



Essential insight for leaders
of the digital economy.

DevOps, from a storage perspective

@cote | 3 Dec 2014



451 Research is an information technology research & advisory company

Research & Data

Advisory Services

Events

Founded in 2000

210+ employees, including over 100 analysts

1,000+ clients: Technology & Service providers, corporate advisory, finance, professional services, and IT decision makers

10,000+ senior IT professionals in our research community

Over 52 million data points each quarter

4,500+ reports published each year covering 2,000+ innovative technology & service providers

Headquartered in New York City with offices in London, Boston, San Francisco, and Washington D.C.

451 Research and its sister company Uptime Institute comprise the two divisions of The 451 Group



Michael Coté

- Research Director of Infrastructure Software, 451 Research
- Focusing on developers and cloud
- I worked in development at startups and large ISVs, and then as an analyst, and in corporate strategy & M&A.

cote@451research.com

[@cote](#)

<http://cote.io>

Agenda

1. Why DevOps and why now?
2. The software delivery pipeline is the mission critical asset
3. Technologies are forming now with few commercial offerings; process, culture, and people are the current “technology”
4. Microservices are emerging as a key DevOps pattern along with other technologies, driving new storage needs for DevOps



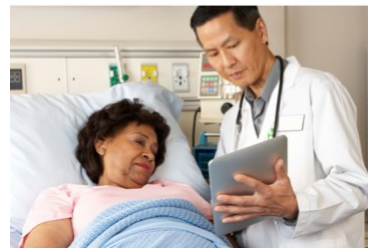
DevOps: why & what?

Think of the baseline as “no software,” and then you’ll understand how much software has to eat

Legacy...



New...

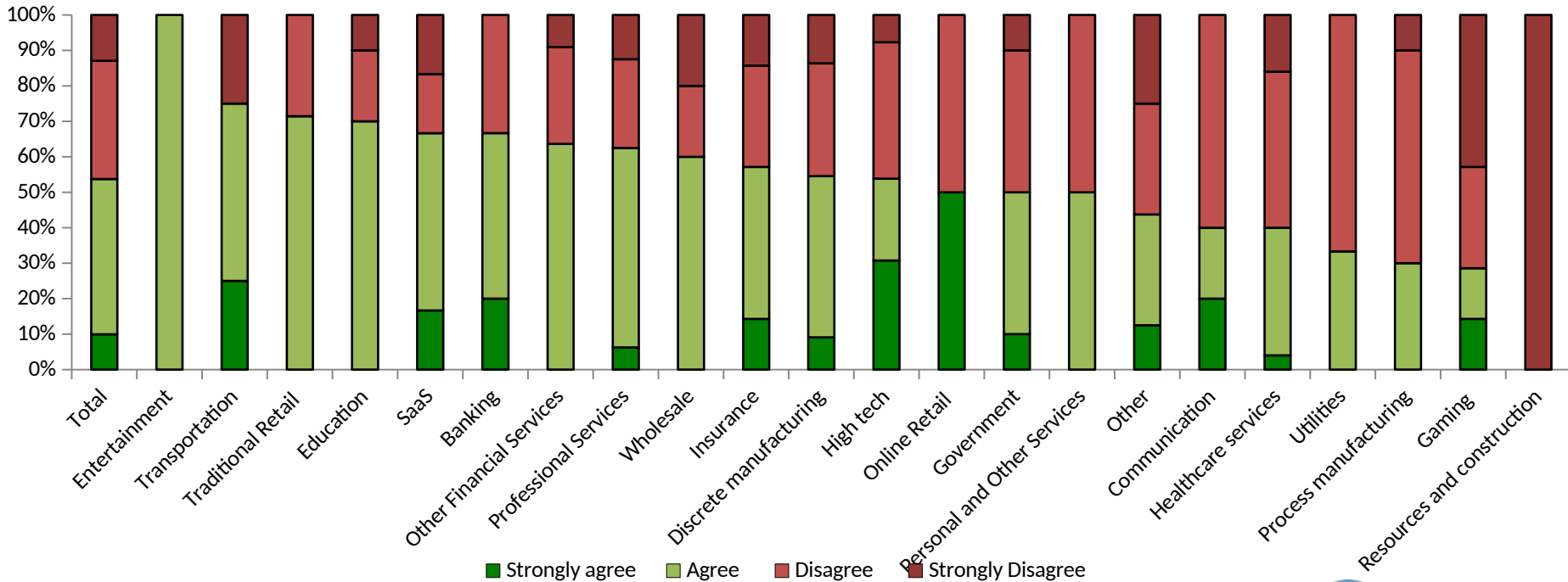


Not seen here: white-collar tool-chains

Sources: [451 HCTS NA 2013 conference](#); [Chris Dancy](#).

