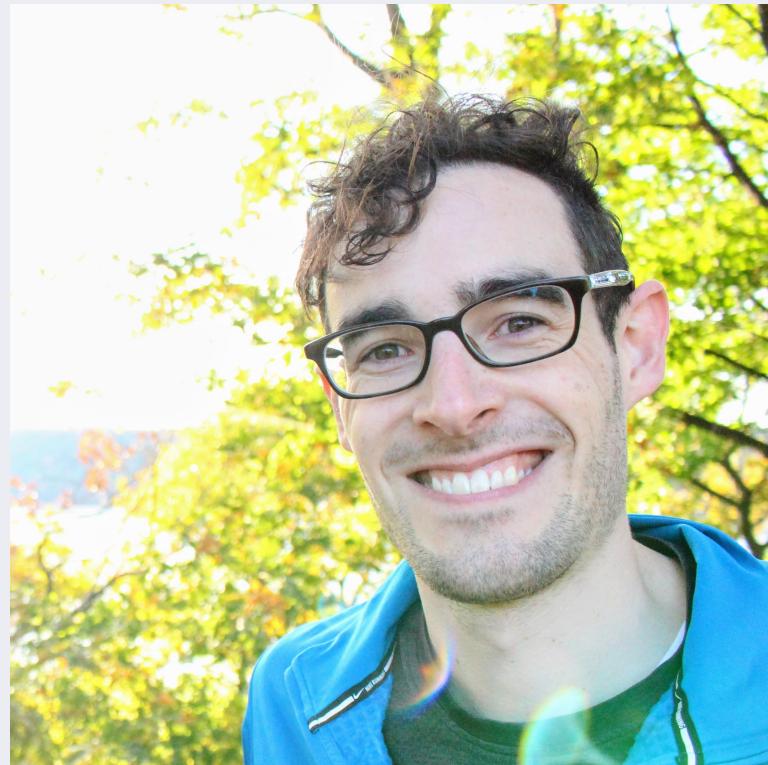


PARIVEDA
SOLUTIONS



Partner
Network



Your business
doesn't have
technology problems



PARIVEDA
SOLUTIONS



Partner
Network

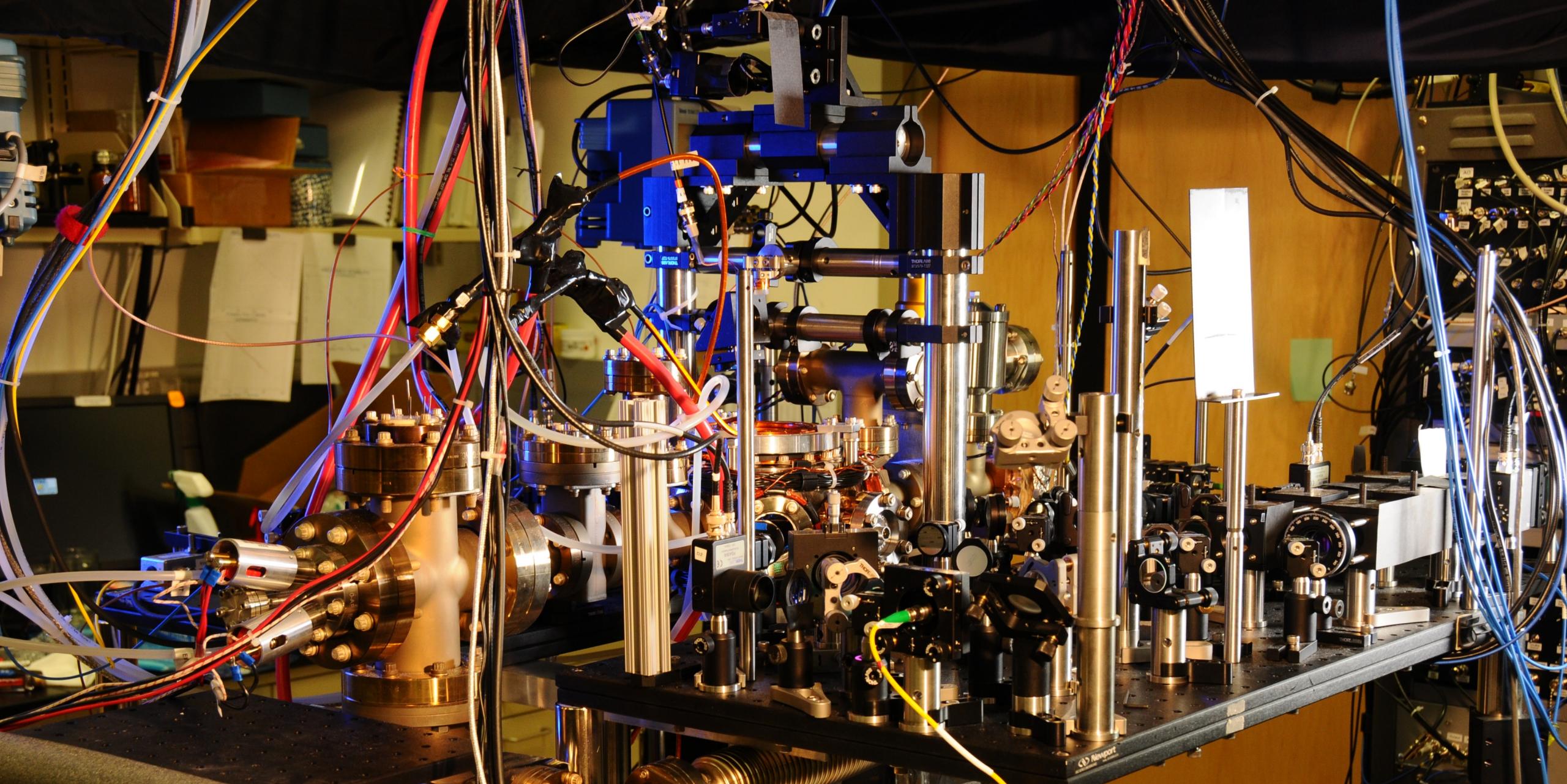




**PARIVEDA
SOLUTIONS**



amazon
web services
Partner
Network



PARIVEDA
SOLUTIONS



Partner
Network

You still have
problems



PARIVEDA
SOLUTIONS



Partner
Network

Tom



":)"

Male
30 years old
Santa Monica,
CALIFORNIA
United States

Last Login:
8/27/2006

[View My: Pics](#) | [Videos](#)

Contacting Tom

[Send Message](#)

[Forward to Friend](#)

[Add to Friends](#)

[Add to Favorites](#)

[Instant Message](#)

[Block User](#)

Tom is in your extended network

Tom's Latest Blog Entry [[Subscribe to this Blog](#)]

private profiles ([view more](#))

Top 8, 16, 20, 24 friends :) ([view more](#))

