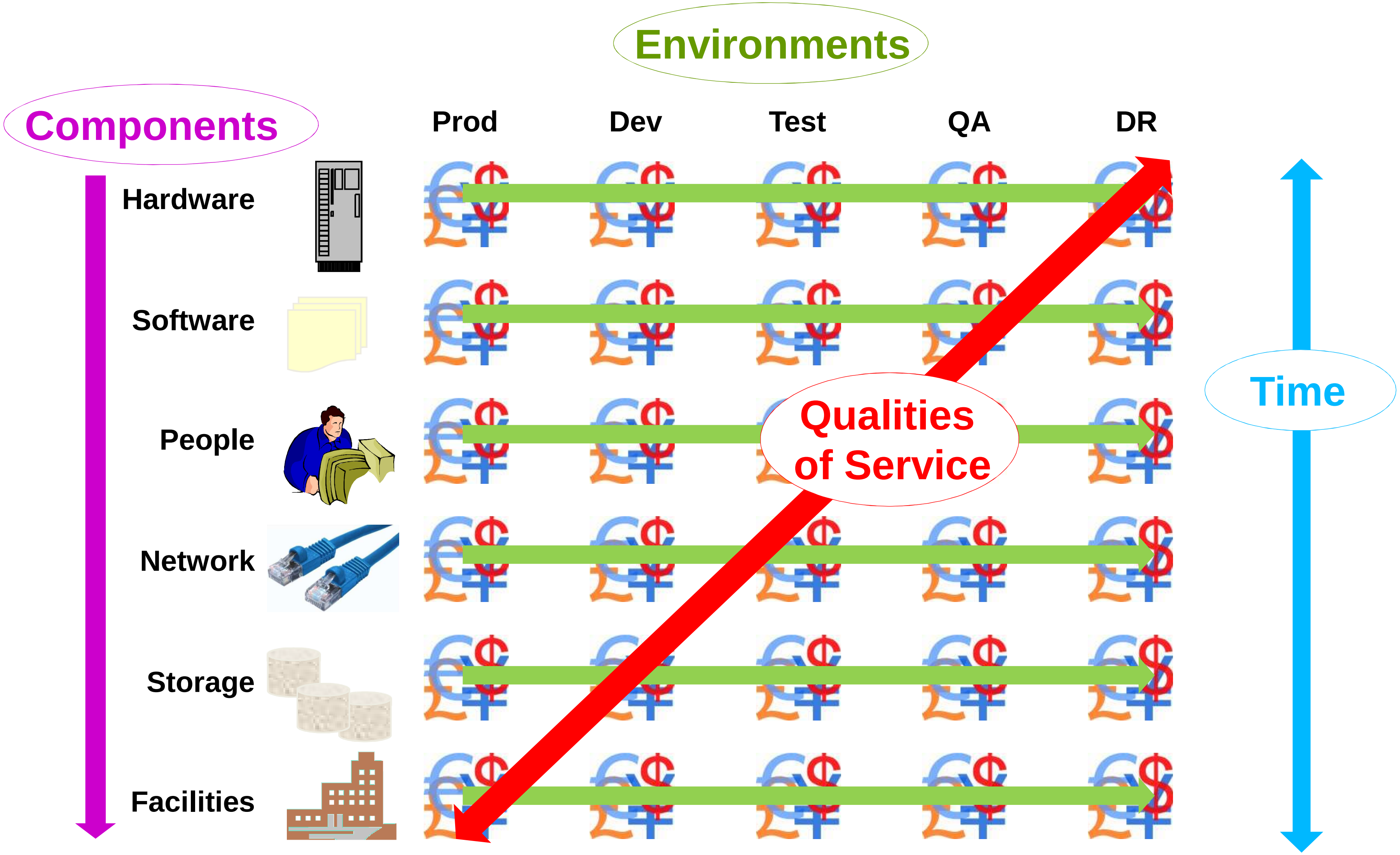
# Real Cost of Ownership Lessons from Customer Engagements

Innes Read, IBM

# Understanding TCO means understanding multiple dimensions of cost



# Most mainframe customers are very focused on cost and efficiency

- Most mainframe customers are exploiting the obvious
  – Specialty processors
  – Linux consolidation if Linux fits your organization
  – Capping to control 4-hr rolling average extremes
  – Spreading workload throughout the day where possible
  – Specific pricing metrics and LPARs for certain work
  – etc.

- We'll touch on some of them briefly just to make sure!

- But we'll also approach this top down, as well as bottom up

# Bottoms up improvement examples...

- Use the system better
  - Flatten monthly 4-hr average usage profile if sub-cap
  - Improve unit cost of components (eg. 3rd party tools)
  - Use spare MIPS to further amortize fixed costs
  - Collocate workloads for higher performance and efficiency

- Break workload up to exploit unique pricing metrics
  - Often less efficient computationally so this is a balance
  - Likely more useful when introducing new workloads

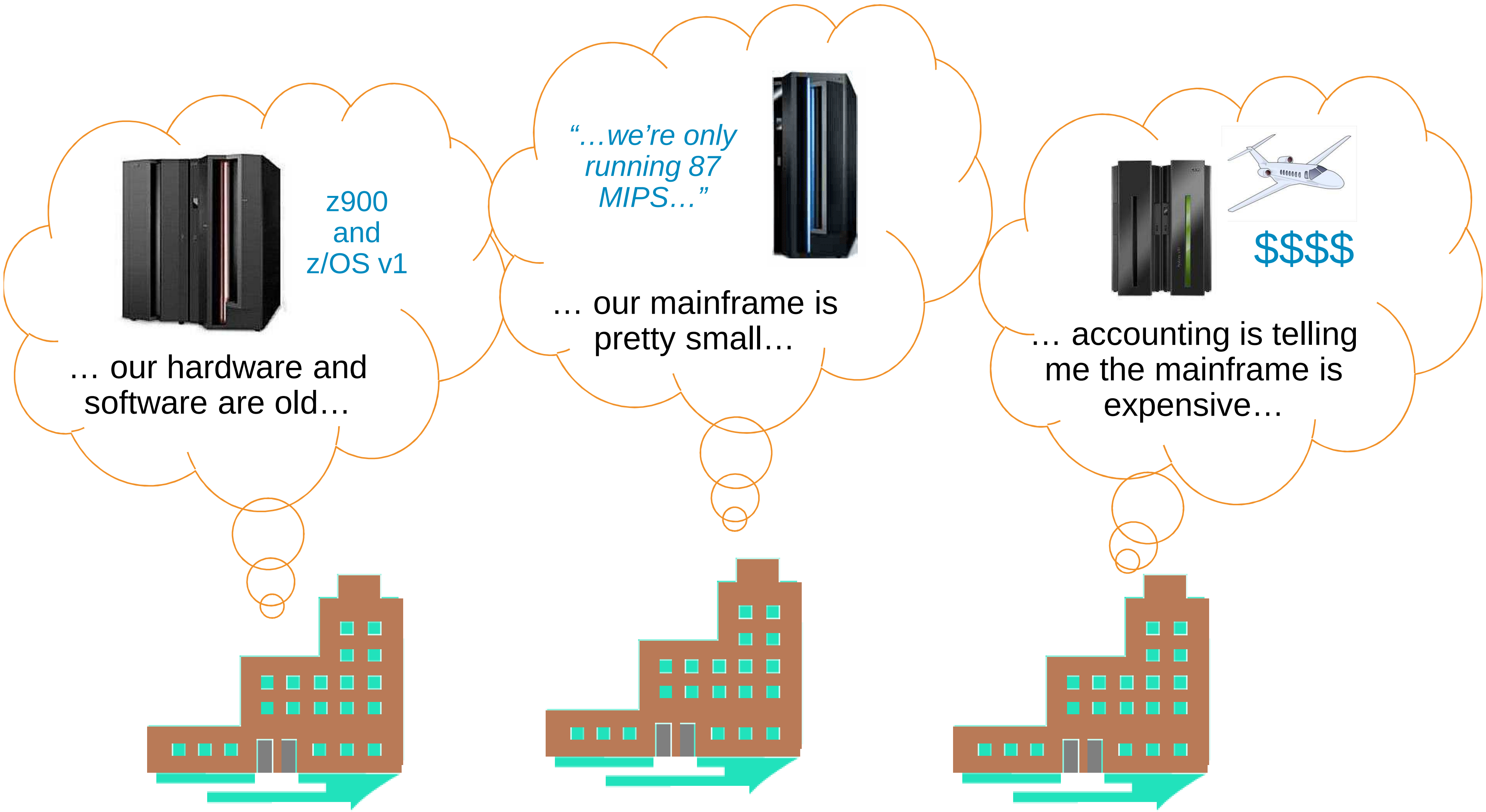
- Reduce “expensive” usage
  - This could be tuning the software stack that runs in the peak hours
  - Or it could be removing applications from the peak hour altogether
  - Assuming there is another time, or perhaps somewhere else, that can do the same work cheaper (or work is not actually required!)