A focus on shorter release cycles is driving the need for cloud & DevOps

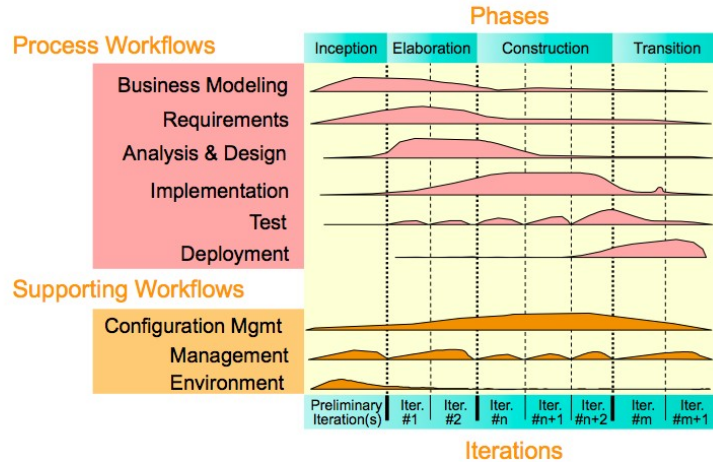
My company would benefit from deploying software more frequently than we currently do (by industry)



Notes: "Total" %'s are 10% strongly agree, 44% agree, 33% disagree, 13% strongly disagree; we are suspicious of "on-demand"; n=201 so beware of very small sample sizes
Source: 2014 Q1 DevOps custom study, 451 Research

Comparing straw men, “waterfall”: getting what you asked for is more important than getting the best solution

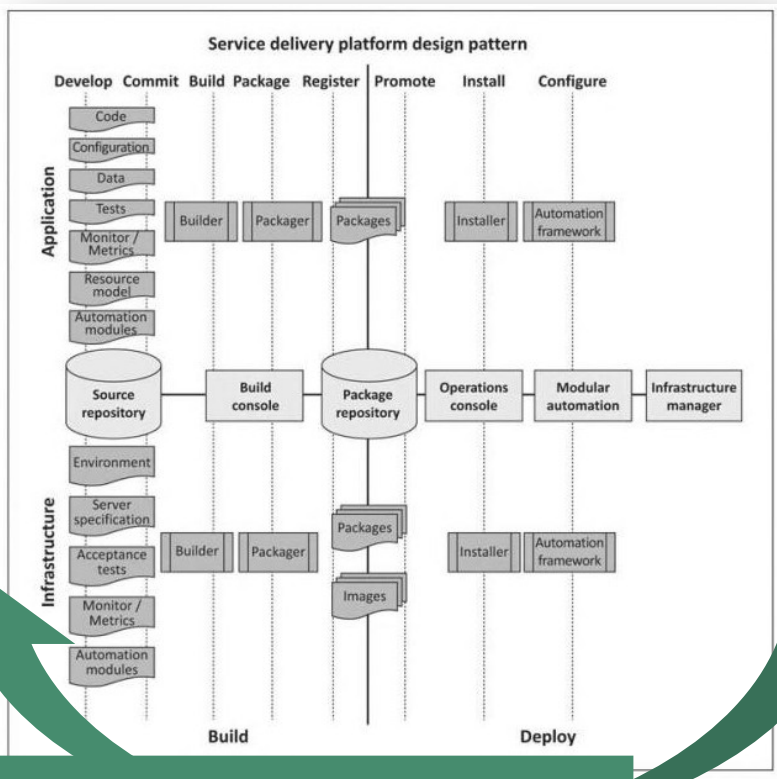
Grady Booch's Two-Tier Process Structure



Characteristics:

- Premise: we know what the product should be, so delivering what's asked for is the differentiator, long-term support requires stability.
- Long cycles (12-18 months)
- Process thinking over product thinking
- Up-front specification with “Phase-gates”
- Operational “silos”
- Promises over learning
- The Leaky Abstraction: Deliverables, not flow

Comparing straw men, “DevOps”: learning the best solution is more important than knowing what you want



Don't forget the feedback Loop!