MySpace Concert & Parties -Georgia, Orlando, Miami! ([view more](#))

In Stores Today - MySpace Records Vol. 1 ! ([view more](#))

MySpace Records - in stores soon! ([view more](#))

[[View All Blog Entries](#)]

Tom's Blurbz



PARIVEDA
SOLUTIONS



Partner
Network



PARIVEDA
SOLUTIONS



Partner
Network



PARIVEDA
SOLUTIONS



Partner
Network

solving
technology
You don't have
technology problems
problems isn't
your business



PARIVEDA
SOLUTIONS



Partner
Network

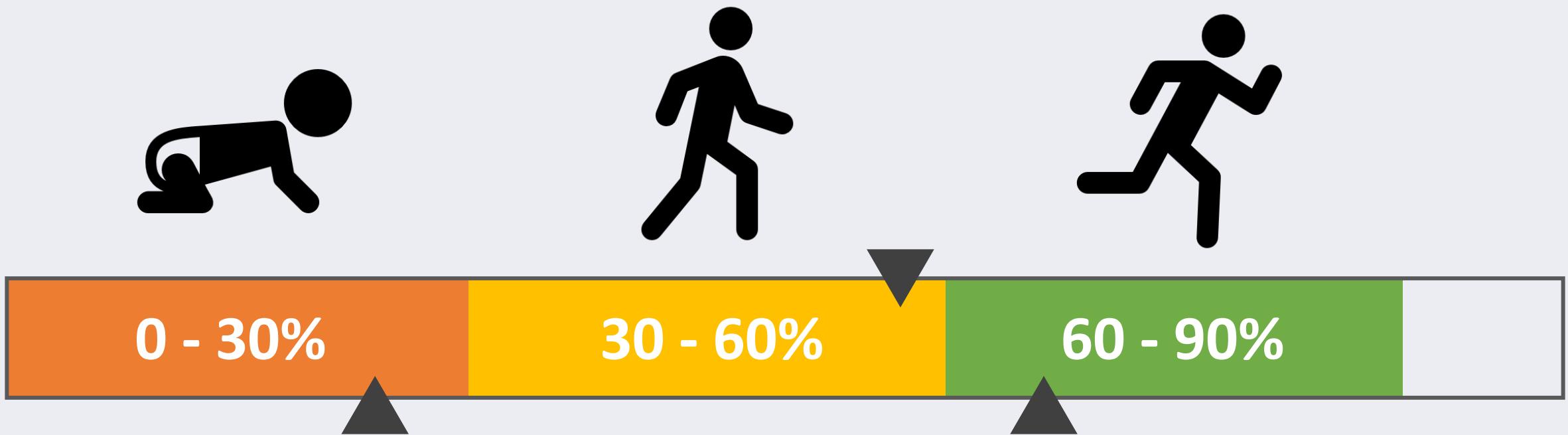
How much time does IT spend on solving business problems?



