Quantitative cost/benefit analysis of Infrastructure choices for computation

Ben Payne

2016

Take-away: armed with quantitative cost/benefit analysis models

- Summary of content:
  - Simple quantitative cost/benefit analysis model evaluating compute choices based on *cost per solution* for **a single recurring analytic**.
  - *Model 1*: Comparison of platforms: “pay-as-you-go” vs “commodity” vs “tailored”
  - *Action*: durability of heavy hitters should be investigated; migrate or optimize
  - Simplistic quantitative cost/benefit analysis model evaluating compute choices based on *total cost of ownership* for **an existing workload**.
  - *Model 2*: Multi-tenancy on existing infrastructure
  - *Action*: determine whether to diversify architecture
- *Desired outcome*: You are armed with quantitative models for decision making

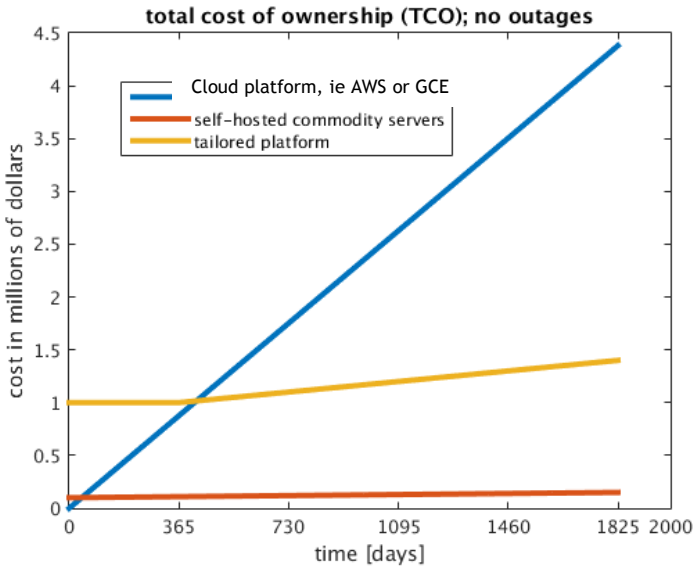
[Model 1] Comparison of platforms, single analytic: "pay-as-you-go" vs "commodity" vs "tailored"

- **AWS**: assume zero capital cost and zero operations and maintenance cost. Fiscal cost is in number of hours used; I assume a rate of $100/hour. Time-to-market is the advantage; assume 1 day for acquisition and creating the analytic. Time-to-solution is assumed to be 100 minutes

- **self-hosted commodity** platform: capital cost is $100,000 (one-time investment), and the operations and maintenance is $10,000 per year (an on-going cost). Time-to-market is 5 days; time-to-solution is 60 minutes.

- **tailored architecture** is costly — $1,000,000 for capital, and $100,000 per year for O&M. Advantage is a tiny time-to-solution, 1 minute, but the time-to-market is 1 year. Time-to-market includes NRE and acquisition.

Total cost of ownership



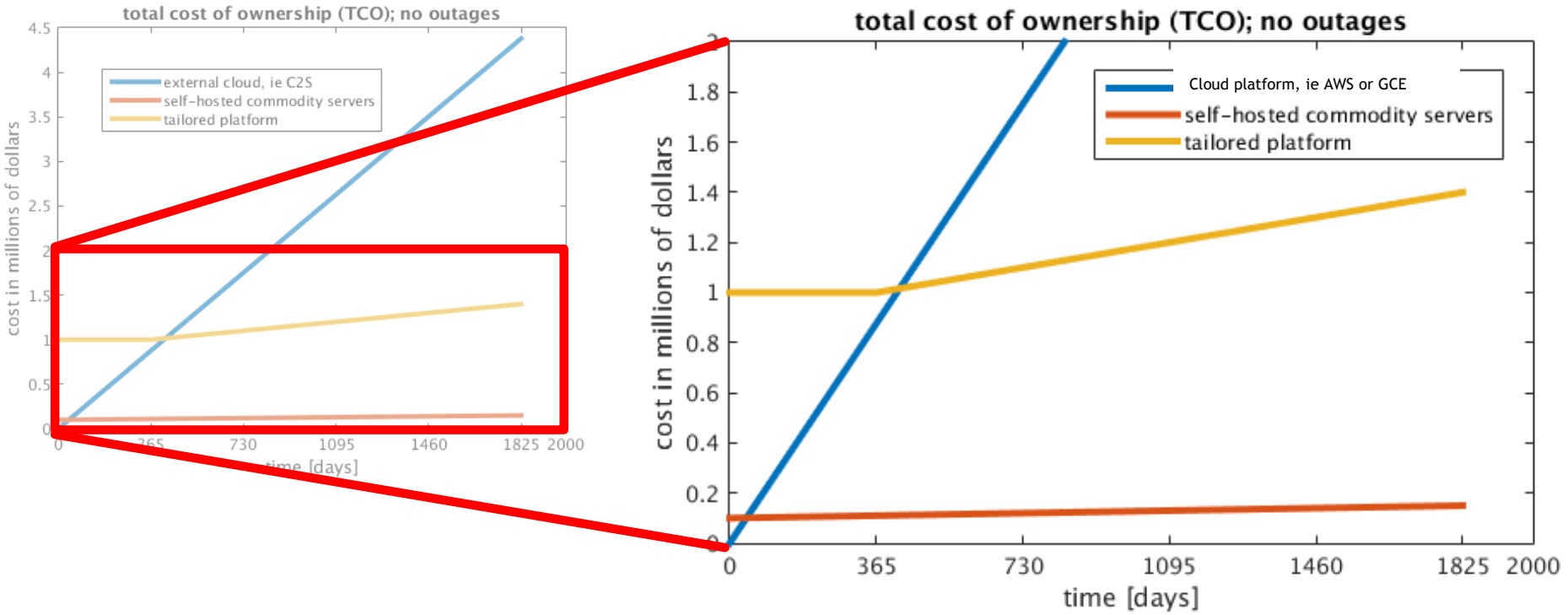
(Lower is better)