Characteristics:

- Premise: we don't know what the product should be, so treat software development like a continuous learning factory, using agility as your differentiator
- Tiny batches (1-7 days)
- Product thinking over process thinking
- Study how the software is used, pivot often
- Reduction of silos
- Learning over specifying
- The Leaky Abstraction: flow, not deliverables



Technologies
in use

Focusing on the delivery pipeline is the current best place to start tools-wise, but “culture” is the parallel challenge: **changing corporate process**

The 3 ways of DevOps

1. Workflow, value-stream focus
2. Improve feedback, stay close to the knowledge
3. Continual experimenting & learning

Technical Traits

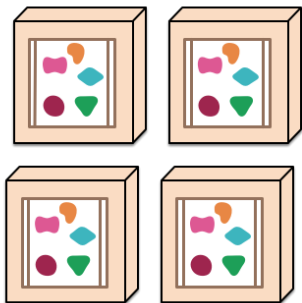
- Cloud Centric:
 - Programmable everything
 - Automate everything
- Version control everywhere
- CI/CD
- ALM, SDLC
- DIY PaaS...?

DevOps application architectures are trending towards “microservices,” hiding the state, relying on APIs

A monolithic application puts all its functionality into a single process...

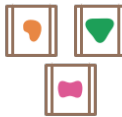


... and scales by replicating the monolith on multiple servers

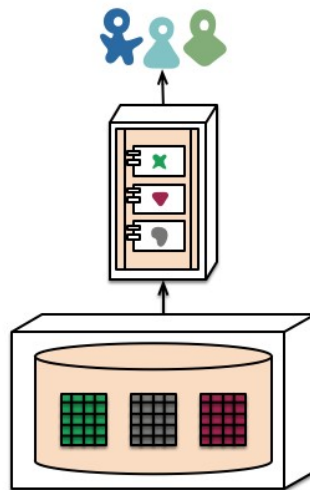
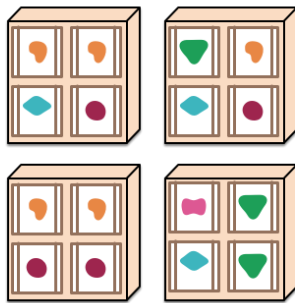


Monoliths vs. Microservices

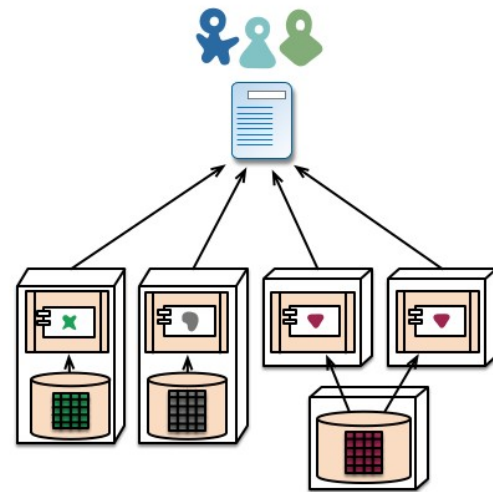
A microservices architecture puts each element of functionality into a separate service...



... and scales by distributing these services across servers, replicating as needed.



monolith - single database



microservices - application databases

Storage policy & technology is decentralized

The emerging DevOps stack & storage

Automation and monitoring

- > Chef, Puppet, Ansible, Salt
- > APM and log management

Containers & cloud platforms

- > Public & private cloud, VPS, hosting
- > Docker, CoreOS, Mesosphere, etc.