- ...

# ... vs. top down opportunities

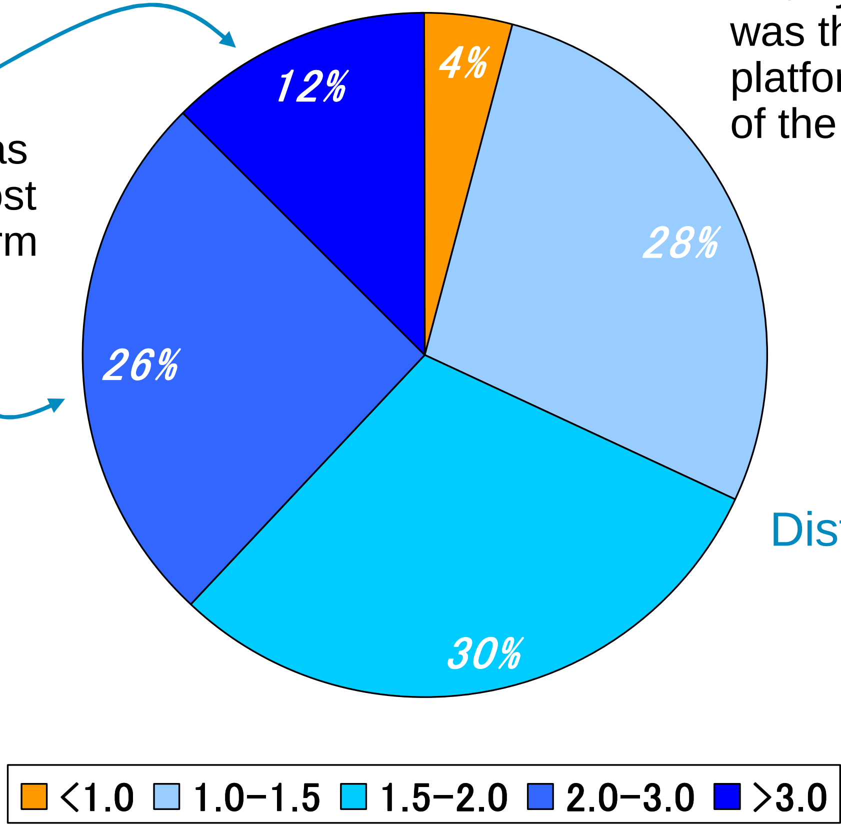
- The mainframe is a long term asset for most organizations
- Most of the cost of a mainframe is already in the base
  - ie. sunk or associated with major core business applications
- The incremental cost of growth on a mainframe is much lower than the average cost of existing applications
  - Incredible economy of scale in software, labor, hardware
  - IBM continues to invest in new technology to enable higher business value workloads
- The value of centralized computing continues to grow
  - Shared resources, common infrastructure – aka cloud
  - Big data and analytics, potentially integrated in real-time with OLTP
  - Seamless integrated views of customers and the business
- Existing IT rules of thumb may be driving inefficient deployments from a technology and cost perspective

# Some mainframe clients are tempted to move workloads off the mainframe, allegedly to save money



# But Eagle team data shows that in 96% of mainframe rehosting cases, clients ultimately end up spending *more* for an offload

In 38% of cases, the distributed platform was 2 or more times the cost of the System z platform




In only 4% of Eagle TCO studies was the cost of the distributed platform cheaper than the cost of the System z platform

Cost ratio – Distributed to System z

Sampling of 97 Eagle team TCO studies from 2007 - 2011

# Why are rehosting costs underestimated?

From HP's "Mainframe Alternative Sizing" guide, published in 2012…

| MIPS Level | z196 Models | Actual MIPS | z10 EC Models | z10 Actual MIPS | z10 BC Models | z10 BC Actual MIPS | z114 Models | z114 Actual MIPS | HP Cores Estimate | Total HP equivalent MIPS |
|---|---|---|---|---|---|---|---|---|---|---|
| **1,000** | 2817-701 | 1,202 | 2097-701 | 889 | 2098-Z02 | 1250 | 2818-Z01 | 782 | 2 | 866 |
| **2,000** | 2817-702 | 2,272 | 2097-702 | 1,667 | 2098-Z03 | 1784 | 2818-Z03 | 2026 | 5 | 1,860 |
| **3,000** | 2817-703 | 3,311 | 2097-704 | 3,114 | 2098-Z05 | 2760 | 2818-Z05 | 3139 | 8 | 3,021 |

Can a 2-chip, quad-core x86-based
Blade server really replace 3,000+ MIPS?

- Simple core comparisons are inherently inaccurate…
- Real world use cases suggest this number is off by a factor of **10-20 times**

# I have a whole pitch on the ugly truth around mainframe rehosting but in summary... it's ugly! A recent example that actually finished...

3x HP DL580 (2ch/20co)
Production / Dev / Test
(2011 x86 technology)


z800 Production /
Dev / Test
(2002 mainframe technology)


***499 MIPS***
*(2.1 processors)*

***60 processors***

Despite a 9-year technology gap,
the Intel platform still required
**29x** more processors