PARIVEDA
SOLUTIONS



Partner
Network



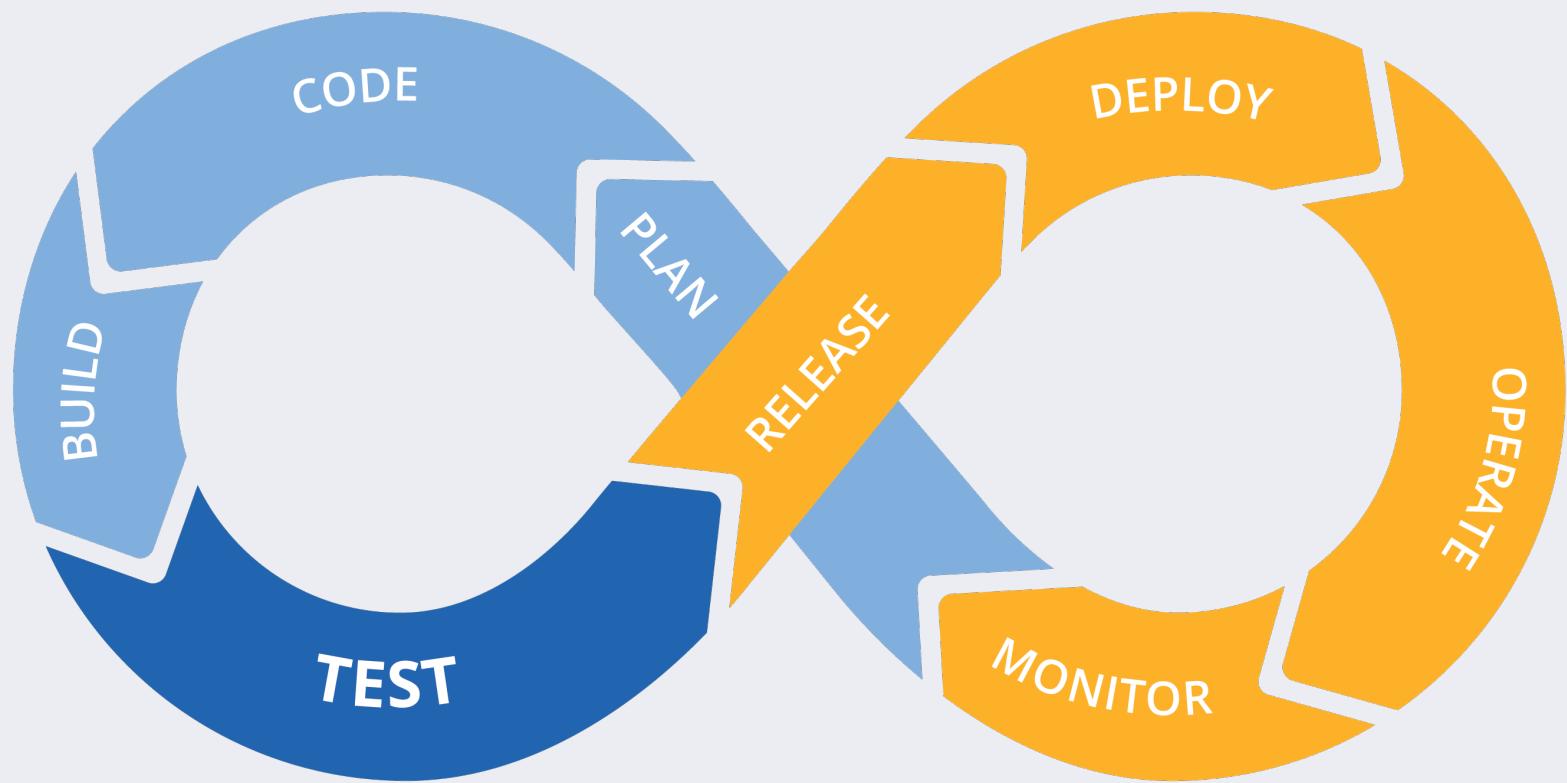
We have nothing new
to offer here



PARIVEDA
SOLUTIONS



Partner
Network



Metric	I	II	III
Culture	- Execute defined processes		- Learning organization - Feedback loops
Organization	- Technology & platform teams	- Lifecycle teams	- Autonomous product lifecycle teams
Projects	- Large, cross-team projects - Multi-year	- Series of small, discrete projects - Some cross-cutting	- Focus on products - Mission-based - Autonomy and experimentation - Iterative
Data decision making	- HiPPO	- Structured, schedule reporting - Frequently in silos	- Data-driven - Everyone in organization has access to all data
Delivery	- Monthly (+) - Manual	- Weekly - Automated in some environments, for some applications	- Multiple times a day - Fully automated
Cloud adoption	- Exploring the cloud	- Moving to the cloud	- 100% cloud adoption
Operations	- Analyzing to see if something broke		- Analyzing to see if business value created
Security	- Security theater	- Firewall security blanket	- Publish security practices



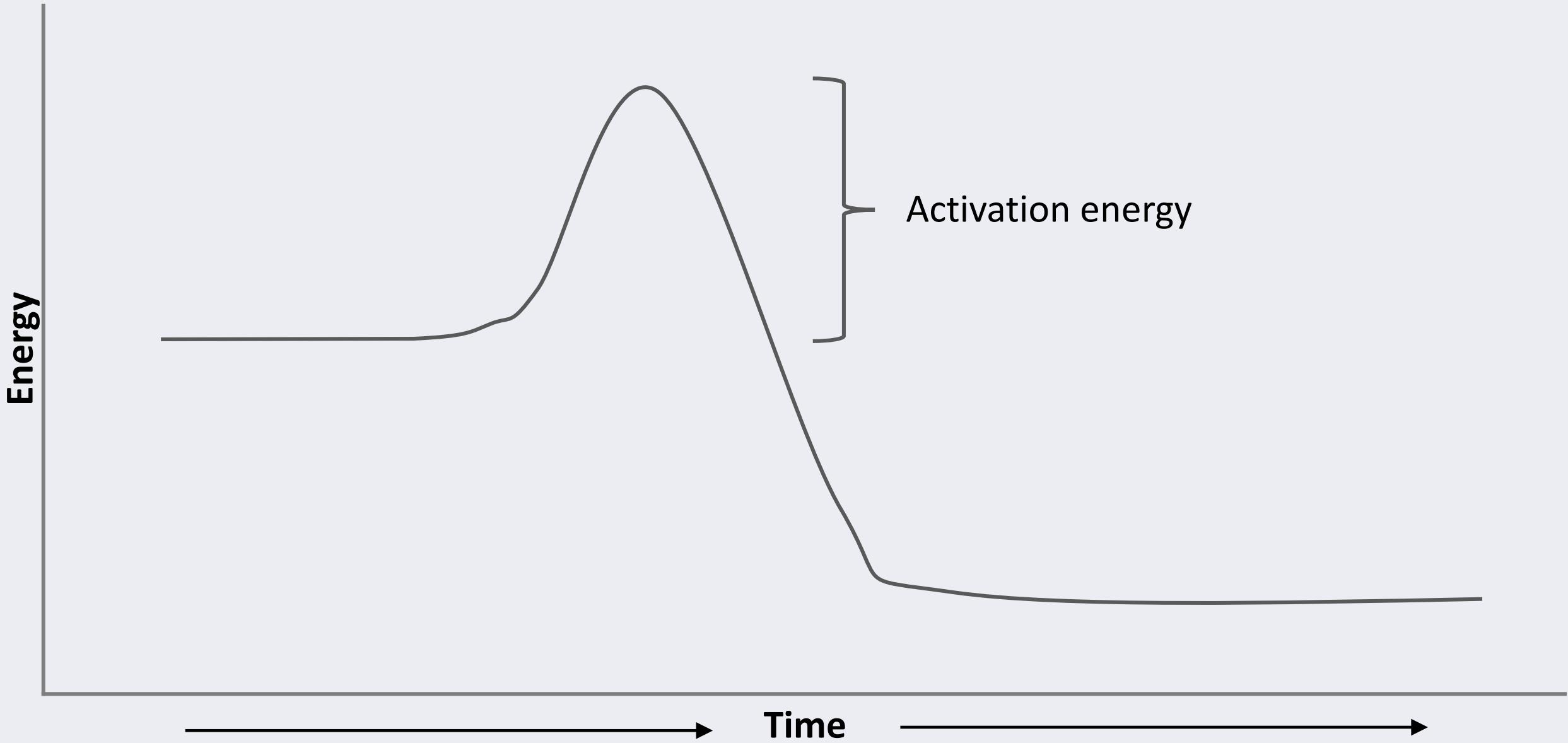
If it's understood, why
are so few doing it?