PaaS – less focused on pure DevOps

- > Public PaaSes, Cloud Foundry (ActiveState, Pivotal, IBM, HP)
- > DIY platforms and PaaS
- > Docker-based PaaS with Flynn, Deis

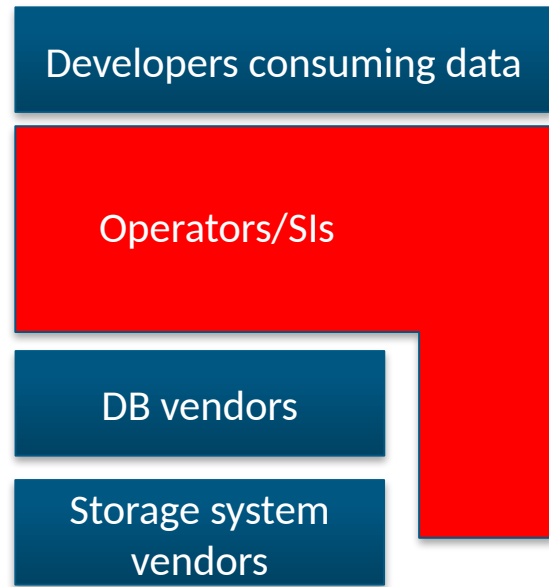
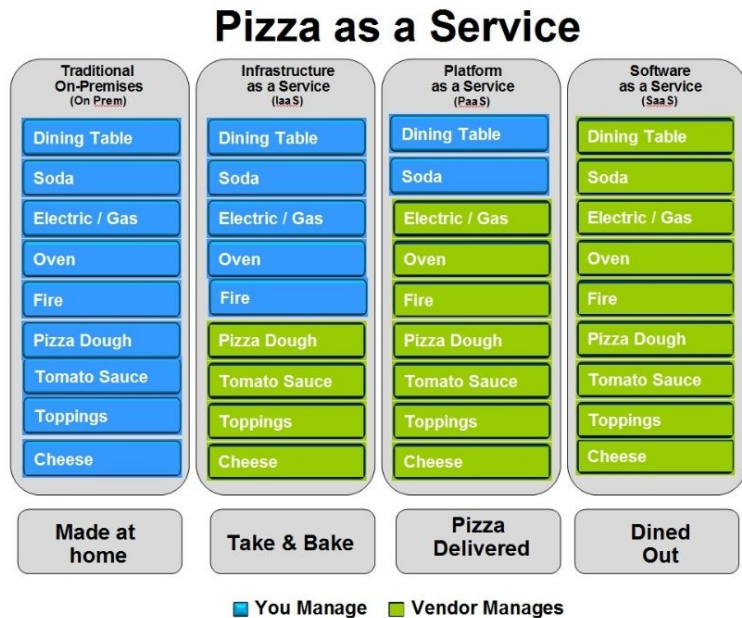
ALM & more – each evolving quickly to DevOps.

- > Atlassian, TaskTop, Codenvy, Bowery, SmartBear, StackStorm, CloudBees, Perforce, IBM, CA Technologies, ServiceNow, AWS, VMware + JFrog, other large players

Storage implications & side-effects:

- New databases: Mongo, Casandra, Hadoop, Riak, Riddis, MySQL, Postgres, DBaaS
- Interop fragmentation – too many APIs, file-systems, etc.
- “Good enough” over “enterprise grade”
- Orchestration, policy, management often left up to developers

Someone still has to make & bake the pizza, who will it be?



A product centric view of cloud hides the hard work:
making it all work together...someone's gotta do it

Storage and DevOps

- A developer-mindset drives DevOps
- APIs are everything – if there’s not an API, developers will ignore it
- Programmable everything – once APIs are established, developers will automate it
- Speed is important – focus on shorter provisioning time, you don’t want to be the bottleneck
- Developers assume enterprise grade storage is a commodity – little DR and backup thinking, worrying about storage HA – the “cloud” just does that

RESTful

Create a brown dog named Al

```
POST /dogs
name=Al&furColor=brown
```

Response

```
200 OK

{
  "dog":{
    "id": "1234",
    "name": "Al",
    "furColor": "brown"
  }
}
```