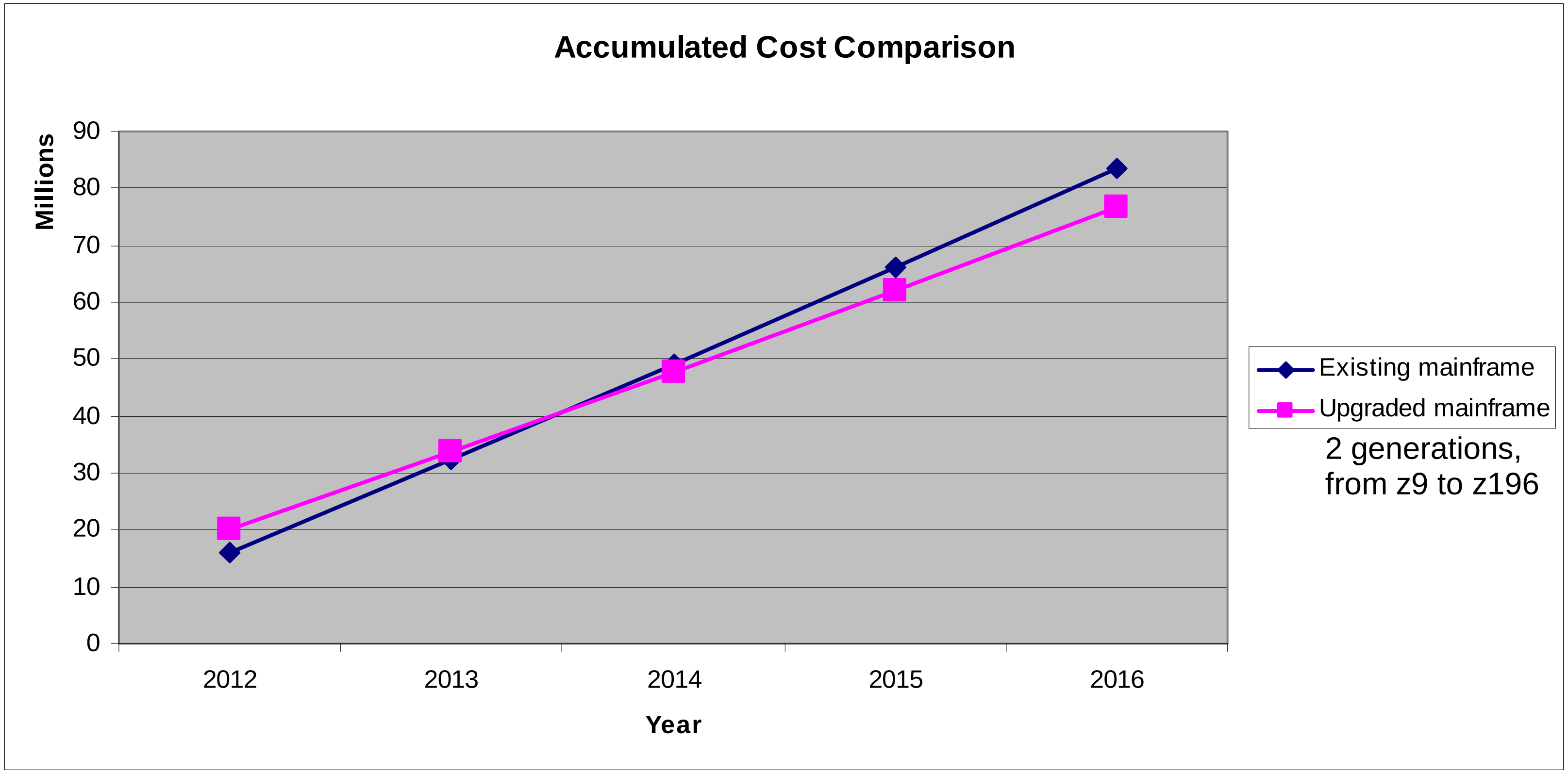
768 Performance Units per MIPS

# Lessons learned can be grouped into three broad categories

- Always compare to an optimum System z environment
- Look for not-so-obvious distributed platform costs to avoid
- Consider additional platform differences that affect cost

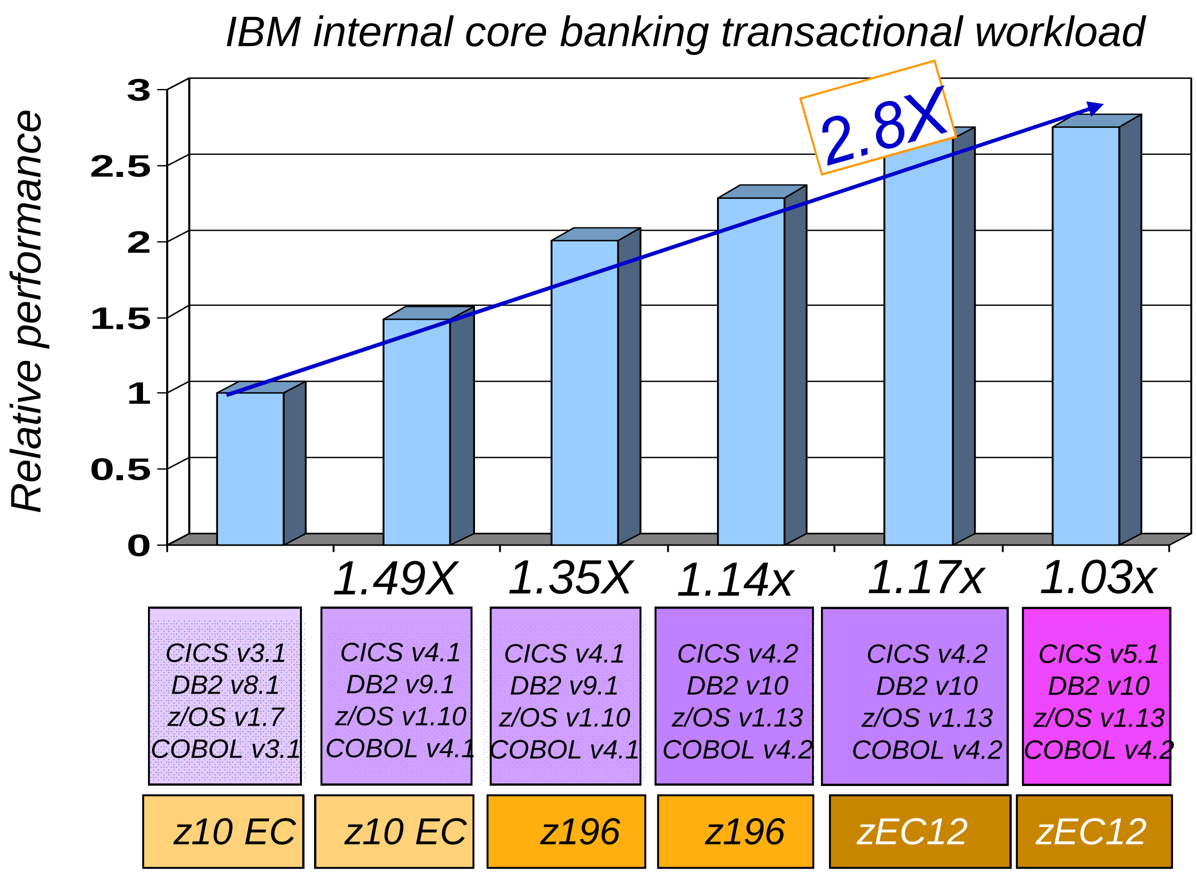
**All examples discussed are from actual Eagle Team customer studies**

# Keeping current with respect to hardware saves money




- Typical customer (European bank) hardware refresh scenario
  - 2M investment pays back >1M savings every year – most cases positive in a 3 year period
  - Savings from technology dividends and specialty processor offload
- Comparing latest technology servers to old mainframes is unfair but often done

# Currency reduces cost for software too (in fact, more than for hardware, just not aligned with a single "upgrade" event)




