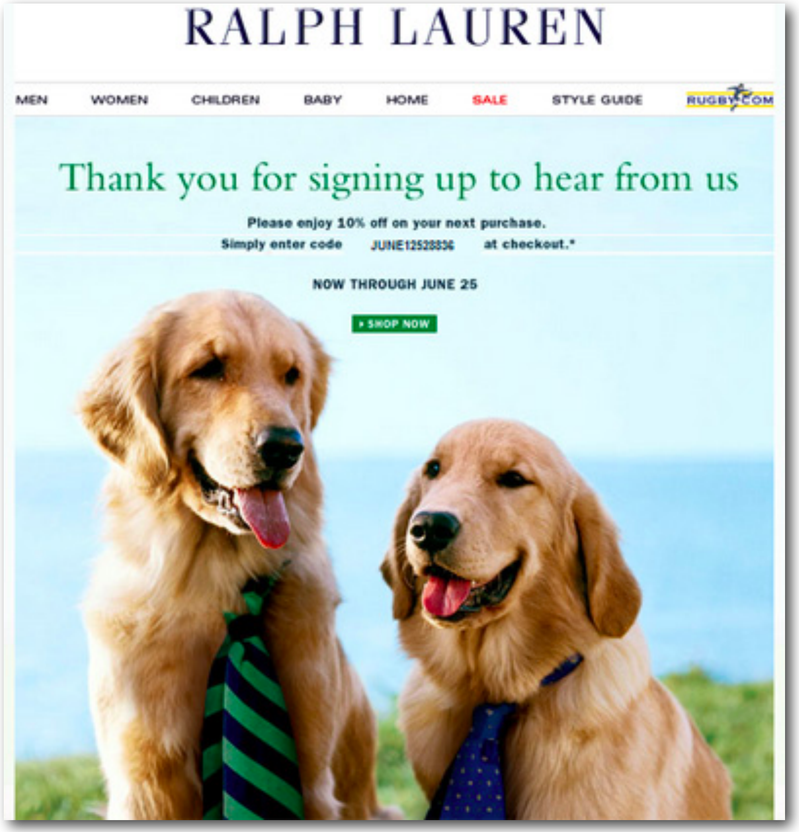
stripe

Square

Complex



Simple



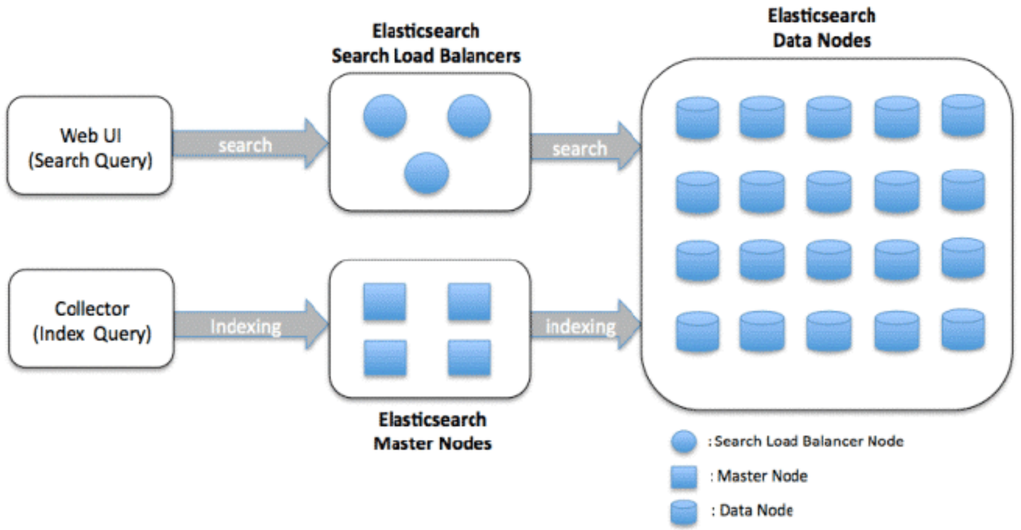
SendGrid
Email Delivery. Simplified.

Complex

Simple

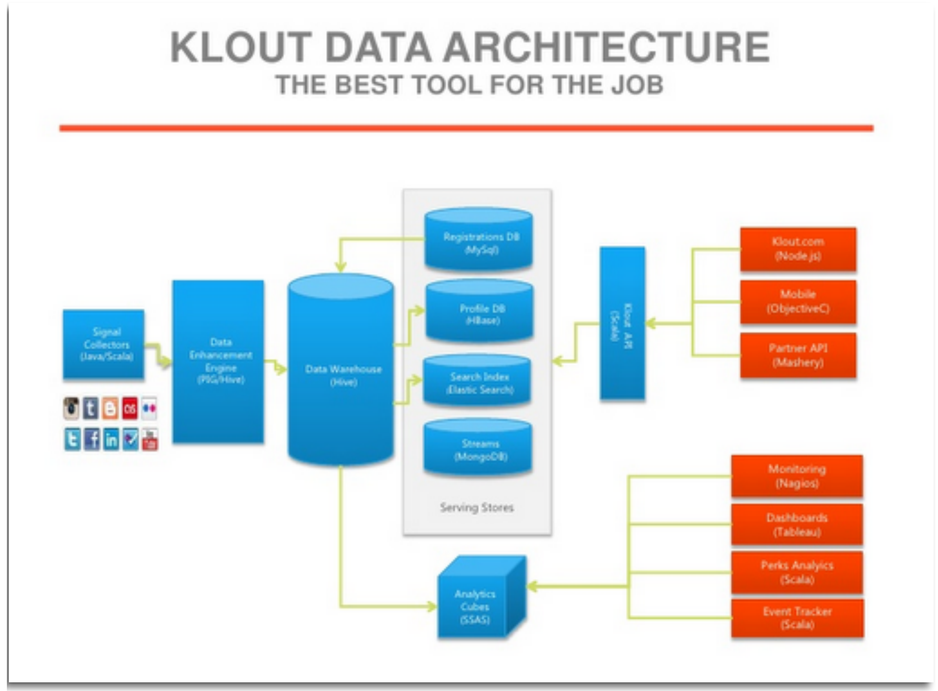
```
author:Dijkstra AND year:[1970 TO 1985]
```