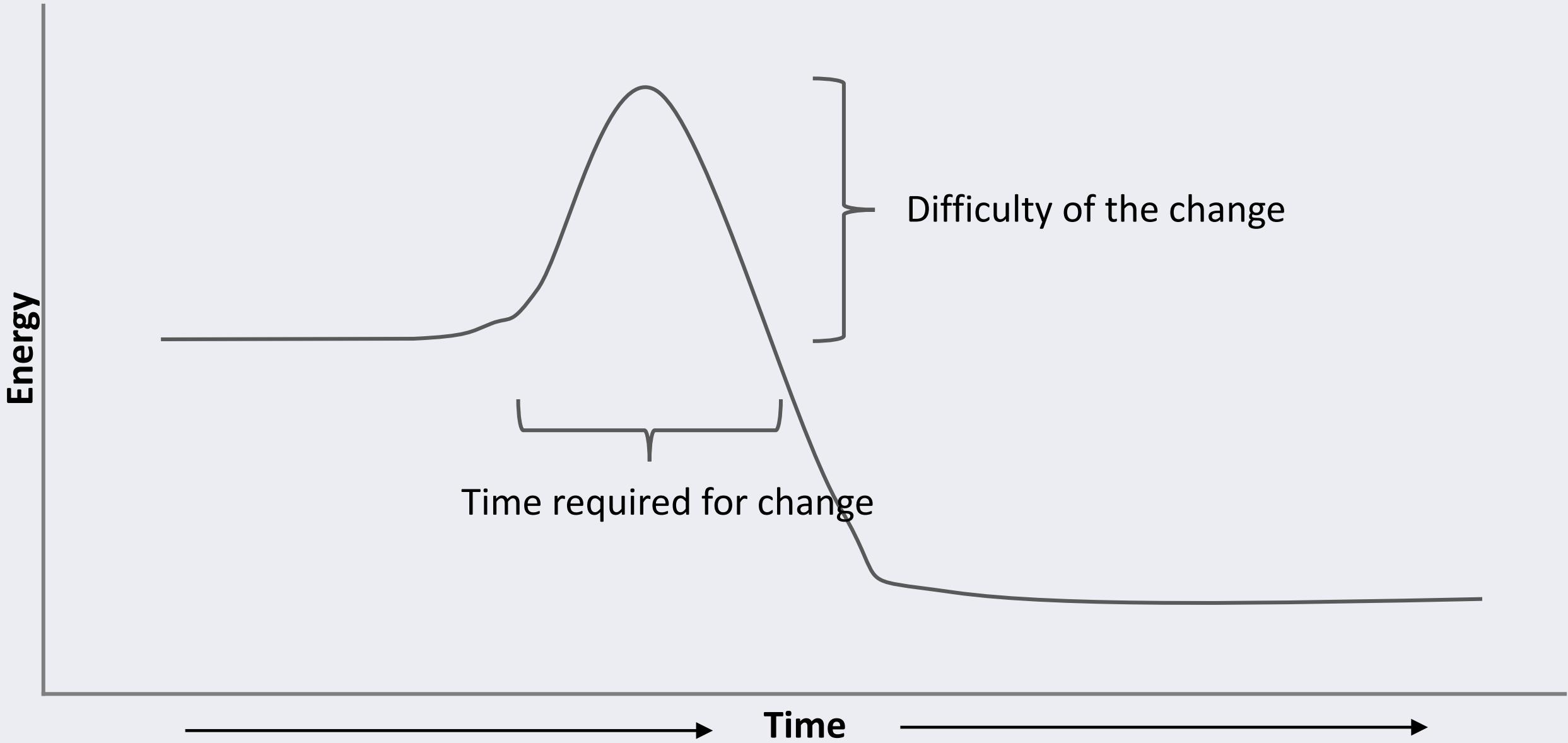


PARIVEDA
SOLUTIONS



Partner
Network

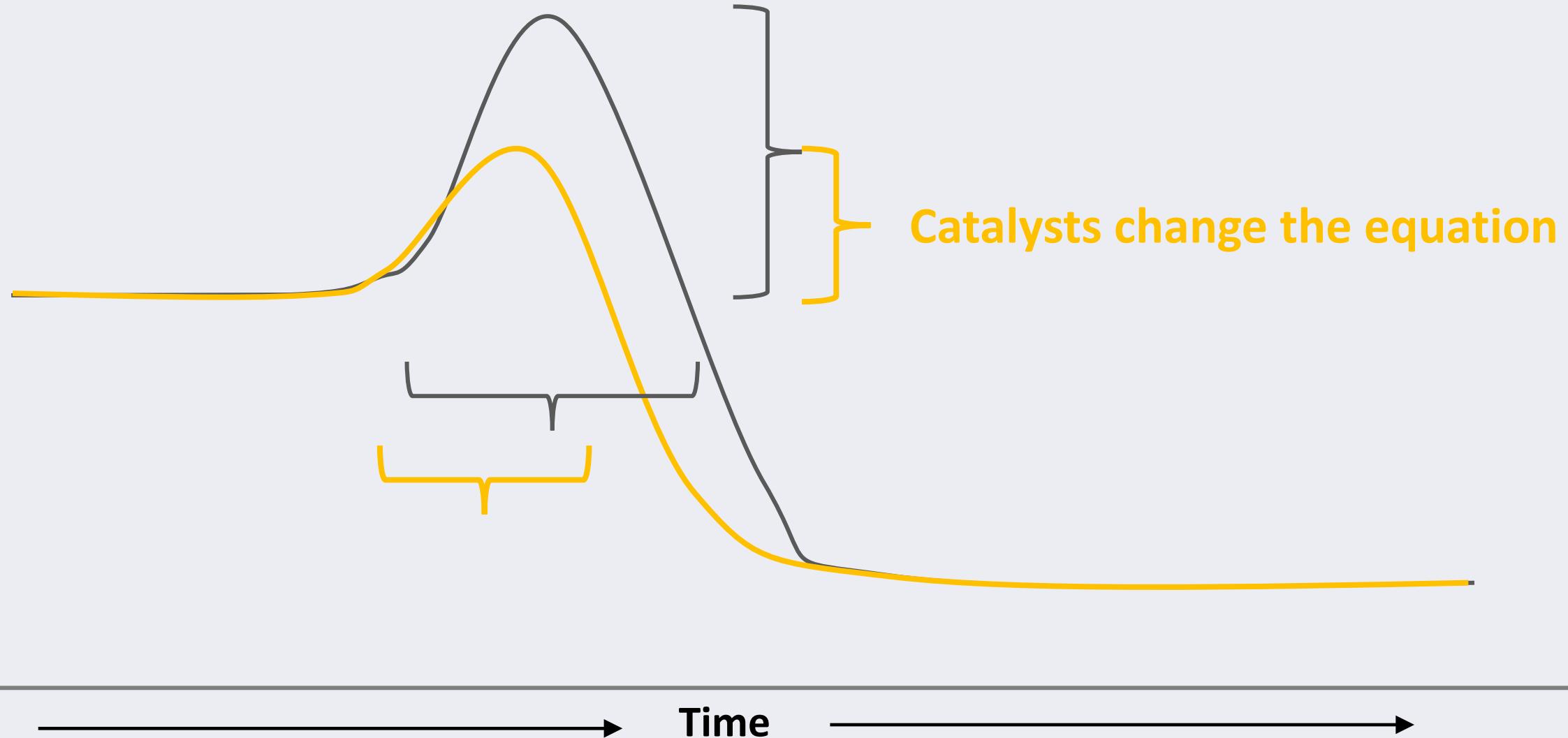






Energy

Time



What is serverless

- Architectural patterns built using utility compute services
 - True variable cost
 - Infinite scalability
- Lambda
 - Function as a service
 - Infinitely scalable out of the box
 - Pay for duration of function invocation – per 100 ms of usage



Benefits of serverless

- True variable cost computing
- Time to market goes from months or weeks to days or hours
- Built in support for things you should be doing already
 - Infrastructure as code
 - API driven development and data analysis to inform product development
 - Small teams



Serverless as catalyst

- It's like magic – haven't met a developer who didn't love it
- Easy to use, easy to experiment with
- Low cost to learn, develop with, experiment with – only cost is time, no infrastructure or hosting
- Starts a chain reaction across your organization



Serverless as Trojan horse

- It's like magic – haven't met a developer who didn't love it
- Easy to use, easy to experiment with
- Low cost to learn, develop with, experiment with – only cost is time, no infrastructure or hosting



Who should do it

- Everyone, but the value will you get is inversely related to the investments you have made in DevOps and the cloud already
- For mature DevOps and cloud companies the benefits will be more incremental because they are already following best practices
- For companies without strong DevOps or cloud practices the value is much higher