## Customer examples:

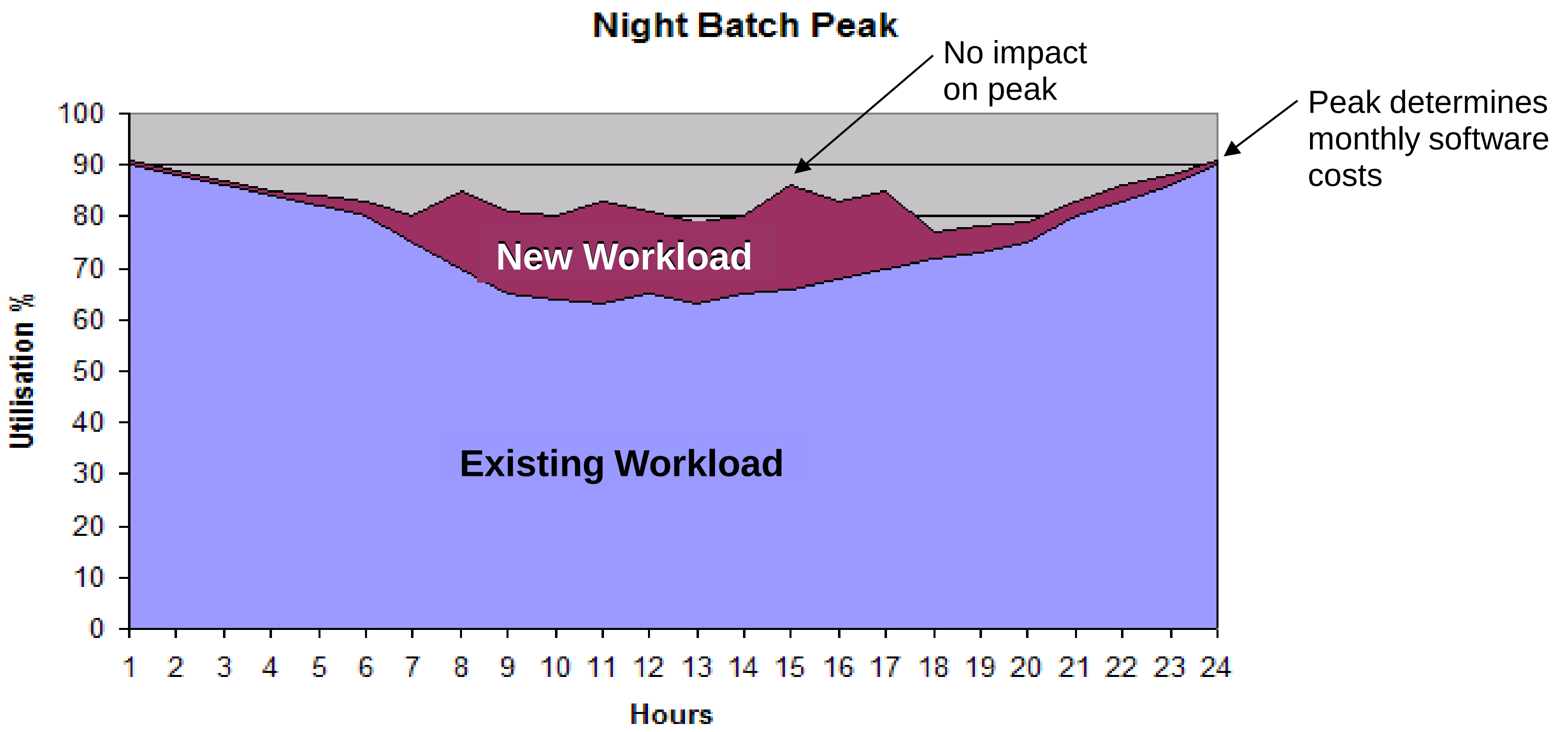
### (1) Large MEA bank

- Delayed upgrade from z/OS 1.6 because of cost concerns
- When they upgraded to z/OS 1.8
  - Reduced each LPAR's MIPS 5%
  - Monthly software cost savings paid for the upgrade almost immediately

### (2) BMW Autos

- Upgraded to DB2 10
- Realized 38% pathlength reduction for their heavy insert workload
  - Other DB2 10 users saw 5-10% CPU reduction for traditional workloads

# Take advantage of sub-capacity pricing to create free workloads




- Standard “overnight batch peak” profile – drives monthly software costs
- Hardware and software are free for new workloads using the same middleware (e.g. DB2, CICS, IMS, WAS, etc.)
- Ensure you exploit any free workload opportunities, and conversely, avoid offloading free applications!

# Save money by replacing ISV software with IBM software

## A medium-sized European financial company…

| Average Profile (BEFORE) | | |
|---|---|---|
| Weighted MIPS | | 8,800 |
| Cost Per MIPS per Year | | Profile |
| IBM Software | 1,000.00 | 24.72% |
| | | 0.00% |
| ISV Software | 1,540.00 | 38.07% |
| TOTAL SW | 2,540.00 | |

| Actuals (AFTER) | | |
|---|---|---|
| Weighted MIPS | | 8,900 |
| Cost Per MIPS per Year | | Profile |
| IBM OTC | 376.09 | 13.66% |
| IBM MLC | 1,023.77 | 37.20% |
| ISV Software | 136.09 | 4.94% |
| TOTAL SW | 1,535.95 | |

IBM software costs increased slightly…

… but ISV software costs decreased dramatically!

*Result:*
***$1,000** per MIPS*
*per year savings!*

# Replacing ISV software with IBM software is also more cost-effective than offloading

## A major global bank considered two options…

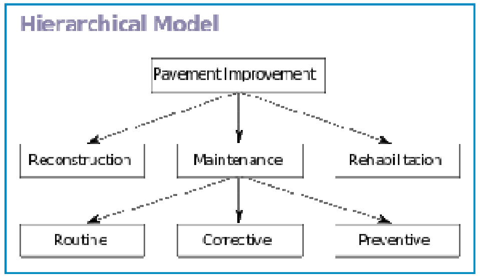
| | *Mainframe Offload* | *Move to IBM Tooling* |
|---|---|---|
| *Investment cost -> time period* | $54M -> 2 years | $3M -> 1 year |
| *Predicted annual cost savings* | $13M (from year 3) | $6M (from year 2) |
| *5 Year TCO, breakeven time* | $140M, year 7 | $101M, year 2 |
| *Assessed level of risk* | Very high | Very low |

*The choice was obvious!*

Large project,
expensive,
high risk,
distant payback

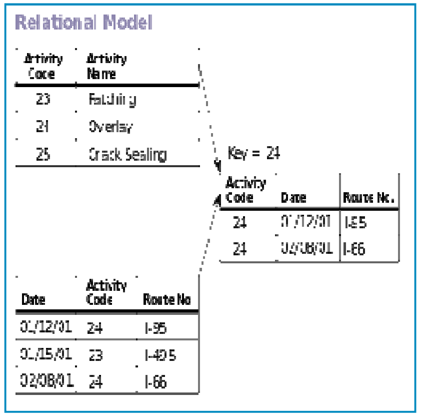
Small project,
cheaper,
lower risk,
instant payback

# Changing databases can have dramatic capacity impacts



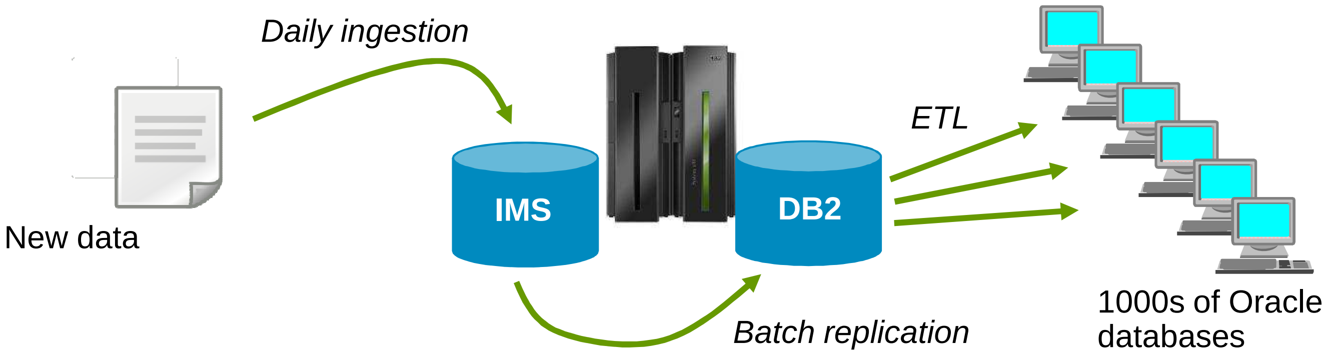

IMS is the most widely used hierarchical data store

*Converting from IMS to DB2 can result in 2-3x more MIPS used and degraded response time*




SQL databases, including DB2

## A European financial company is attempting a conversion while continuing to run the business…




*In 4 years, only 30% converted and €500M spent so far*

Source: Wikipedia

# Linux on System z consolidation usually has lower costs

***Which platform provides the lowest TCA over 3 years?***

Oracle DB workload

3 OLTP Database Workloads, each supporting 18K tps

Oracle Enterprise Edition

Oracle Real Application Cluster

3 Oracle RAC clusters
4 server nodes per cluster

12 total HP DL580 servers (192 cores)

**$13.2M** (3 yr. TCA)

3 Oracle RAC clusters
4 nodes per cluster
Each node is a Linux guest
zEC12 with 27 IFLs

**$5.7M** (3 yr. TCA)

***Half** the cost*

TCA includes hardware, software, maintenance, support and subscription.
Workload Equivalence derived from a proof-of-concept study conducted at a large Cooperative Bank.

