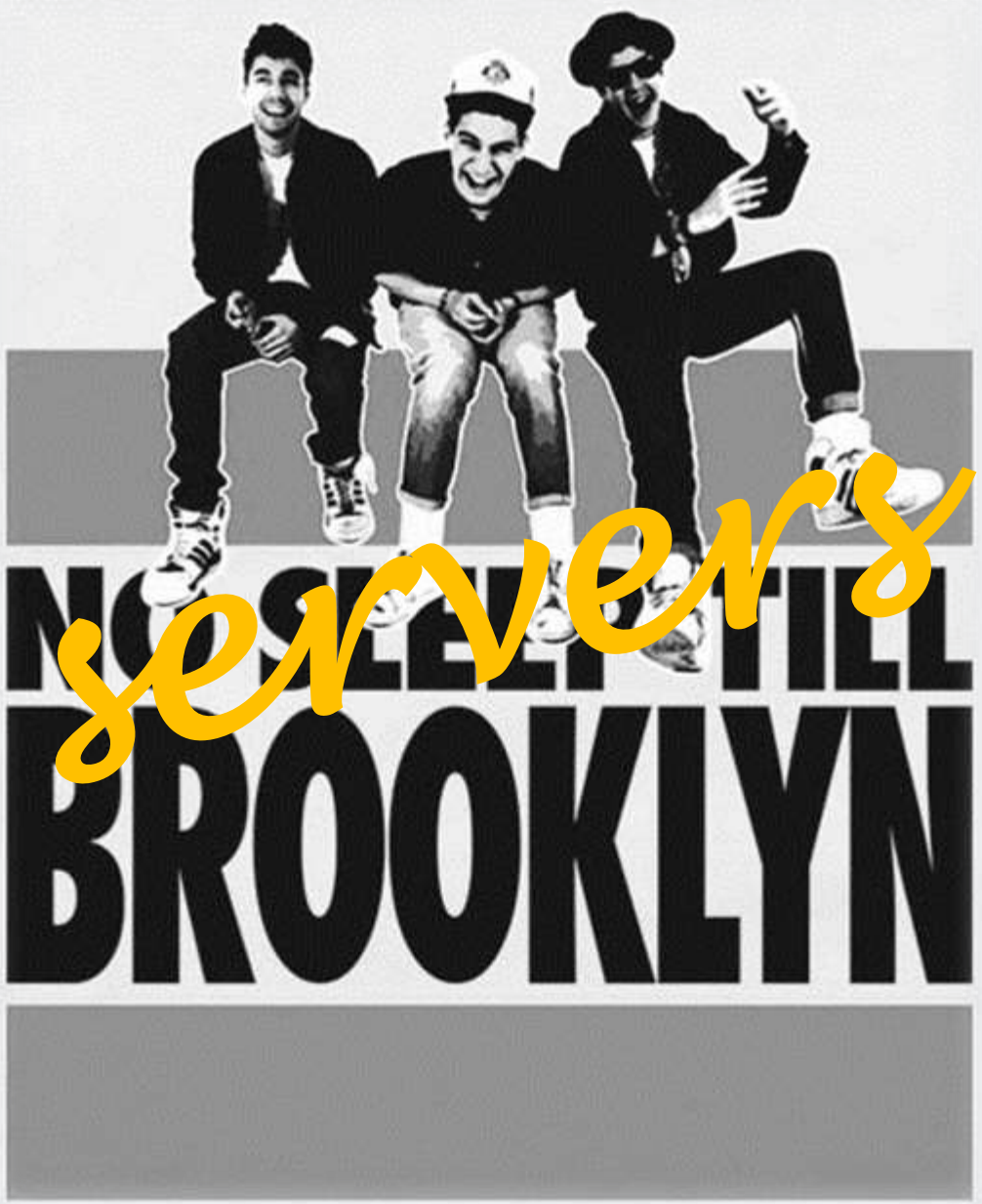