(Assume a single analytic)

Observations:

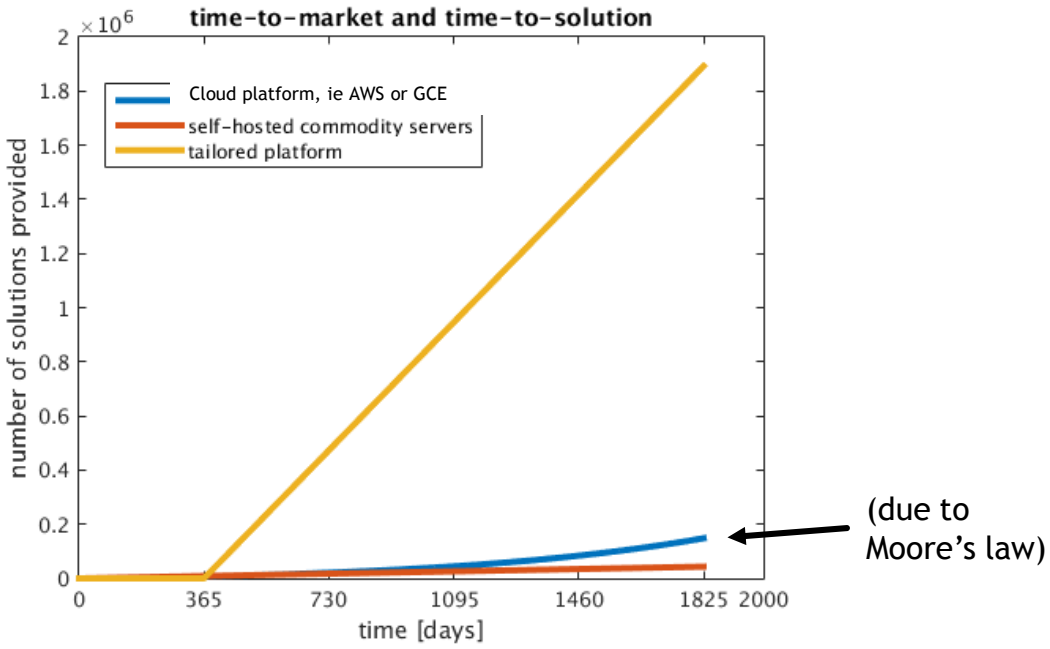
- No capital investment means initial cost is lowest
- Tailored architecture costs more than commodity
- Paying by the hour gets expensive

Total cost of ownership



For the current parameters, pay-by-the-hour is not as cost effective;
you do get access to compute faster

Time-to-market and time-to-solution



(due to Moore's law)