How to try it

- If the people you've hired have built a mature DevOps and cloud company they already know about serverless and love it, and probably are already using it
- If the people you've hired haven't built a mature DevOps and cloud company
 - If they would like to but can't for whatever reasons, they likely know about serverless and have used it on side projects and love it
 - If they aren't pushing for DevOps or cloud there's a good chance they don't know anything about serverless (or DevOps practices in general)
- Know the group you are working with



Challenges to implement

- Retailers run large off the shelf software packages, how to incorporate serverless models around it



GameStop story



PARIVEDA
SOLUTIONS



Partner
Network

Where it's already started



NORDSTROM

Coca-Cola



StanleyBlack&Decker



MAJOR LEAGUE BASEBALL

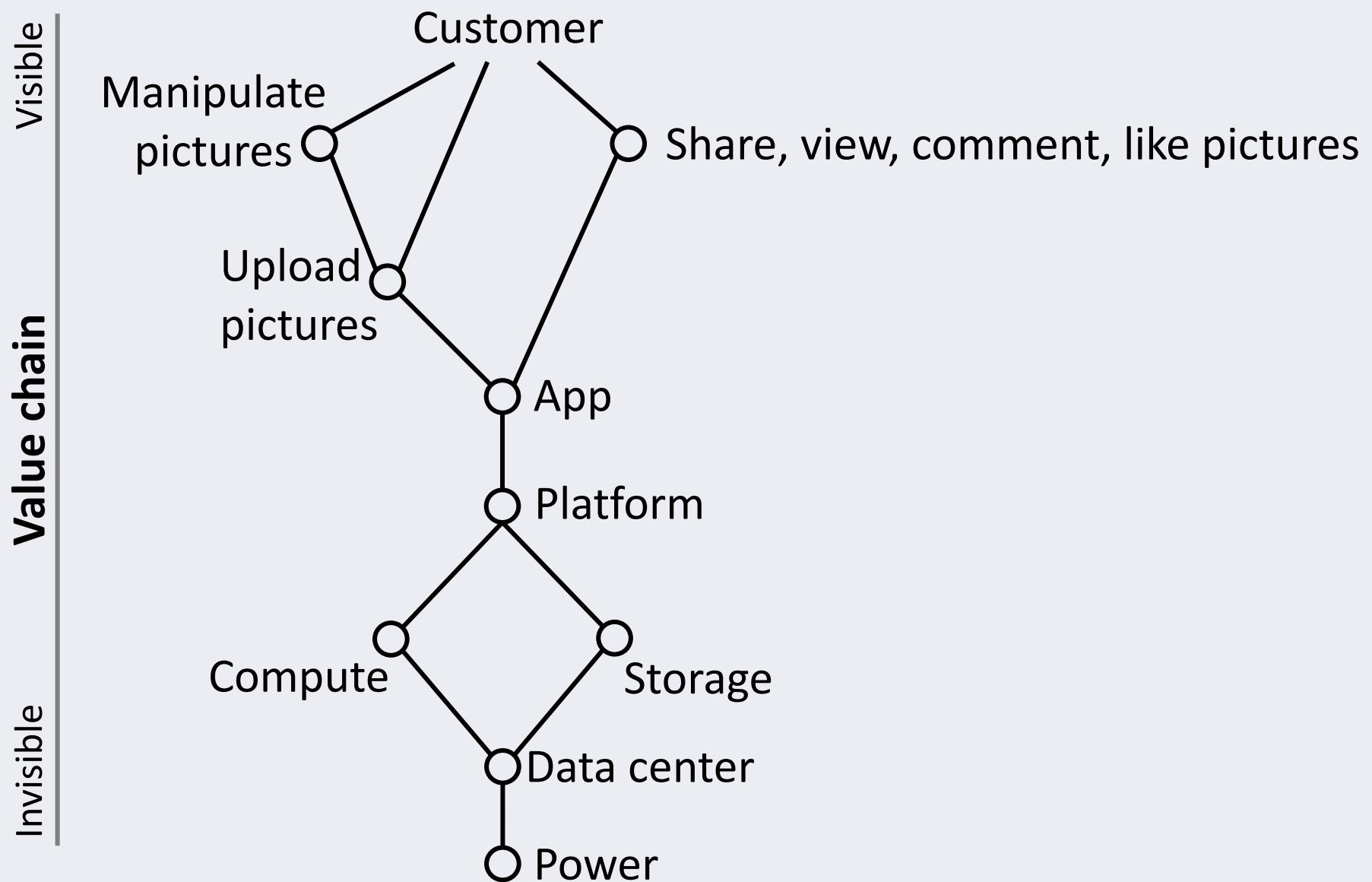
The Seattle Times

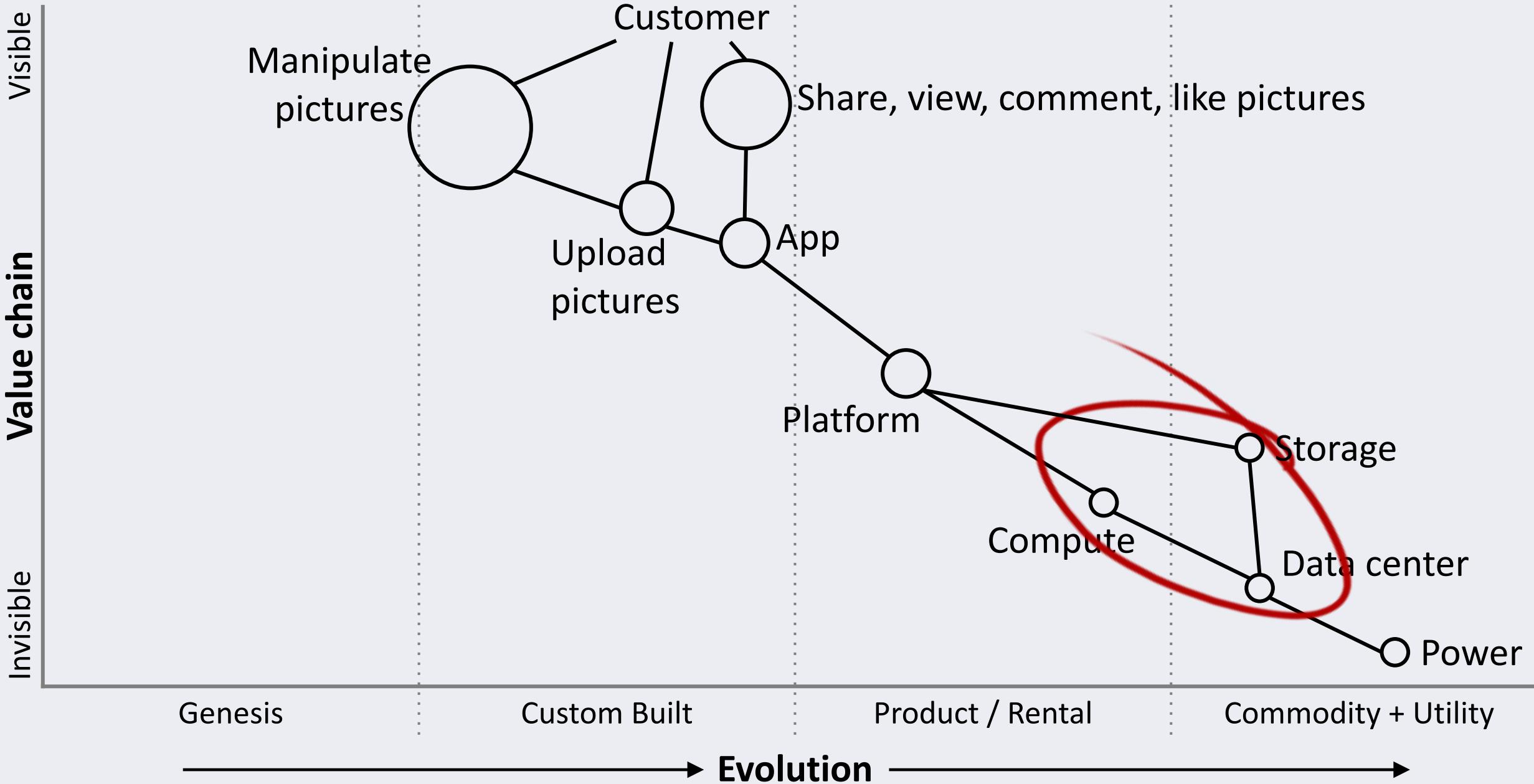


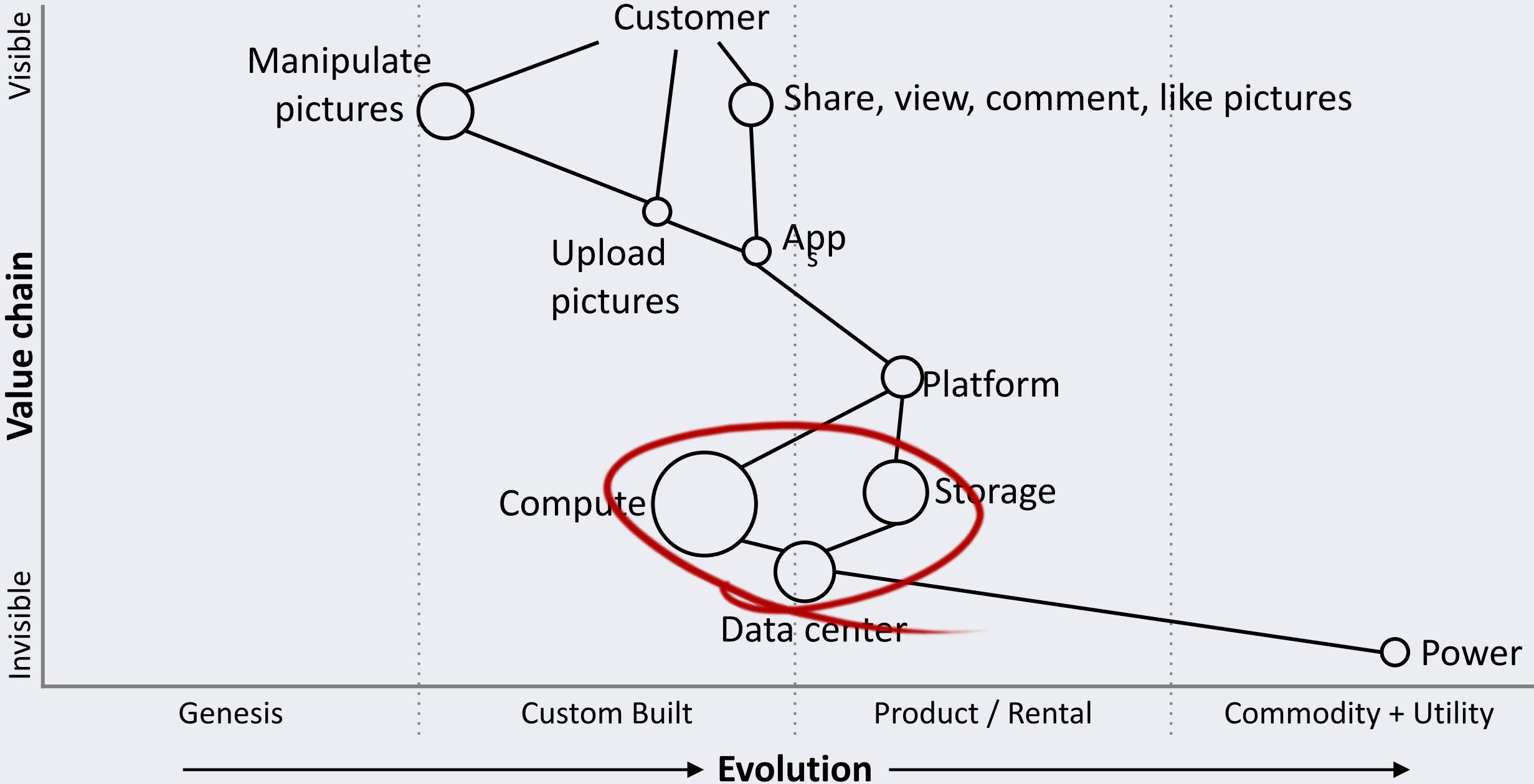
PARIVEDA
SOLUTIONS



Partner
Network



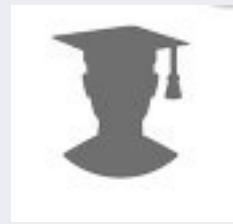




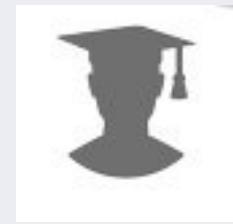
What does it mean for my people?