# Lessons learned can be grouped into three broad categories

- Always compare to an optimum System z environment
- Look for not-so-obvious distributed platform costs to avoid
- Consider additional platform differences that affect cost


**All examples discussed are from actual Eagle Team customer studies**

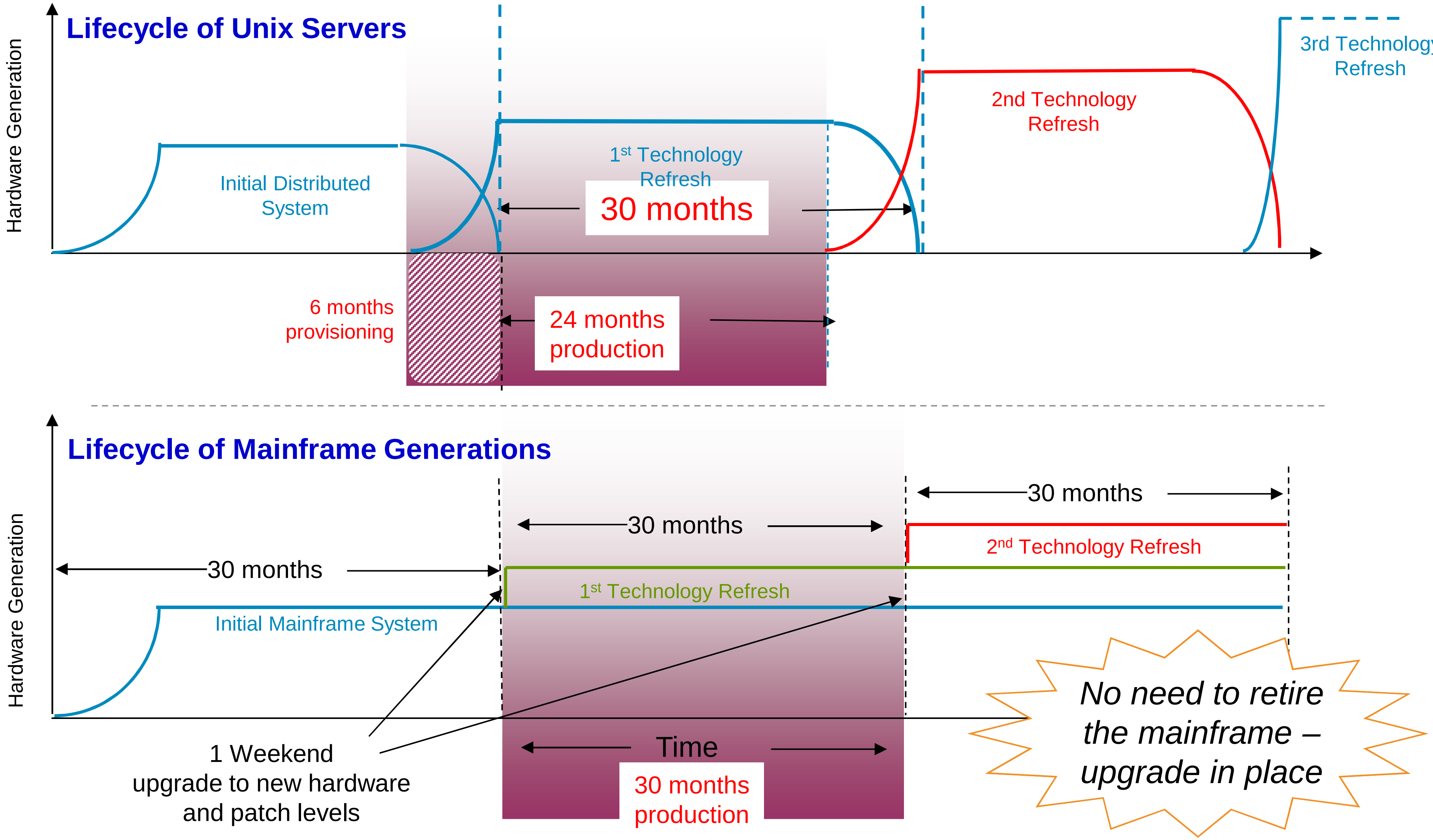
# Distributed servers are typically replaced every 3-5 years

- Hardware refreshed in 2-7 year intervals, with average 3-5 years
- New, complete servers purchased each time
  - Typically additional growth capacity added (e.g., CPU, memory, I/O, etc.)

- Upgrade normally consists of purchase of additional (new) MIPS capacity
- Existing MIPS, memory, I/O facilities, specialty processors, etc. often carried over to new hardware

5 year TCO studies make sure to include 1 hardware refresh

# Distributed server refresh leads to periods of reduced productivity along with extra costs



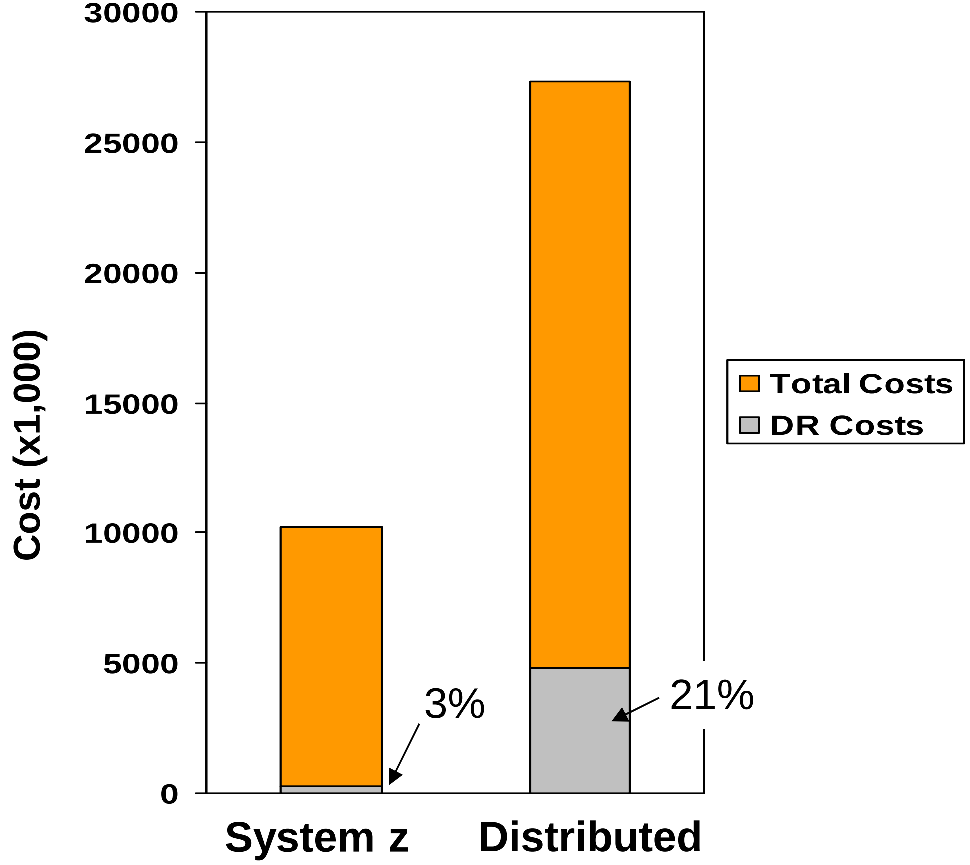
# Disaster Recovery on System z costs much less than on distributed servers

**A large European insurance company with mixed distributed and System z environment:**

Disaster Recovery Cost as a percentage of Total Direct Costs:

System z – **3%**

Distributed – **21%**




**Two mission-critical workloads on distributed servers had DR cost > 40% of total costs**

# Disaster Recovery *testing* is also more expensive on distributed platforms

A major US hotel chain calculated how much it was spending for DR testing of its 200 distributed servers…

| | *Person-hours* | *Elapsed days* | *Labor Cost* |
|---|---|---|---|
| *Infrastructure Test (3 times)* | 1,144 | 7 | $89,539 |
| *Full Test (4 times, inc. Infra Test)* | 2,880 | 13 | $225,416 |
| Annual Total – Distributed | 14,952* | 73 | $1,170,281 |
| | | | |
| Estimated Total – Mainframe | 2,051* | 10 | $160,000 |

## Customer estimates for Recovery Time Objective (RTO):

**48-60 hrs**
Distributed

**20 mins.**
Mainframe

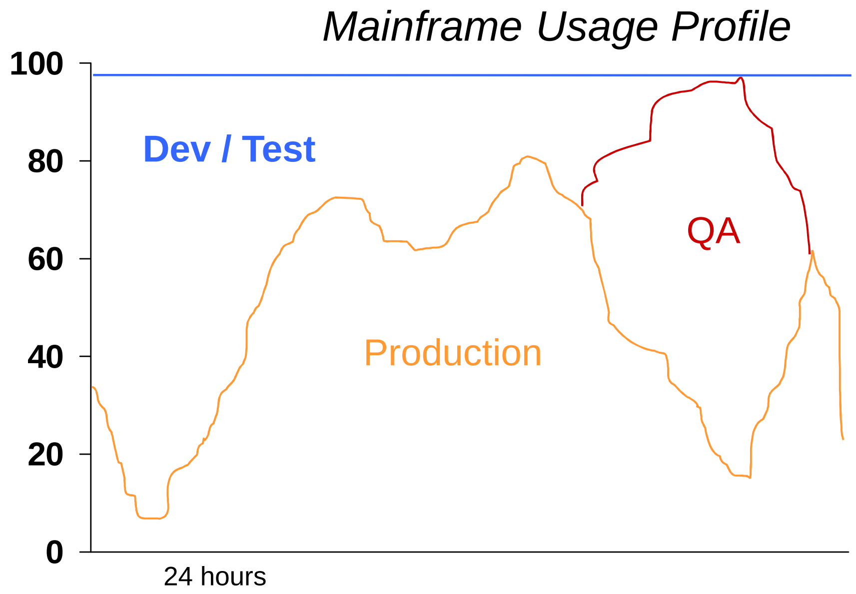
*Mainframe both simplifies and improves DR testing!*