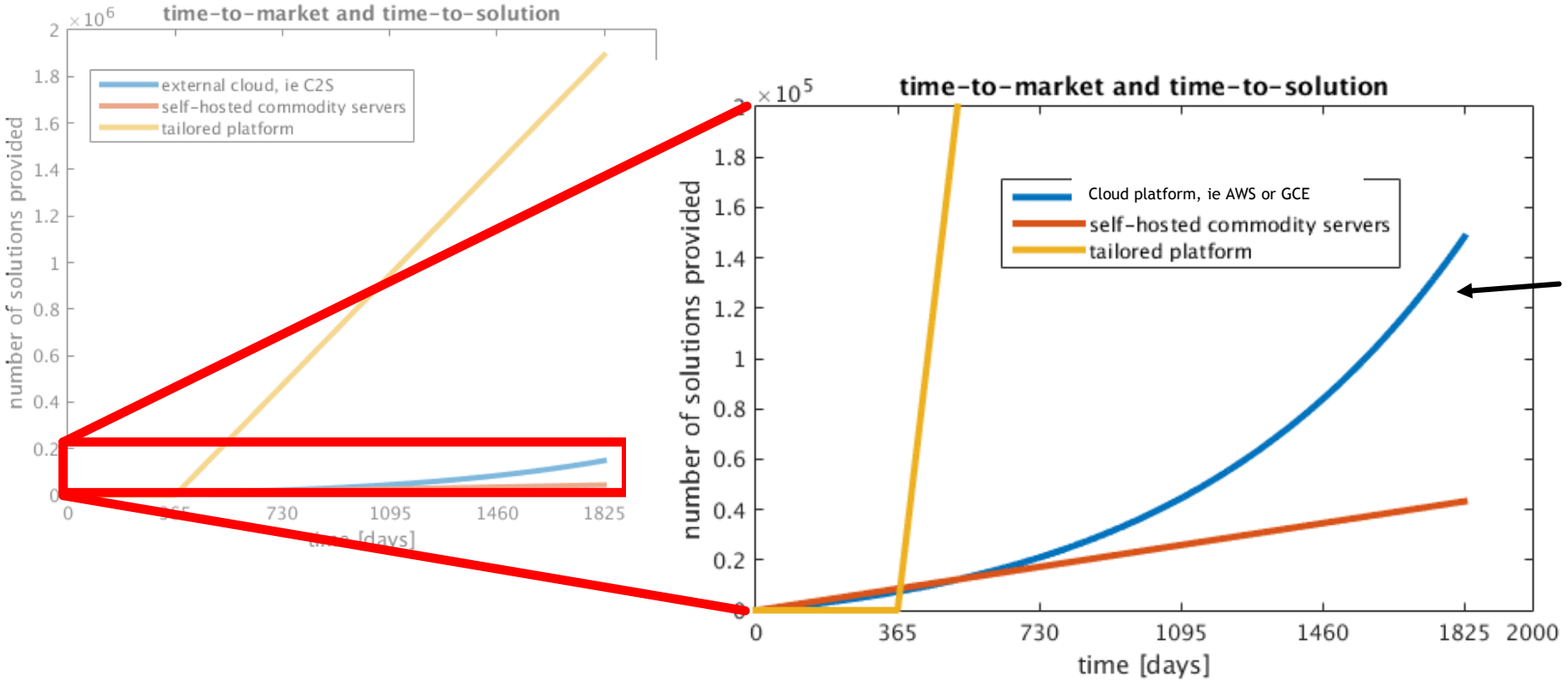
- The tailored architecture provides more solutions over lifespan

(Higher is better)

System availability:

- External cloud: 99.9%
- Self-hosted cloud: 99%
- Tailored architecture: 90%

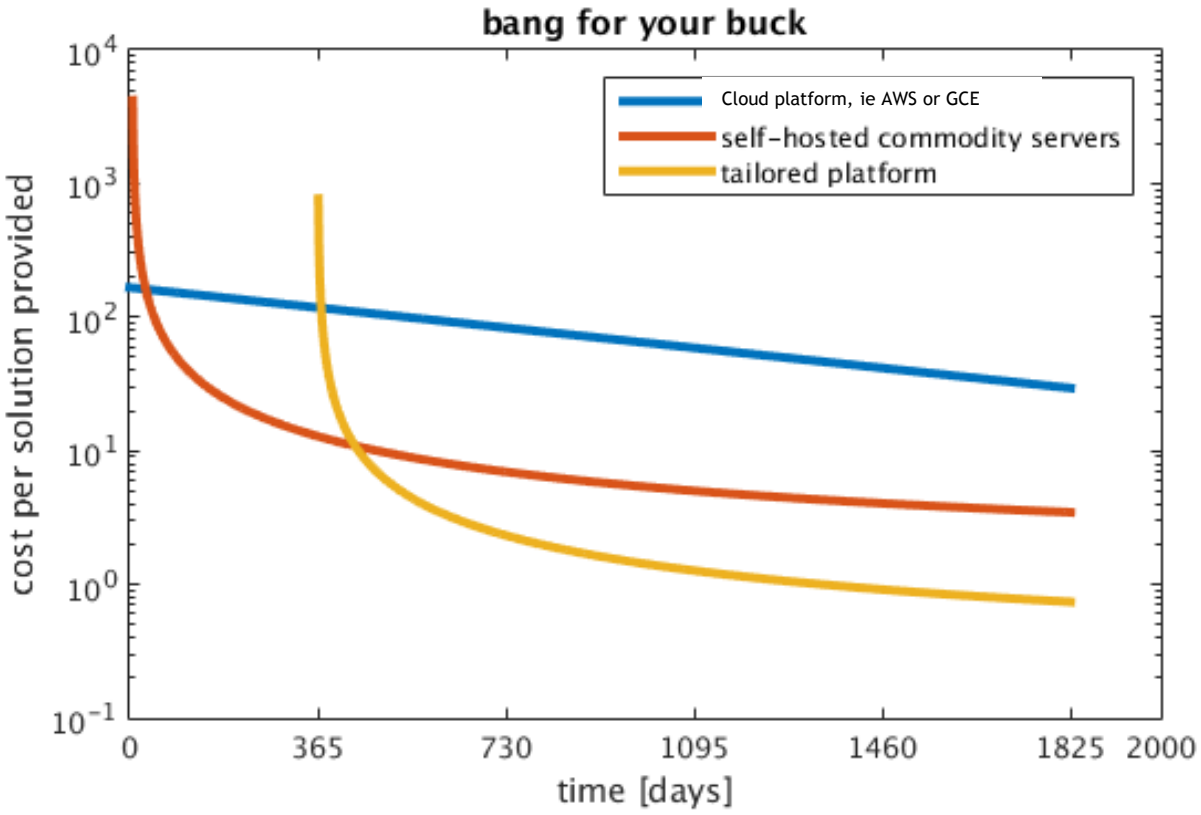
Time-to-market and time-to-solution



(due to Moore's law)

For the current parameters, ratio of time spent in NRE and acquisition to time-to-solution significantly impacts throughput

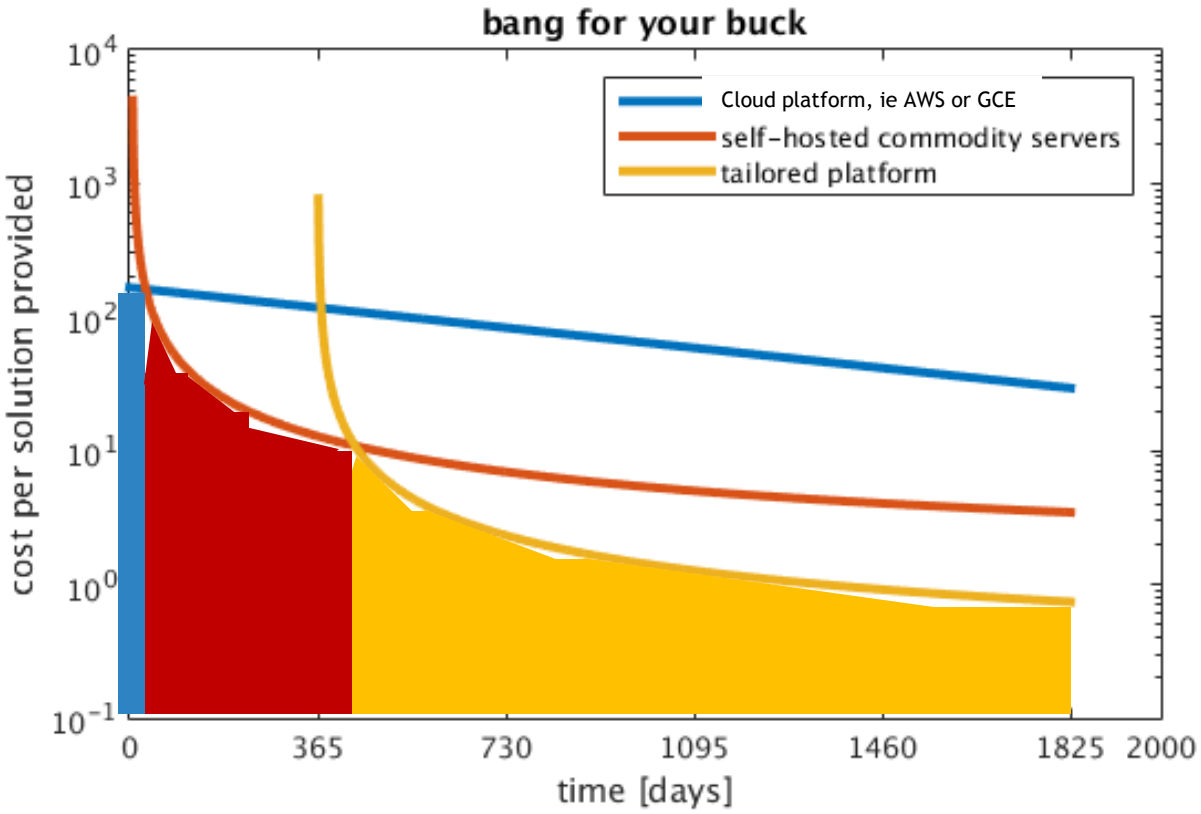
Bang-for-your-buck



(Lower is better)