Rename Al to Rover - Update

```
PUT /dogs/1234
name=Rover
```

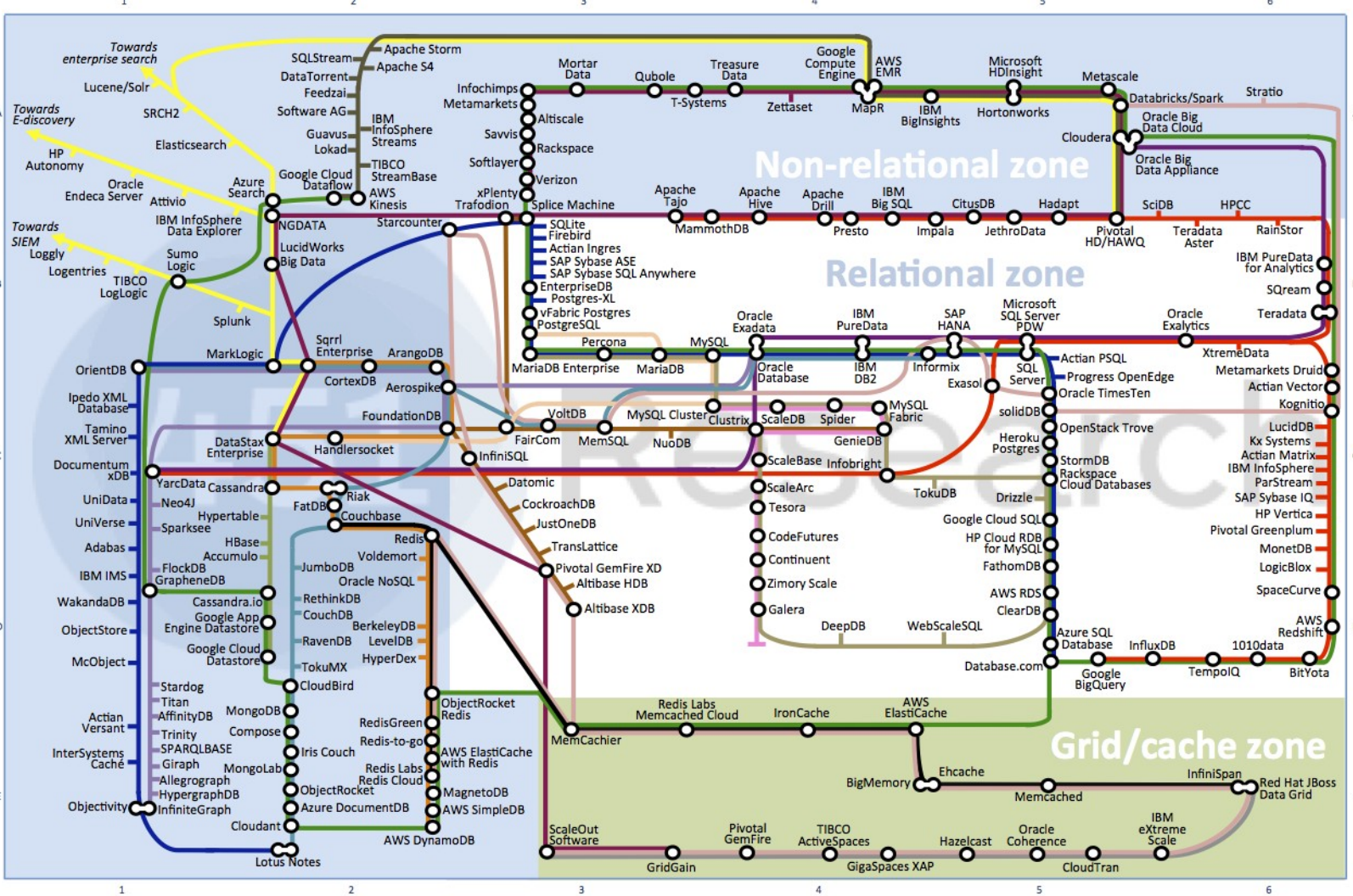
Response

```
200 OK

{
  "dog":{
    "id": "1234",
    "name": "Rover",
    "furColor": "brown"
  }
}
```

Data Platforms Map

October 2014



- Key:**
- General purpose
 - Specialist analytic
 - -as-a-Service
 - BigTables
 - Graph
 - Document
 - Key value stores
 - Key value direct access
 - Hadoop
 - MySQL ecosystem
 - Advanced clustering/sharding
 - New SQL databases
 - Data caching
 - Data grid
 - Search
 - Appliances
 - In-memory
 - Stream processing

<https://451research.com/dashboard/dpa>

The background of the slide is a stylized circuit board pattern. The top half is orange and the bottom half is blue. The circuit lines are white and light blue, creating a complex, interconnected network of paths and nodes.

Thank You!

@cote – <http://cote.io>



NEW YORK
LONDON
BOSTON
WASHINGTON, D.C.
SAN FRANCISCO