Complex



Simple

Complex



Orchestrate.io is a HTTP REST service that unifies 5 common query types

Key/Value

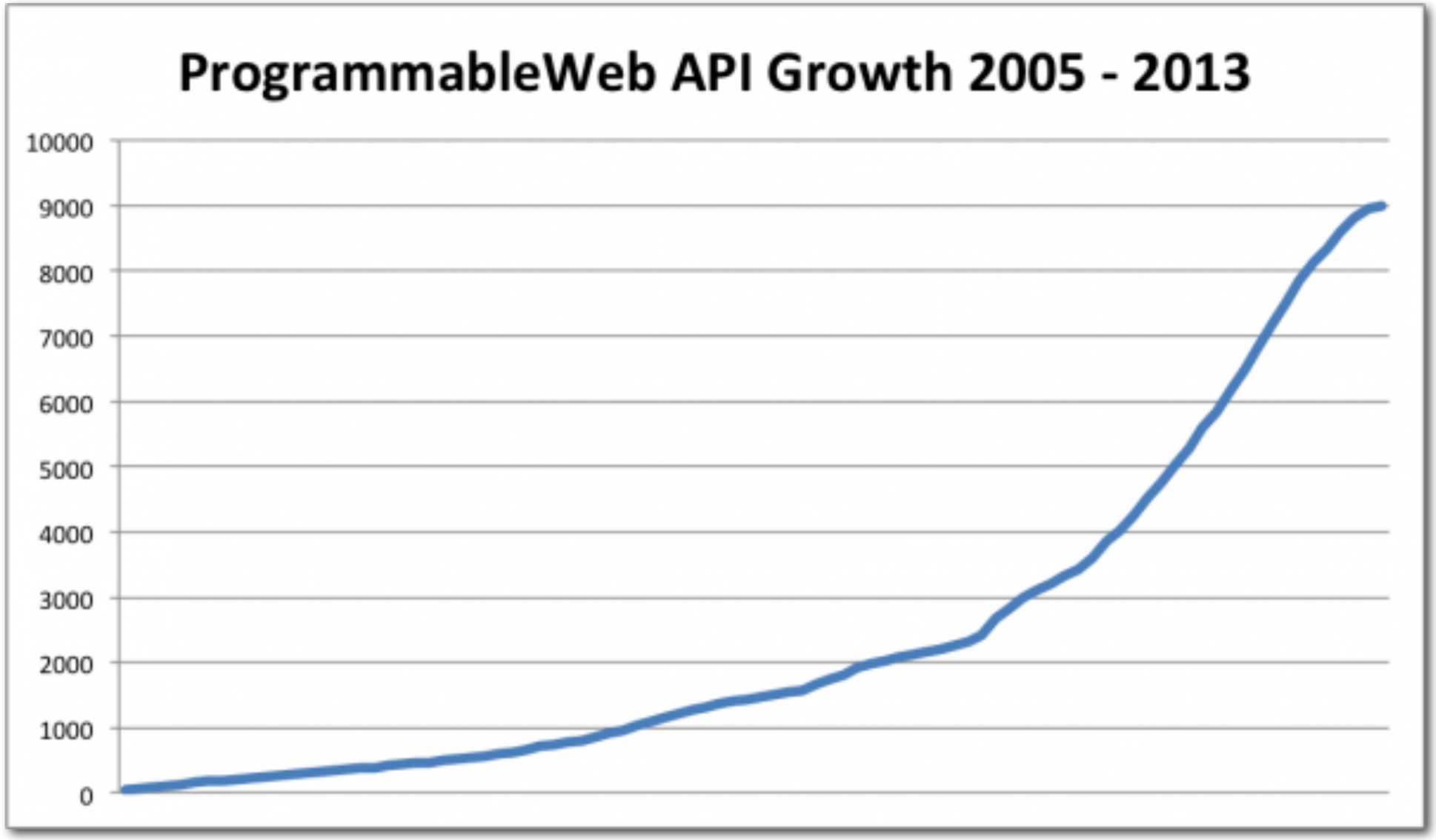
Full-text Search

Graph

Time-ordered events

Geolocation

API Growth


ProgrammableWeb API Growth 2005 - 2013
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0

Where does your business provide value?

Simple

Complex

Ease of use

Leverage

Economies of Scale

Simplicity

h/t @AntonyFalco