* Does not include DR planning and post-test debriefing

# Non-production environments require fewer resources on the mainframe

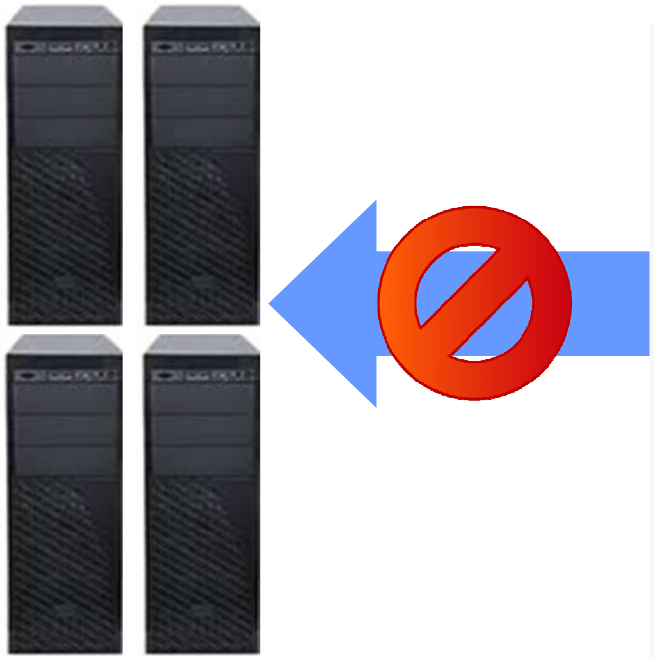
- Development and Test Capacity
  - Mainframe – Prod +20%
  - Distributed – a range, often Prod +200%

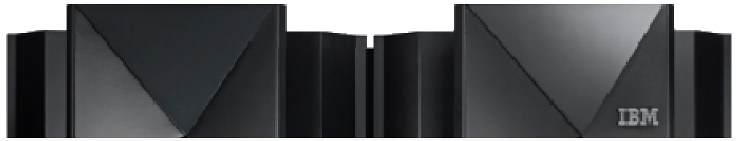
*Production* *QA* *Dev/Test*



# Replacement technologies are not always available for many mainframe functions

Rehosted platform





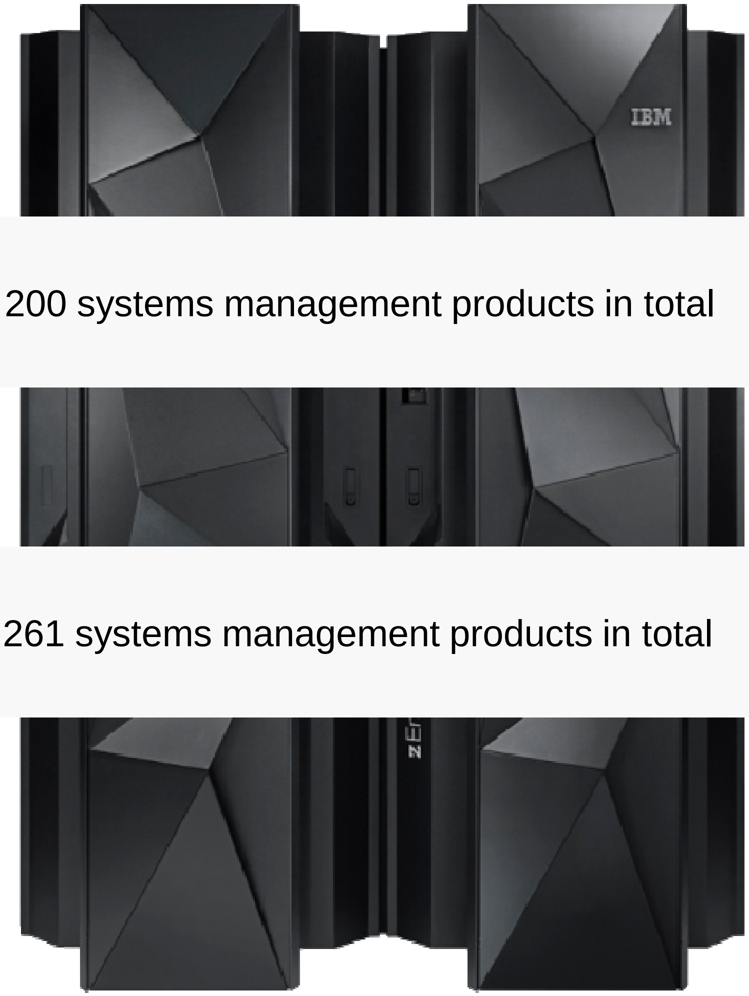

- Hierarchical databases – e.g., IMS DB and IMS DC
- Languages – e.g., PL/I, ASM …
- Batch environments including JCL with symbolic substitution, Batch pipes, Generation Data Group files for batch recovery
- System management and database tools
- 3270-style user interfaces, BMS maps, APIs…
- File structures – e.g., VSAM (alternate indexes not supported), QSAM and Partitioned Data Sets
- Print facilities including PSF, AFP, Info Print Server, JES2/3 spool
- Ability to read old backup tapes

# Eagle studies for two US retailers highlight missing systems management functionality

## Rehosted platform

- 15 replacement applications (7.5%)
- Cost = $8.4M OTC + $1.8M annually

- 53 replacement applications identifies (20%)





- 200 systems management products in total

- 261 systems management products in total

- Options?
  - Re-write applications to avoid usage
  - Write new code to perform the function
  - Add staff to manually perform the function

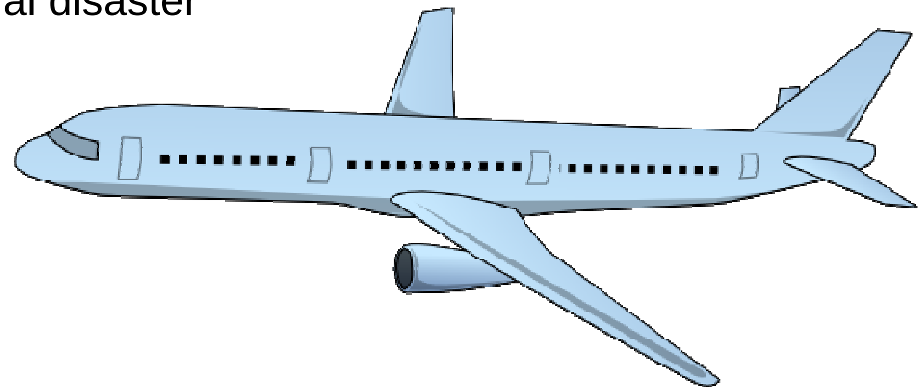
# Lessons learned can be grouped into three broad categories

- Always compare to an optimum System z environment
- Look for not-so-obvious distributed platform costs to avoid
- Consider additional platform differences that affect cost