Over the lifespan of the system, tailored IaaS is better ROI

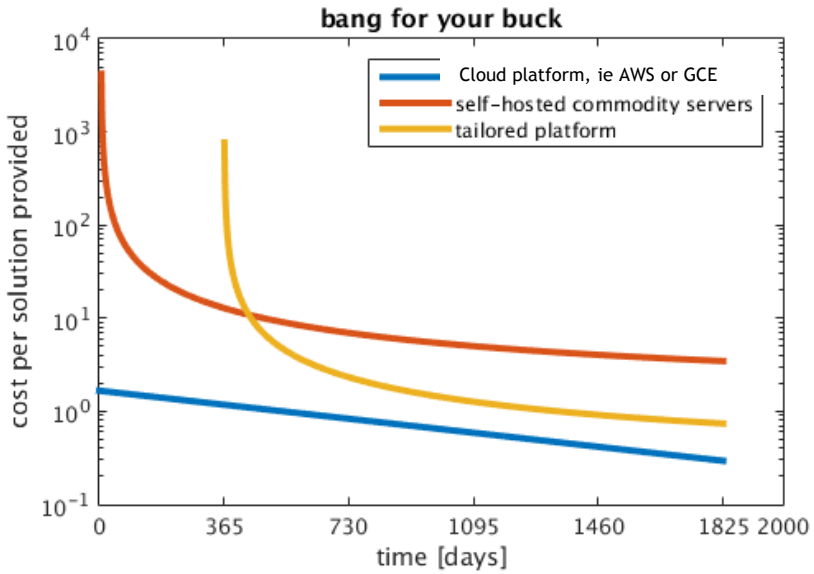
Bang-for-your-buck (cost per solution)



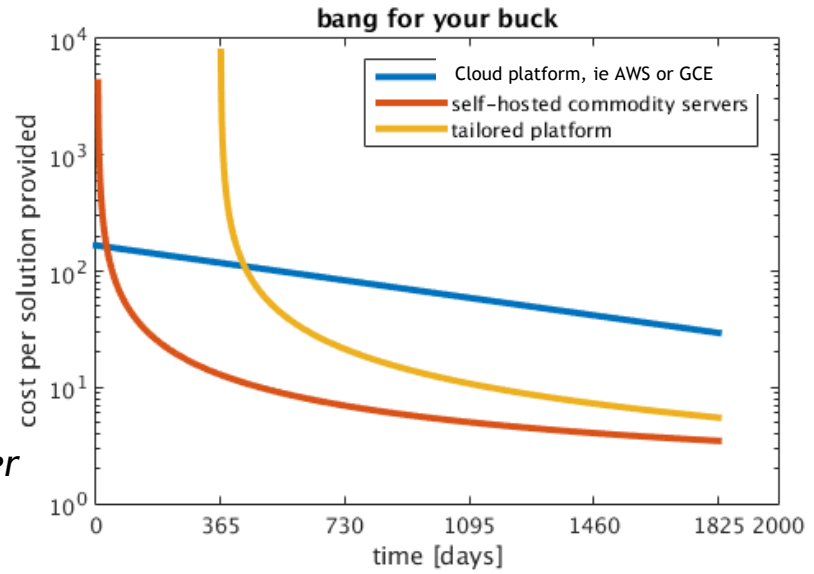
(Lower is better)

"best" choice of compute platform depends on durability

Sensitivity analysis for cost per solution



Lower is better



External cloud is best when cost is $1/hour rather than $100/hour

Self-hosted commodity is best when capital cost of tailored is $10M rather than $1M

End of “single analytic” model
Start “multi-tenancy” model

[Model 2] Multi-tenancy on existing compute infrastructure

- Suppose existing platform environment is homogenous
- Suppose we will get money to spend on compute for next fiscal period

- Should that money be invested on more of the same, or spend the money on a novel architecture which yields a heterogeneous compute portfolio?

- Assume fixed functionality (same analytics this year and next)
- Assume fixed amount of work this fiscal period and next

Input Parameters

Existing infrastructure

- number of existing compute units for architecture A: 5
- capital cost per compute unit for architecture A: $20
- O&M cost per compute unit for architecture A: $5
- Money to invest in next fiscal period: $100
- For each analytic, CPU hours for architecture A: {40, 100, 10, 5, 2}

Existing workload

Input Parameters

Existing infrastructure

- number of existing compute units for architecture A: 5

- capital cost per compute unit for architecture A: $20
- capital cost per compute unit for architecture B: $50
- O&M cost per compute unit for architecture A: $5
- O&M cost per compute unit for architecture B: $8
- Money to invest in next fiscal period: $100

Existing workload

- For each analytic, CPU hours for architecture A: {40, 100, 10, 5, 2}
- For each analytic, CPU hours for architecture B: {30, 10, 8, 4, 1}

Projected workload

Result of analysis: current fiscal period

For each analytic, CPU hours for architecture A: {40, 100, 10, 5, 2}

Existing workload

Existing infrastructure

current_tts =

31.4000

Result of analysis: next fiscal period

For each analytic, CPU hours for architecture A: {40, 100, 10, 5, 2}

Existing workload

current_tts =

31.4000

tts_homogeneous =

19.6250

Existing infrastructure

+ spend $100 on capital and O&M for additional “A”