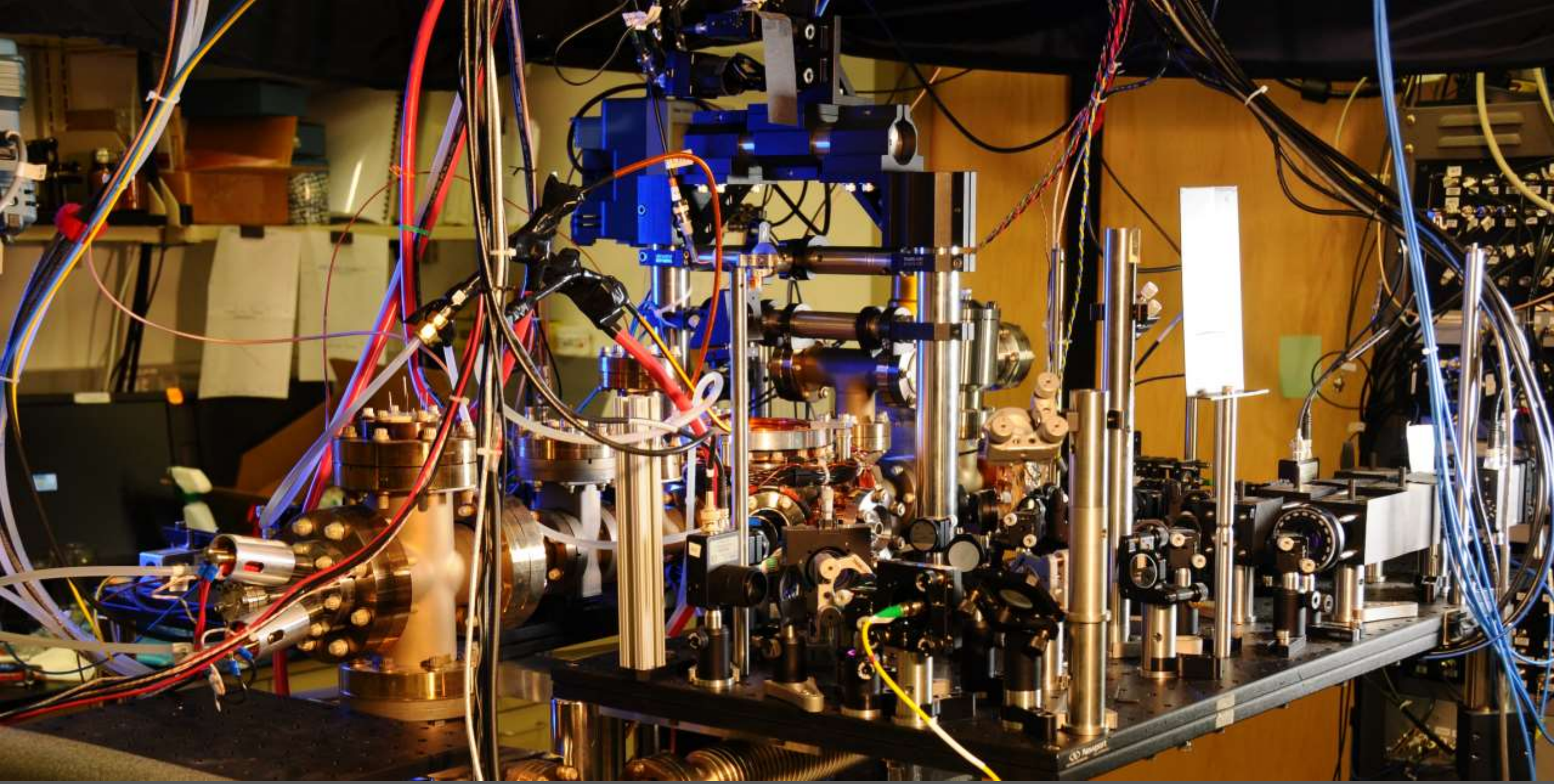