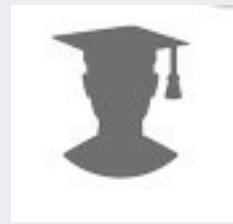
Solutions Architect
(Bit blueprint creator)



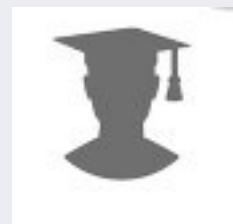
Data Architect
(Stores bits, manages access)



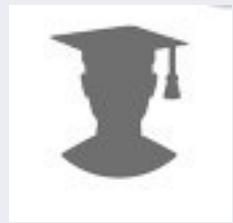
Developer
(Describes how to move bits)



Team Lead
(Keeps bit machine well-oiled)



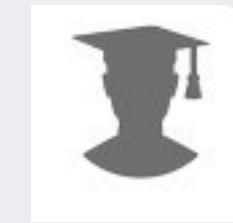
Product owner
(Talks to customer, attempts to talk to bit-team)



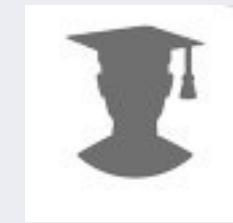
Release Engineer
(Moves bits)



Ops Engineer
(Watches bits)



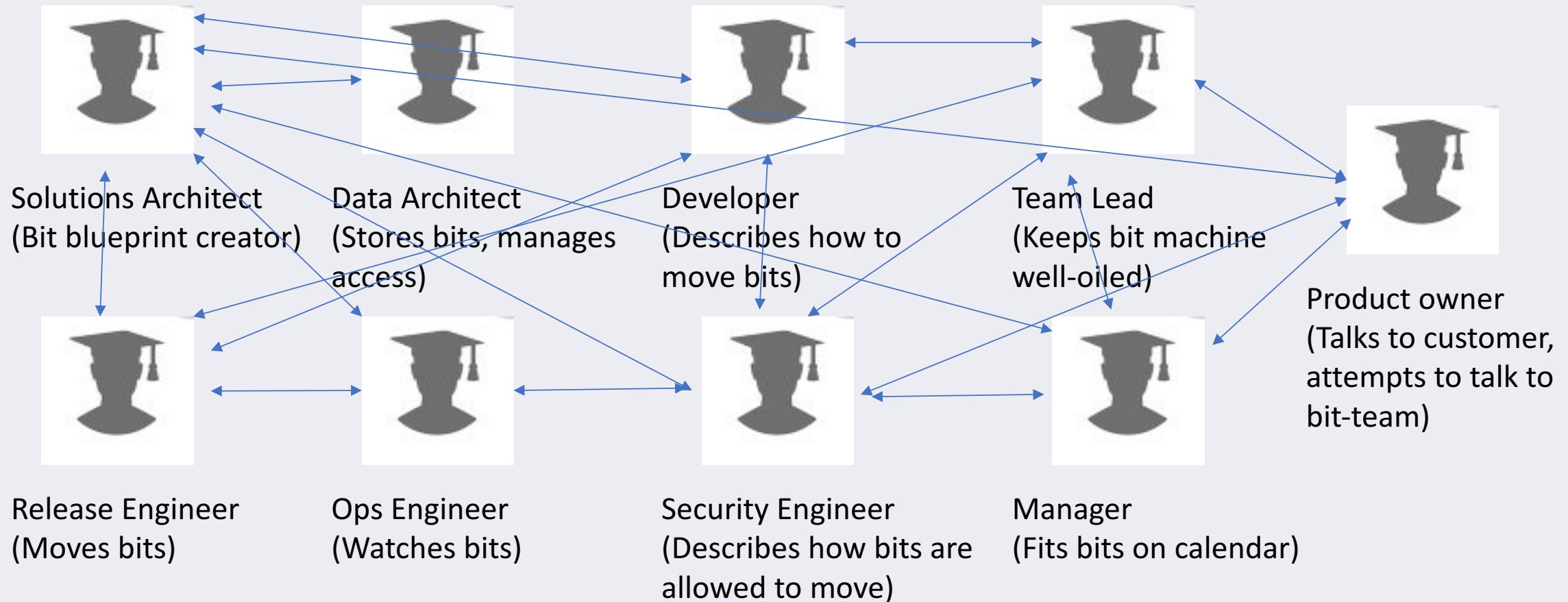
Security Engineer
(Describes how bits are allowed to move)



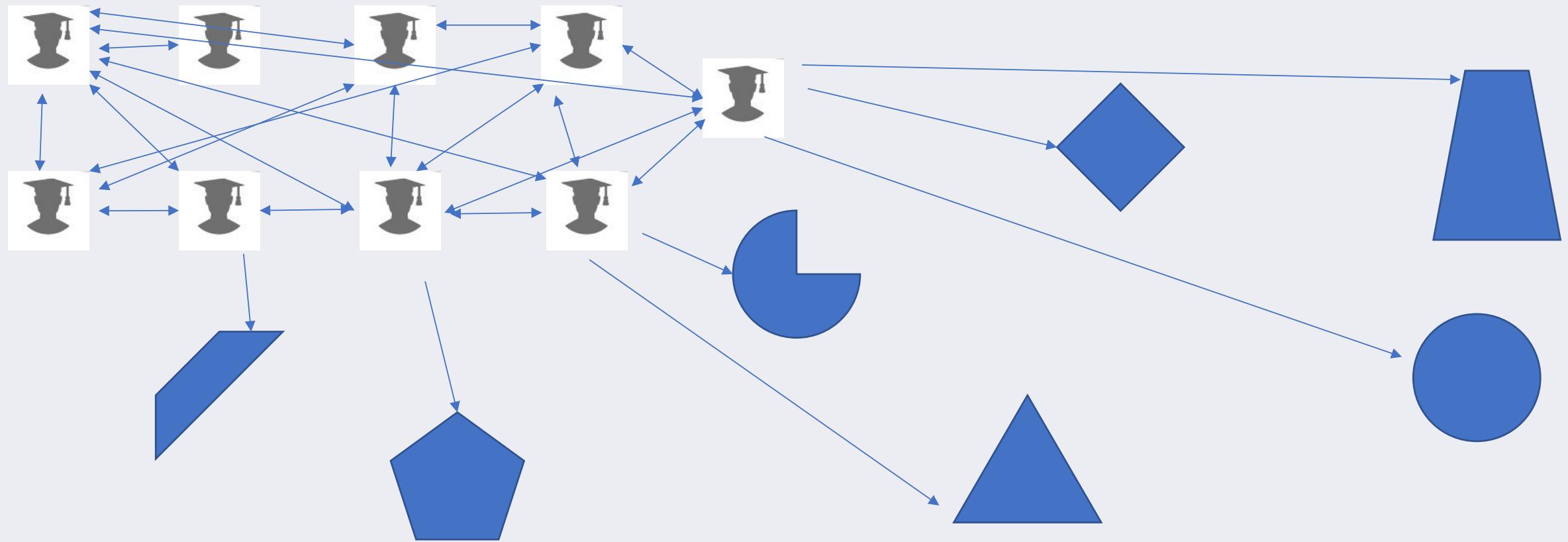
Manager
(Fits bits on calendar)



And how they communicate?