Result of analysis: next fiscal period

current_tts =

31.4000

min_tts_heterogeneous =

21.4000

Existing infrastructure

+ spend $100 on capital and O&M for additional “B”

Result of analysis: next fiscal period

For each analytic, CPU hours for architecture A: {40, 100, 10, 5, 2}

For each analytic, CPU hours for architecture B: {30, 10, 8, 4, 1}

current_tts =

31.4000

Existing infrastructure

min_tts_heterogeneous =

21.4000

+ spend $100 on capital and O&M for additional “B”

Best analytic distribution, platform A =

40 0 10 5 2

best analytic distribution, platform B =

0 10 0 0 0

Result: maintaining homogeneous architecture yields better throughput

current_tts =

31.4000

tts_homogeneous =

19.6250

min_tts_heterogeneous =

21.4000

Best analytic distribution, platform A =

{40 0 10 5 2}

best analytic distribution, platform B =

{0 10 0 0 0}

- For these parameters, change of architecture doesn't outweigh the cost of the acquisition + O&M
- Even though architecture B is always faster for every analytic, the overall mission throughput is lower for the amount of money spent. Therefore, **stick with homogeneous architecture**

Result: different initial conditions yield opposite outcome

capital cost per compute unit for architecture A: $20

current_tts =

31.4000

tts_homogeneous =

19.6250

min_tts_heterogeneous =

21.4000

Best analytic distribution, platform A =

{40 0 10 5 2}

best analytic distribution, platform B =

{0 10 0 0 0}

*capital cost per compute unit for architecture A: **$30***

current_tts =

31.4000

tts_homogeneous =

22.4286

min_tts_heterogeneous =

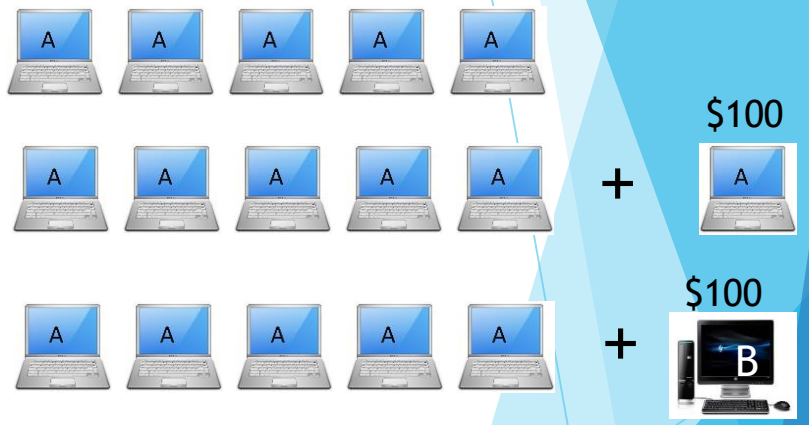
21.4000

Best analytic distribution, platform A =

{40 0 10 5 2}

Best analytic distribution, platform B =

{0 10 0 0 0}



Methodology applies to variety of situations

- I have a homogenous commodity IaaS and want to know whether to buy invest in a novel architecture
- I have tailored IaaS and want to know whether to invest in commodity IaaS
- I have both tailored and commodity IaaS and want to know what future distribution of funds maximizes throughput
- Any platform, any set of analytics

- Need capital cost, O&M cost, and job runtimes per platform
- Need existing infrastructure, job list, and amount of money to be invested
- Output: which platform acquisition maximizes mission throughput

End of models discussion
Start of observations

Analysis of production compute environment

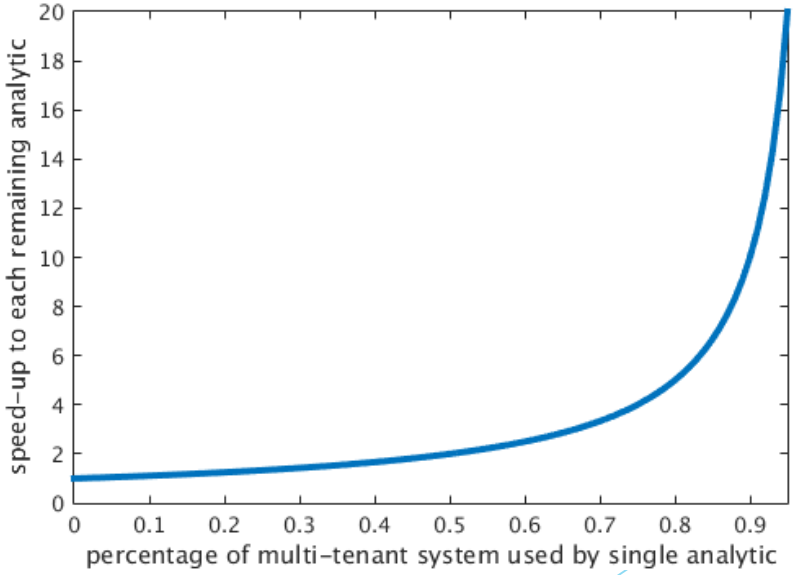
- Are there “heavy hitters/resource hogs,” or does each analytic use the same amount of the compute platform?
- Are the “heavy hitters” durable? Over what time scale?