servers
BROOKLYN

Your business
doesn't have
technology problems

PARIVEDA SOLUTIONS

amazon web services | Partner Network



You still have problems

MySpace.com | Home The Web MySpace Search Help | SignOut

Home | Browse | Search | Invite | Film | Mail | Blog | Favorites | Forum | Groups | Events | Videos | Music | Comedy | Classifieds

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**)

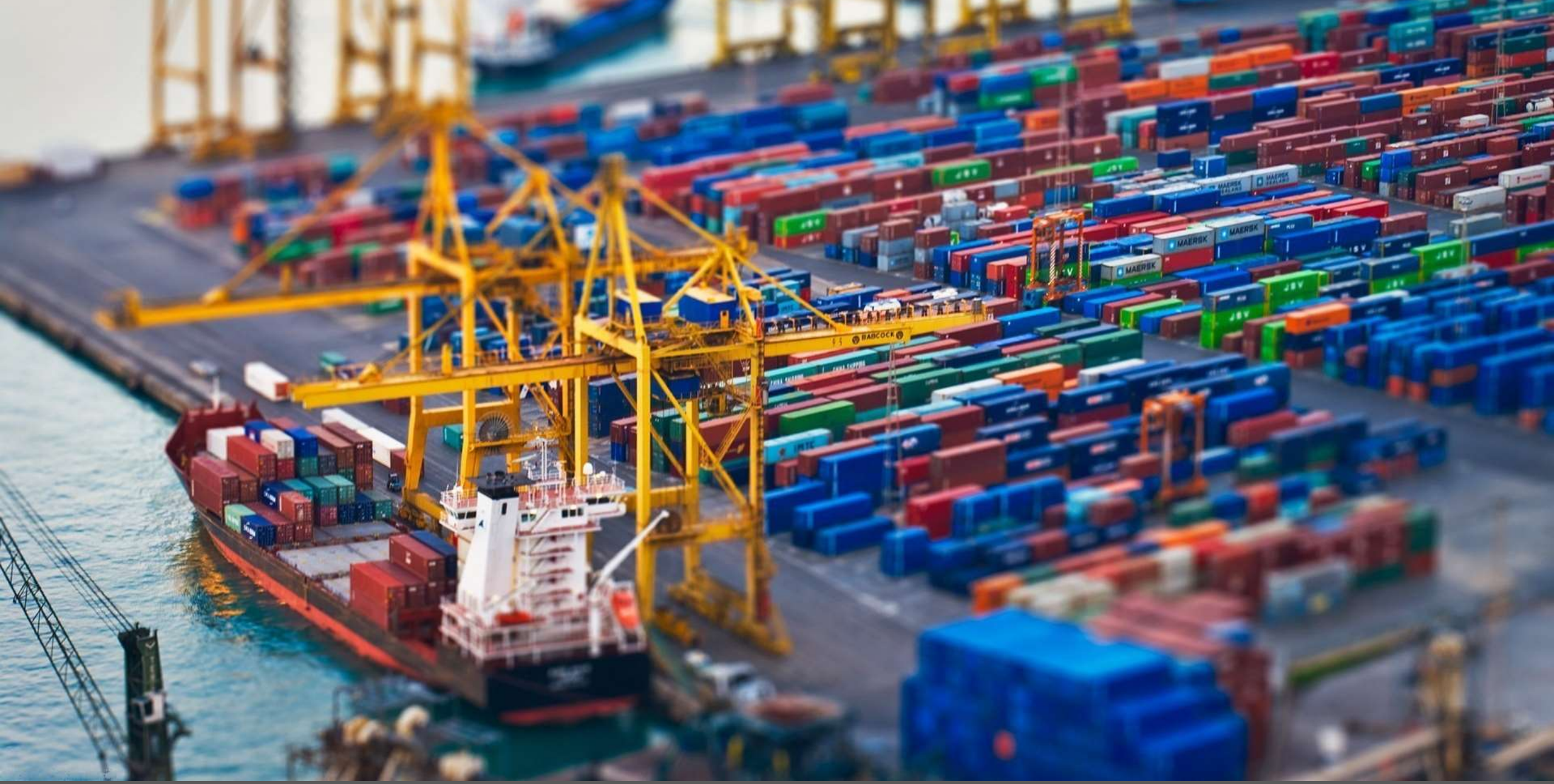
MySpace Records - in stores soon! (**view more**)