**All examples discussed are from actual Eagle Team customer studies**

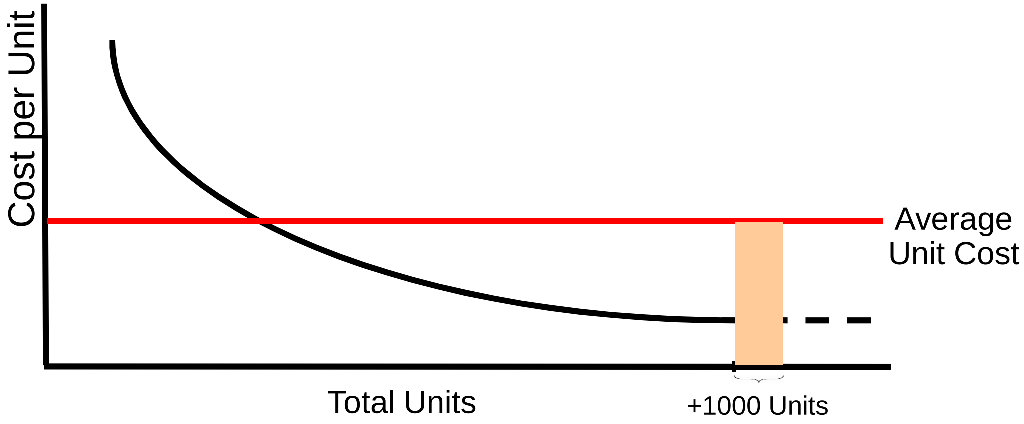
# Mainframes with Capacity on Demand can respond to unforeseen business events

- Transportation company experienced a natural disaster
  - Required them to re-run a whole weeks worth of business while continuing to operate normally
  - Able to turn on double capacity immediately to achieve this



- Customer decided to run a Super Bowl advertisement with very short notice
  - Informed IT department to expect a massive capacity spike
  - Temporarily turned on additional capacity
  - Stress tested their systems prior to the event despite short notice

# The cost of adding incremental workloads to System z is less than linear




- Mainframes are priced to deliver substantial economies of scale as they grow
- Doubling of capacity results in as little as a 30% cost growth for software on z/OS
- Average cost is significantly more than incremental cost

## Example: European bank compared costs of growing WAS applications on distributed and on mainframe

### Incremental cost of adding one large WAS application to platform (5 yr. TCO):

| | |
|---|---|
| **€1.56M** | **€1.29M** |
| **(378K OTC, 192K Y1, 249K Y2-5)** | **(657K OTC, 42K Y1, 147K Y2-5)** |
| Distributed | Mainframe |

*Future deployments will be targeted to the mainframe!*

# Distributed platforms don't often benefit from consolidation and therefore MUST run at low utilizations – even with virtualization!

- **Large insurance company considering moving applications to virtualized x86**
  - Believed this was a high utilization, low cost platform compared to other alternatives
  - Note costs are normally impacted largely by core count and software cost per core

- **Used readily available utilization data to demonstrate extremely low x86 utilization**
  - On average the provisioned systems were used at less than 15%, peak less than 20%
  - This despite many of the 75 hosts running up to 40 VMs each (unusually high)

- **Further investigation shows the various practical constraints that lead to this effect**
  - RAM shortages (normally no physical RAM overcommit allowed)
  - Limited virtual CPU overcommit (vCPU co-scheduling issues)
  - Enforced separation of production from non-production (isolation issues)
  - Limit to the number of VMs per host (to limit workload migration time requirement)
  - Presence of many idle workloads (wasting RAM and driving up the RAM/core ratio to impossible levels, thereby forcing idle cores)

- **System z does not suffer from these issues and normally runs at high utilizations**
  - Averages normally above 50%, often see 65% and above, unheard of on other platforms
  - Most System z machines run more workloads in a single LPAR than other platforms run on the whole physical server, even for large servers – and hence benefit from significant consolidation

# Typical x86 hypervisor deployment rules of thumb

- As reported by a very large x86 hypervisor customer

  1. **Host memory virtualization <95%**
  2. **Number of VMs per host <40:1**
  3. **vcpu to physical <4:1**
  4. **PROD and Non-PROD separated**

- Note that none of these constraints are associated with CPU utilization!

- Basically the primary constraint is physical RAM – once virtual machines are defined that represent 95% of physical, no more guests are allowed
- The second constraint is the number of VMs on the server – this is likely due to the hypervisor only allowing a limited number of concurrent migrations, and they take a long time, so shutting down a physical box can require hours of manual work
- The third constraint suggests there may be vCore overheads and/or restrictions around co-scheduling of virtual to physical cores
  - ie. A VM will not run unless all of its virtuals can be backed by physicals at once
- The final constraint is not surprising given what we've seen in workload management testing – non production workloads can easily consume CPU intended for production workloads despite configuring the x86 hypervisor to explicitly avoid that ("bleed" between supposedly isolated workloads)

# Consolidation capability is a much larger sizing factor than hardware age or differences in technology

- **Most customers use benchmark results to differentiate servers on some level**
  - e.g. RPE's from Gartner/Ideas, SAPS for SAP, tpcc results, etc.
  - These generally give a good estimate of change in technology over time
  - They may also be able to differentiate between specific technologies for specific workloads

- **Unfortunately these benchmarks do NOT capture the most significant factor in IT today**
  - The ability to effectively consolidate many mixed workloads into a shared pool of resource
    - Shared in this context means all unused resource is available to any workload immediately
    - x86 hypervisors try to define shared across "clusters" of servers but that is not truly shared
    - Oracle RAC tries the same thing with their "shared" cache – still not actually shared!

- **So let's examine the magnitude of the different factors for an example case**
  - To keep things simple we'll stay with "distributed" style workloads and compare x86 to large POWER
    - The result going to System z is dramatically more impressive, especially going to z/OS
  - Benchmarks suggest that the same generation x86 cores and POWER cores are similar on a core-to-core basis (of course some benchmarks say higher, others lower)
    - So the technology factor is basically 1 (and we set age factor to 1 by using same age hardware)
  - So what is a typical consolidation factor (assuming both platforms are virtualized)?

- **A typical real customer example for a single mid-sized application suggests a 3-4x factor**
  - So the ability to consolidate effectively is substantially more important in sizing than benchmarks suggest
  - Also note benchmarks tend to be single-image, incorrectly implying less capacity per additional core

# Most customer cost tracking (or chargeback) creates adverse selection issues which require unintuitive steps to correct

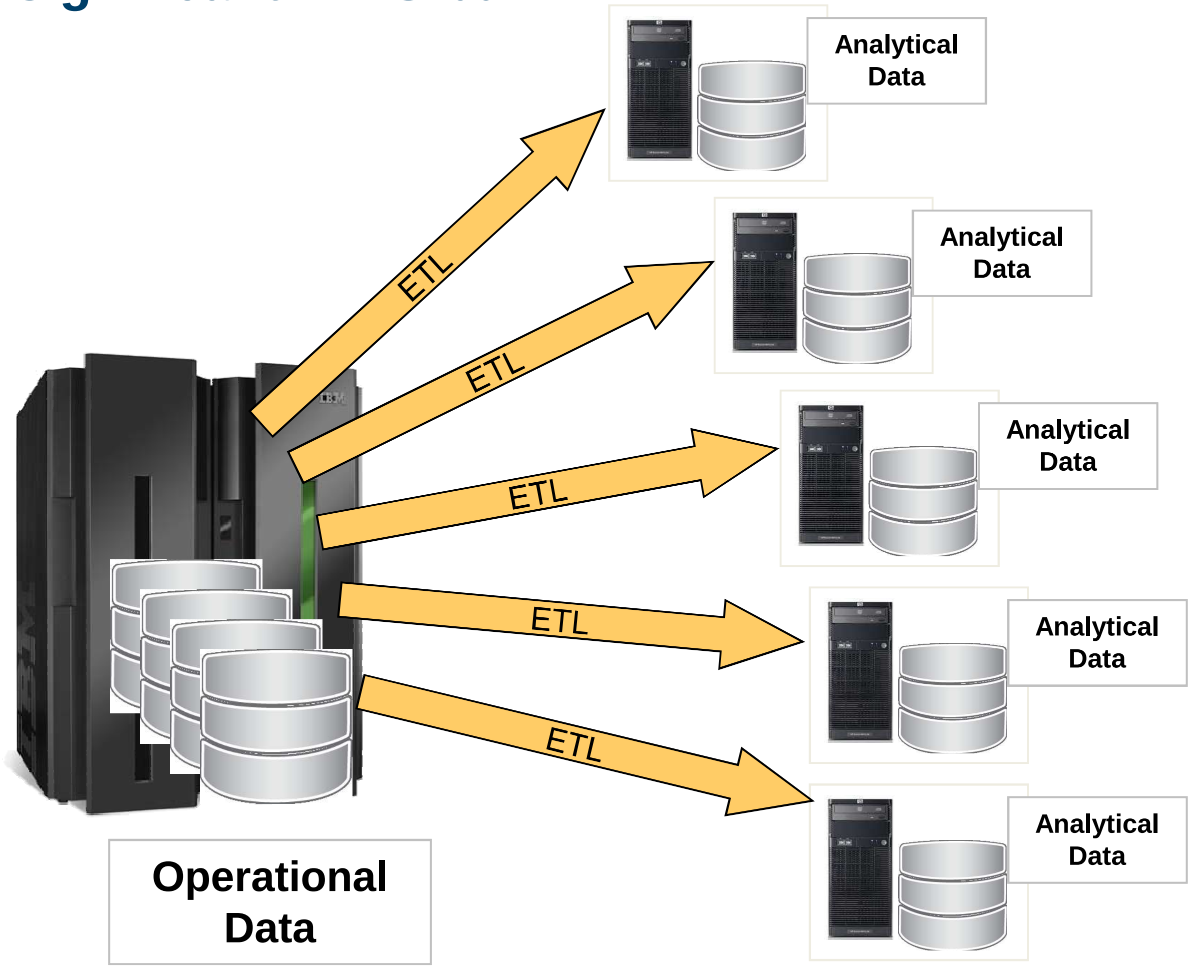
- **Simple thought experiment involving a single shared infrastructure system**
  - We'll use a real customer example of an x86 based "private cloud" (production only)
    - 600+ virtual servers, mostly running Windows, some Linux
  - Calculating an average cost per VM for the whole system gives us a $/VM/yr