If distribution of utilization per job is *not* uniform → exploit this

- Durable, large scale analytics should get more attention
  - Optimize
  - Move to tailored infrastructure
- Quantize the cost/benefit of moving from commodity IaaS to something else
  - Many analytics, compare RoI for a single investment period (ie fiscal year)
  - Single analytic, compare RoI for multiple platforms over lifespan (ie 5 years)

Impact to remaining analytics when heavy hitter is migrated off platform

- Suppose a compute platform has hundreds of unique analytics run per day
- Suppose 1 of the analytics consumes 40% of the compute resources
- If that one analytic is moved off the platform, each of the remaining analytics can speed up by (1/(1-0.4)) = 1.67



Evaluate cost/benefit of architecture choices

- collect **measurements** from current platforms (time-to-solution for every analytic) over some time period
- then project impact on time-to-solution of augmenting existing architecture with tailored solution
- don't need to re-write every analytic; assume improvement the heaviest analytic by some speedup factor >>1
- Need to know cost per rack of compute, both capital and O&M

Backup content

- Criticisms of numerical models
- Criticisms of single analytic model
- Criticisms of multi-tenant model

Criticisms of numerical models

- Some analytics aren't feasible due to scale of data

  [For example, a data set so large it doesn't fit in RAM of an HPC]

  - In that case, the data could be partially loaded, then iterated. The time-to-solution would be large but quantifiable

- Jobs vary due to input data, configuration parameters
  - Correct. Either assume ensemble averaging over large number of jobs, or account for the distribution of runtimes.

More assumptions that could be added

- Assume software application evolves
- Assume rate of software application evolution is proportional to the size of the workforce
- Assume system administrator positions can be exchanged for software developer positions

  → Given some number of positions and some budget, how to maximize throughput?