[**View All Blog Entries**]

Tom's Blurbs

PARIVEDA SOLUTIONS
amazon web services | Partner Network

BLOCKBUSTER
®



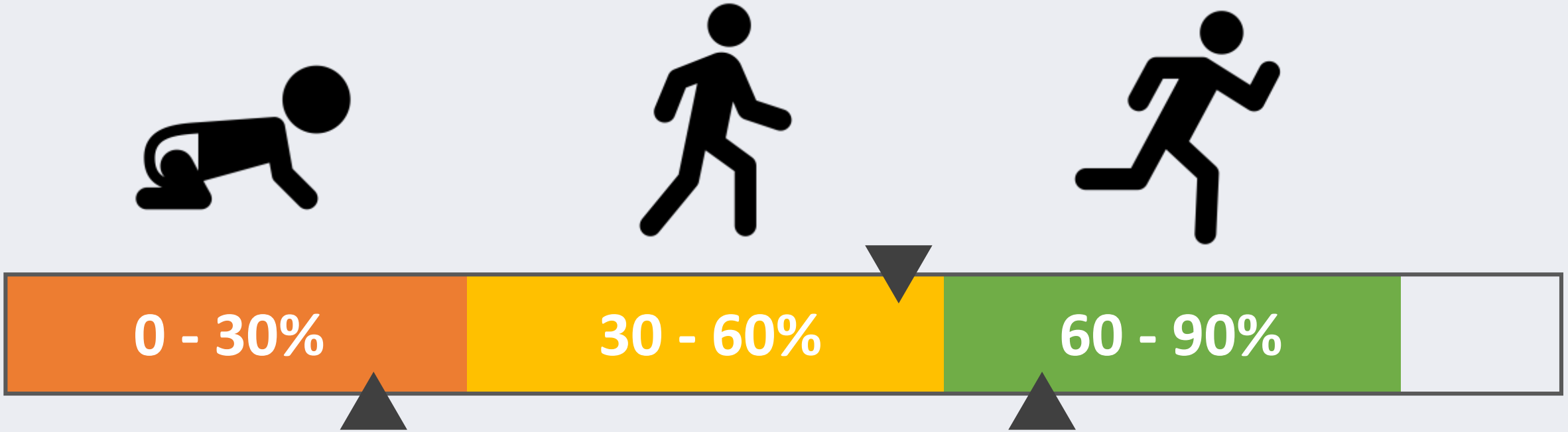
PARIVEDA SOLUTIONS

amazon web services | Partner Network

Solving technology problems isn't your business

You don't have technology problems

How much time does IT spend on solving business problems?