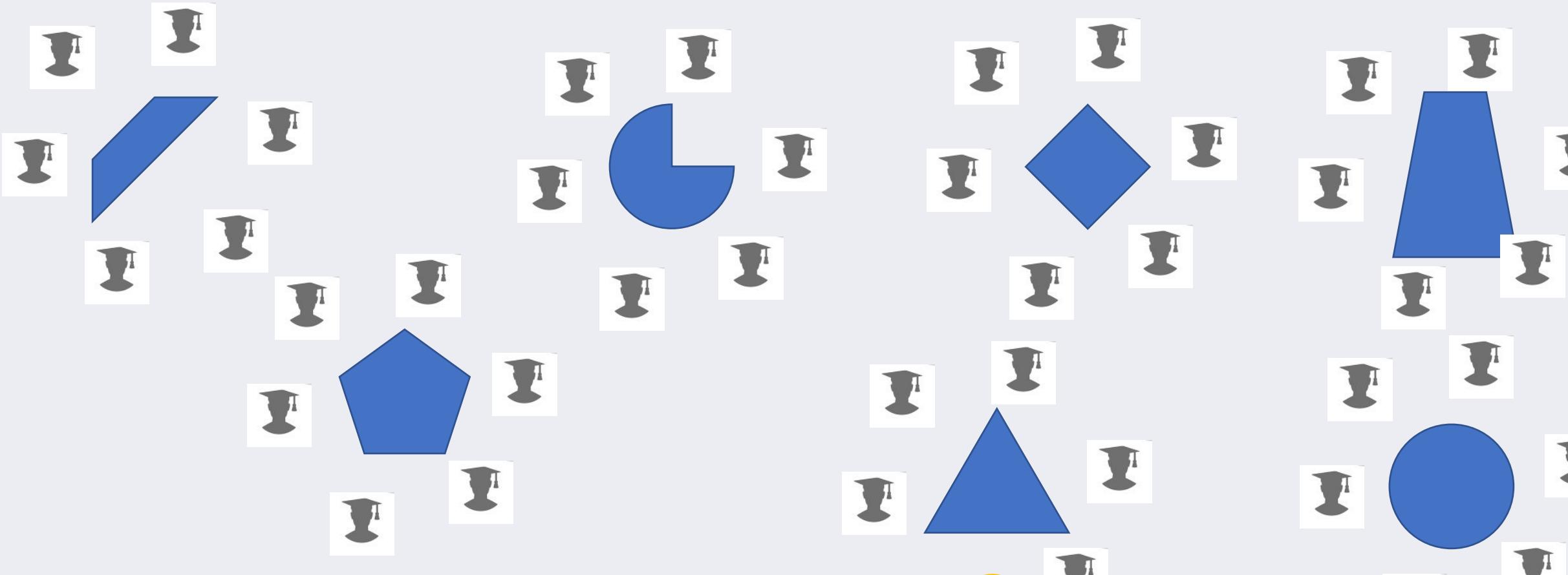


And what products do they support?



Is this customer-centric?

Product Teams



More customer-centric, means ☺'er customers



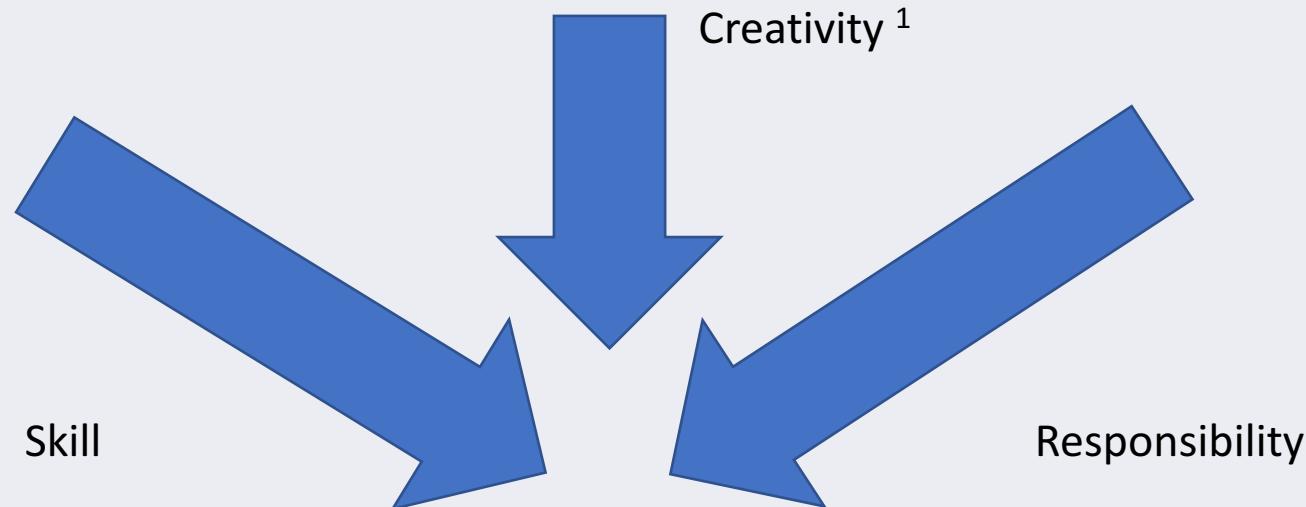
PARIVEDA
SOLUTIONS



Partner
Network

Don't fire your way to innovation

- Instead, seek to merge skill with responsibility
- Your people, instead of being glorified bit managers, can do what they are good at – injecting creativity



¹ what people are still better at than computers, for the time being



But what about governance?

- “We standardized on Java!”
- “We only run [insert OS here]!”
- “We only use [insert framework here]!”
- The above are the result of trying to solve the Principal-agent problem: “the agent knows some things that the principal does not, and it is too costly for the principal to monitor every action that the agent takes” (“Misbehaving: The Making of Behavioral Economics.”)
- So instead of monitoring every action, the principal (IT Governance) reduces the jurisdiction(?), and thus cost, to govern.



The dark underbelly of limiting choice

- By being prescriptive about standards, you've succeeded in reducing the governance cost
- (Applause!)
- But you've prohibitively raised the cost of experimentation.
- Your talented high-dollar developers can only use the hammer you've provided.
- (Insert Hammer/Nail)
- Don't be surprised if your talent leaves
- Or worse, your talent only starts to see nails.



So you are proposing the wild west, except with modern weapons?

- No.
- With evolutionary architectures, strict data access policies, serverless technologies, and infrastructure-as-code, you can simultaneously allow variation and have cheaper, more reliable IT governance than you ever imagined possible.
- The tools aren't here yet, but they're coming quick.



Adoption Paths

- Build vs Buy equation begins to change
 - If you have to customize *anything*, you may want to build
- Come back to activation energy
 - Experiment with internal tools
 - Greenfield “Shining City on the Hill” model
 - Skunkworks
 - Others?