Unquantized aspects not captured in numerical models

- Unanticipated use cases for platforms
- Novel analytics

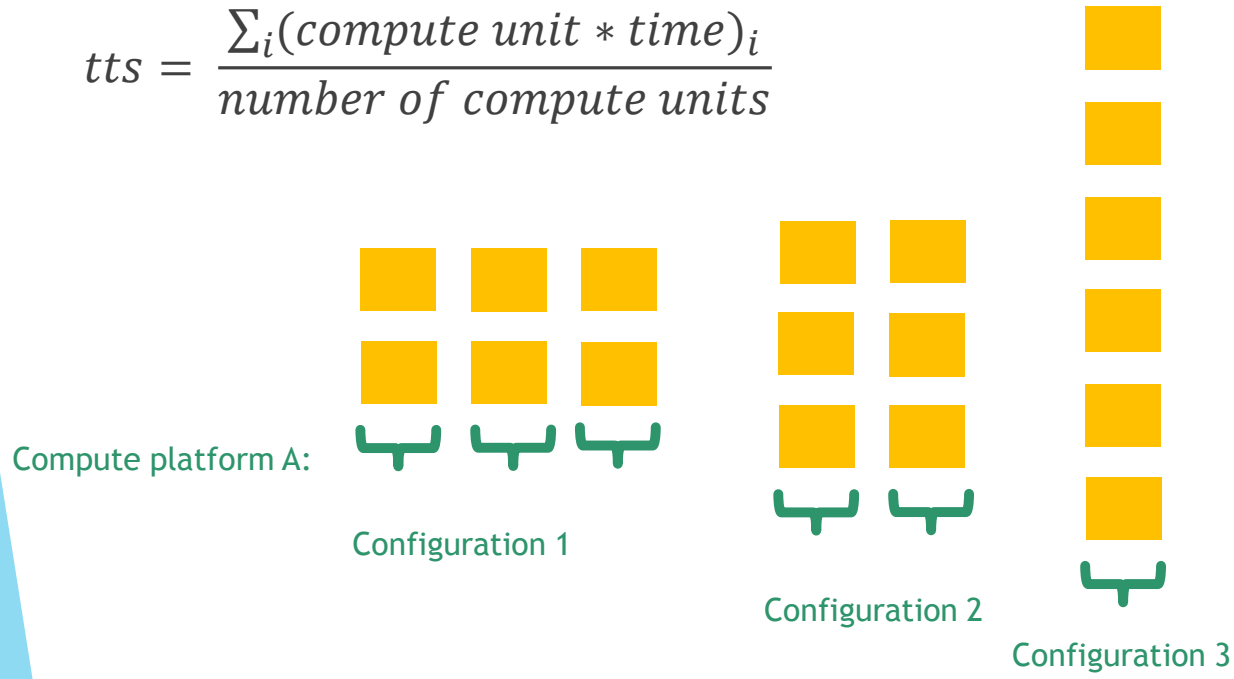
Notes on Model 1: single analytic

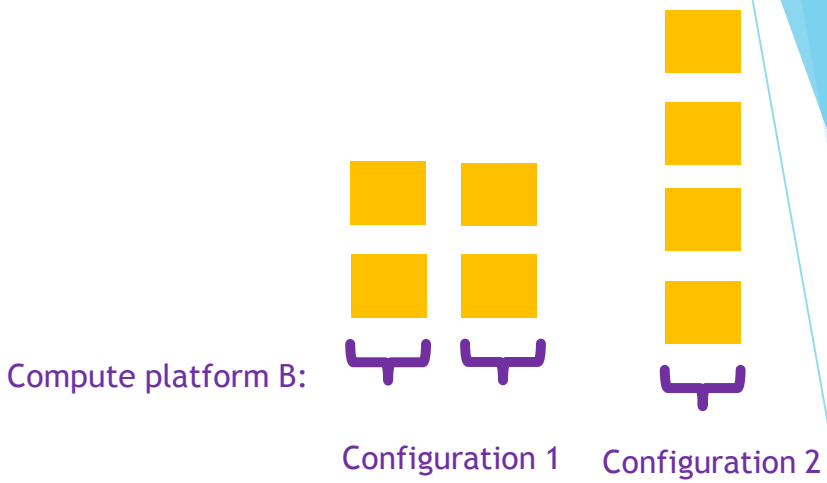
Potential Modifications to single analytic

- Rather than assume time=0 is the starting point for all systems, a more realistic portfolio has different start times for different platforms.
- Rather that forcing to a single platform, what's the benefit of augmenting platform A with surge capacity on plaform B?
- A single platform typically runs a multiple analytics.
- A lifespan of 5 years neglects the typical hardware refresh at year 2 or 3.

Attack the assumptions: space-time trade-off

$$tts = \frac{\sum_i (compute\ unit * time)_i}{number\ of\ compute\ units}$$





- Space-time trade-off of real applications is not linear; see Amdahl's law
- Scaling depends on how much data is exchanged; synchronous communication

Attack the assumptions: space-time trade-off

- Each job has a scaling curve
- A scheduler searches for the layout of jobs on N compute resources which minimizes time-to-solution
- Scheduler output would yields the time-to-solution per platform

Consequence of choosing sub-optimally

- In the 0 to 1 month window, optimal choice is *pay-by-the-hour*: $74k for 444 solutions
  - If you go with *commodity* or *tailored IaaS*, then you are paying money ($100k and $1,000k, respectively) and not getting many results (618 and 0, respectively)
- In the 1 month to 1 year window, optimal choice is *commodity IaaS*: $111.9k for 10,290 solution
  - If you go with *pay-by-the-hour*, then you are paying more money ($1,051k) and getting less mission throughput (9520 solutions)
  - If you go with *tailored IaaS*, then you are paying money ($1,020k) and not getting results (94,610 solutions)
- In the 1 year to End-of-Life window, optimal choice is *tailored IaaS*: $1,400k for 1,892,000 solutions
  - If you go with *pay-by-the-hour* or *commodity IaaS*, then you are paying more money and getting less mission throughput

Consequence of choosing sub-optimally

Days	Platform	Money spent	Solution count
31	Pay-by-the-hour	$74,400	444
	Commodity IaaS	$100,700	618
	Tailored IaaS	$1,000,000	0
438	Pay-by-the-hour	$1,051,000	9520
	Commodity IaaS	$111,900	10,290
	Tailored IaaS	$1,020,000	94,610
1825 (5 years)	Pay-by-the-hour	$4,380,000	148,300
	Commodity IaaS	$149,900	43,200
	Tailored IaaS	$1,400,000	1,892,000

Best bang-for-your-buck

→ Commodity IaaS yields similar solution count for 10x less cost than C2S

→ Tailored IaaS yields 12x more solutions and costs 3x less than C2S

Notes on model 2: multi-tenancy

Potential Modifications to multi-tenancy

- Include a switching cost
- Include a switching time – might not be able to use platform B on day 1.
- Rather than “existing A + new A” or “existing A plus new B”, enable exploration of “existing A + some more new A + some new B”

Criticisms of the multi-tenant model

- For the multi-tenant, single fiscal period model, comparing analytics written by novices to optimized analytics written by experts is not fair
  - The point is to find which analytics should be optimized or moved to a tailored architecture
- For the multi-tenant, single fiscal period model, the workload doesn't account for unknown analytics
  - I don't include unquantized aspects in this numerical model
- For the multi-tenant, single fiscal period model, the workload doesn't reflect possible changes in distribution of analytics
  - Correct. If you can provide a projection of the change, that can be used as an input.

Attack the assumptions: known tasks; known amount of work

- Projection is needed for the fiscal period being modeled
- We have data for what was done in the previous period
- Extrapolate historical data using a scaling factor