0 - 30%
30 - 60%
60 - 90%

DevOps & cloud best practices maturity model

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

Serverless

- Patterns
- Framework
- AWS

Define what we're talking about

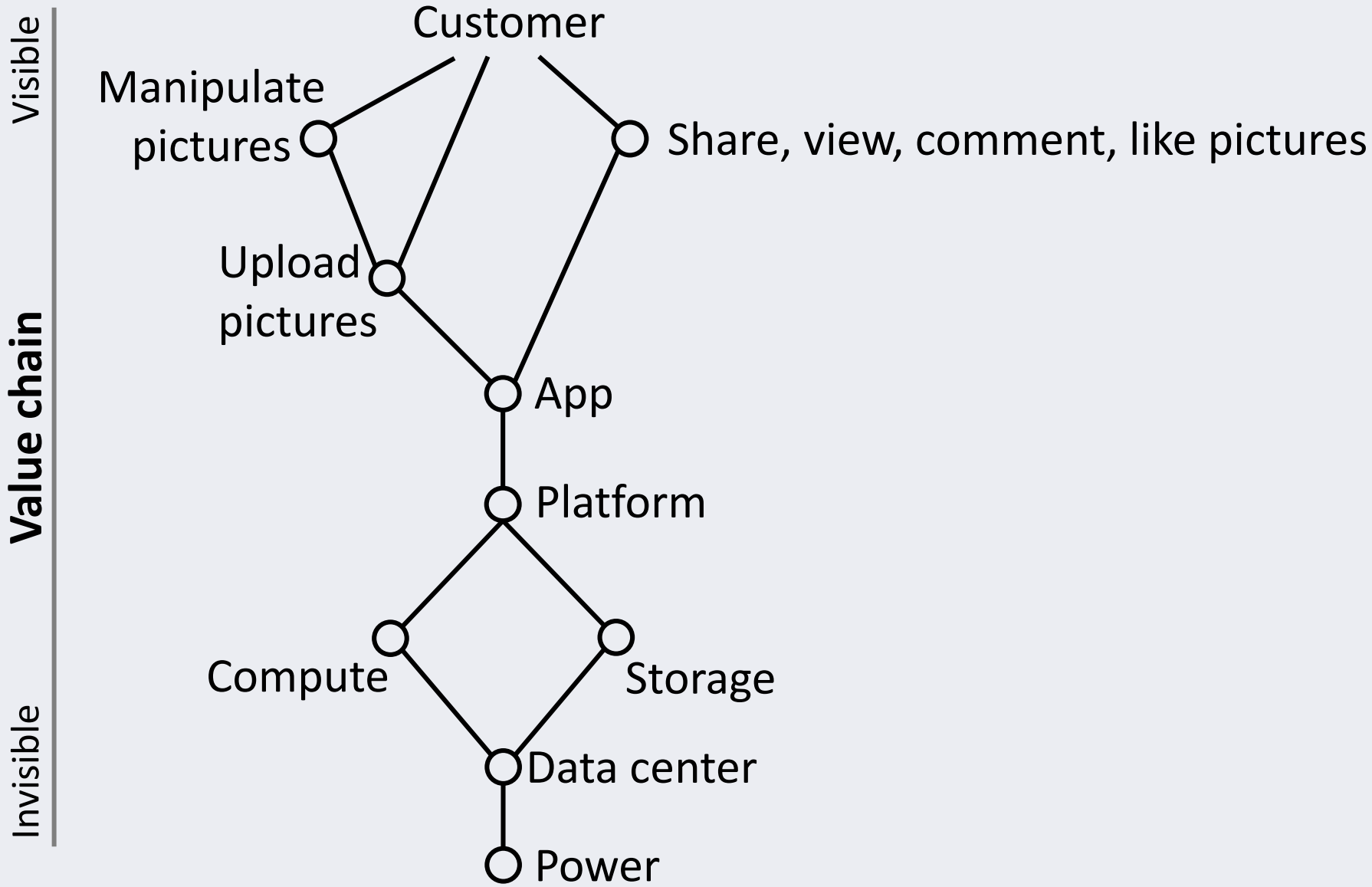
- Serverless as general pattern
- Specifically using experience we have with using the Serverless Framework on AWS

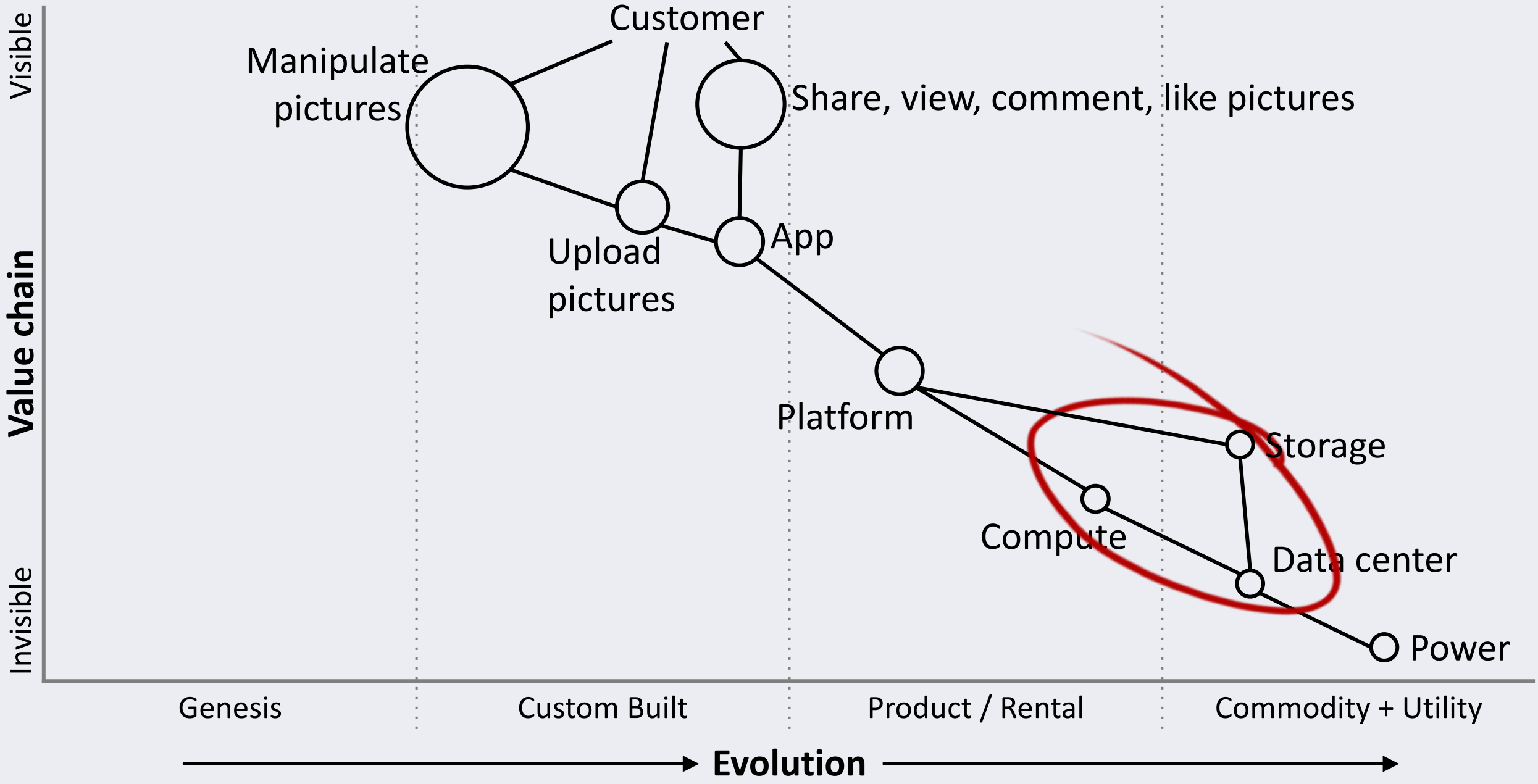
Where's the audience at, and our premise

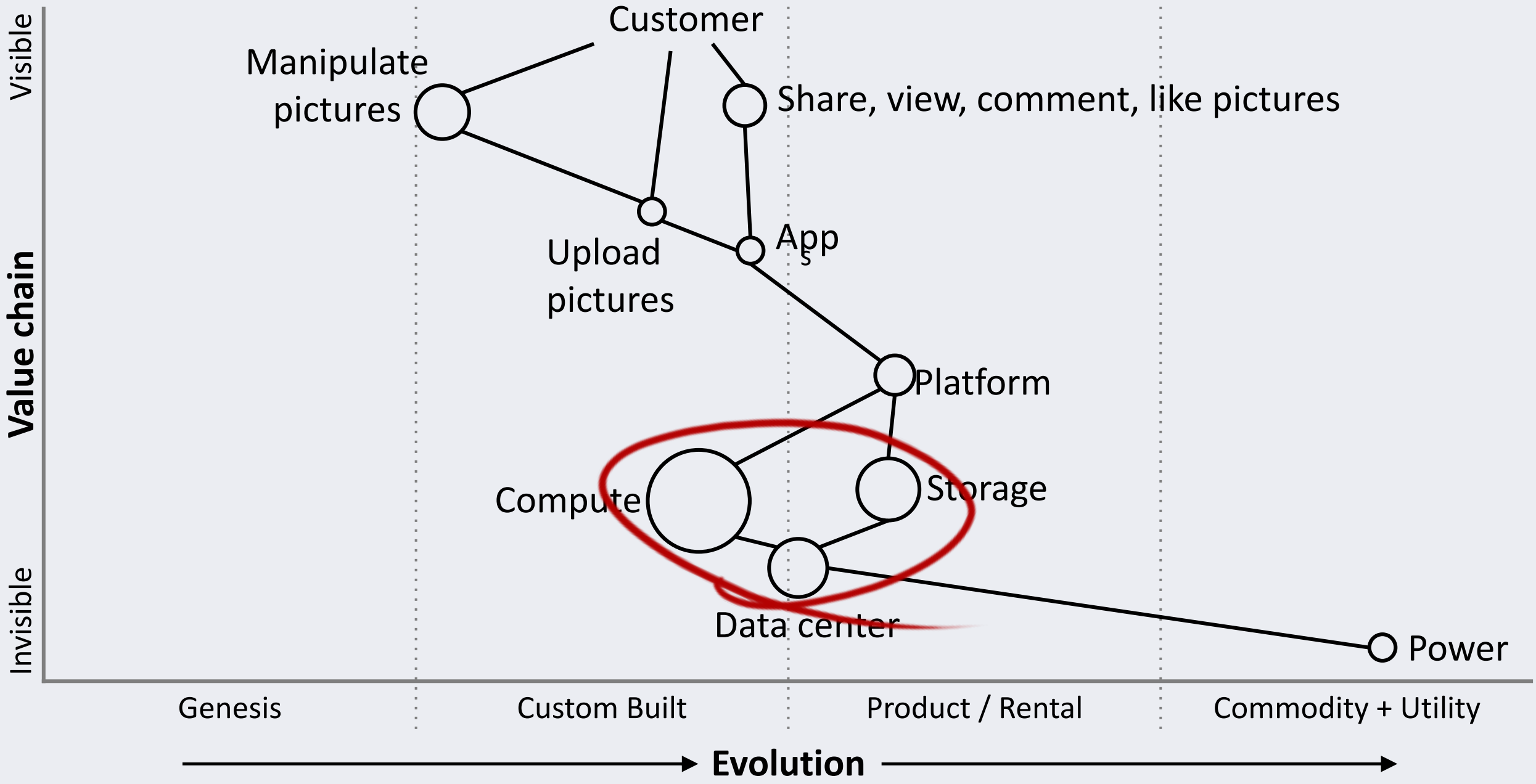
- Maturity framework of DevOps/cloud adoption
  - Show framework, ask people where they are, we will show how serverless gets you to most mature stage
- Makes it easier for you to do all the things you should have done before
  - Serverless democratizes the DevOps movement
  - Serverless is the pinnacle of the DevOps movement
  - We think serverless is the best opportunity for adopting DevOps best practices at your organization

Serverless is a commodity

- Lets you move yourself up the value chain
- Wardley graph included here
  - (revisit at the end when discussing adoption paths)
    - Serverless is focusing on the top of the value chain
    - Lift and shift is focusing on the bottom of the value chain


Visible
Value chain
Invisible
Customer
Manipulate pictures
Share, view, comment, like pictures
Upload pictures
App
Platform
Compute
Storage
Data center
Power


Visible
Value chain
Invisible
Customer
Manipulate pictures
Share, view, comment, like pictures
Upload pictures
App
Platform
Storage
Compute
Data center
Power
Genesis
Custom Built
Product / Rental
Commodity + Utility
Evolution


Visible
Value chain
Invisible
Customer
Manipulate pictures
Share, view, comment, like pictures
Upload pictures
App
Platform
Compute
Storage
Data center
Power
Genesis
Custom Built
Product / Rental
Commodity + Utility
Evolution

DevOps best practices

- Infrastructure as code
- CI/CD for increased speed and stability
- Monoliths to microservices to functions
- Small teams, highly cohesive and loosely coupled
- Only pay for what you use
- Polyglot – the right language/framework for the job

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 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

Where it's already started