- **Now divide the same infrastructure into two pools**
  - Large pool of lightest VMs taking about 50% of the actual resource consumed
  - Small pool of heaviest VMs, also taking about 50% of the actual resource consumed
  - Calculate an average cost per VM for these two new pools
    - The light VM pool is now much cheaper than the heavy pool on a per VM basis
    - This is because both pools use about the same resources, but have very different VM counts

- **Follow the logic to understand that this creates a stunning case of adverse selection**
  - Line of Business users are now incented to move workload from the heavy to the light pool
  - Even though the total cost of the whole infrastructure is unchanged whatever happens!

- **What if you have a pool which is actually more efficient for most workloads?**
  - It tends to run the heaviest workloads, and therefore shows up as more expensive – sound familiar?
  - The incentives and cost data tell the business they should move off to the cheaper pool(s)
  - The real solution is actually the opposite – to move more workload onto the more efficient pool
    - This is trivial to prove mathematically, I'll leave that as an exercise for the reader!

# A common "Mainframe Quarantine" strategy can eventually result in significant MIPS burn



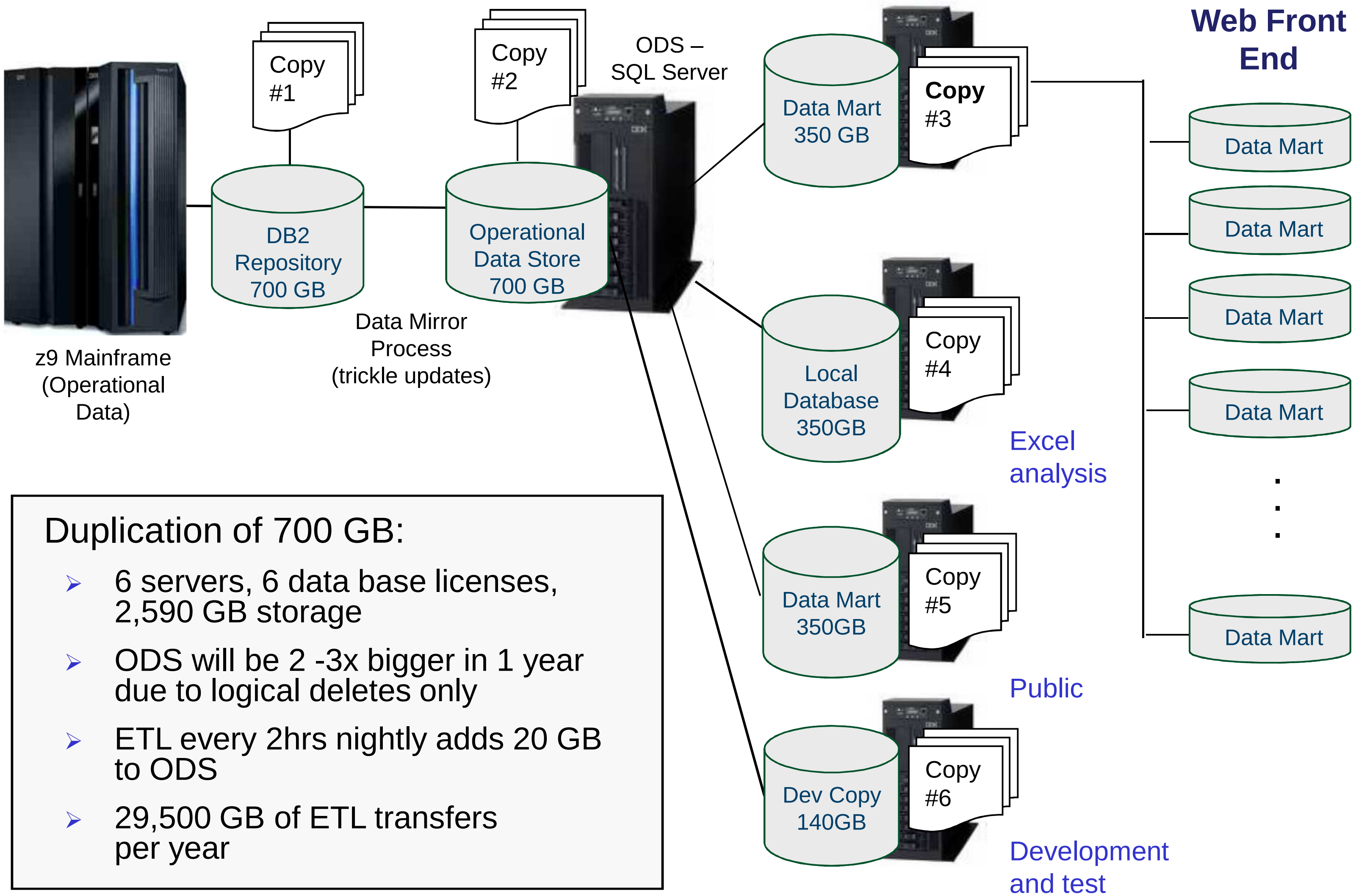

## A large European bank:

- 120 database images created from bulk data transfers
- 1,000 applications on 750 cores with 14,000 software titles
- ETL consuming 28% of total distributed cores and **16% of total MIPS**

## A large Asian bank:

- One mainframe devoted exclusively to bulk data transfers
- ETL consuming 8% of total distributed core and **18% of total MIPS**

# Data proliferation within a state government judicial system is out of control



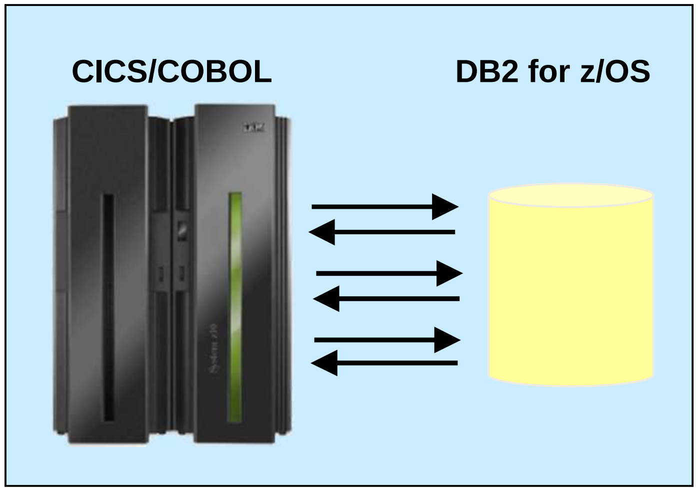

Duplication of 700 GB:

- 6 servers, 6 data base licenses, 2,590 GB storage
- ODS will be 2 -3x bigger in 1 year due to logical deletes only
- ETL every 2hrs nightly adds 20 GB to ODS
- 29,500 GB of ETL transfers per year

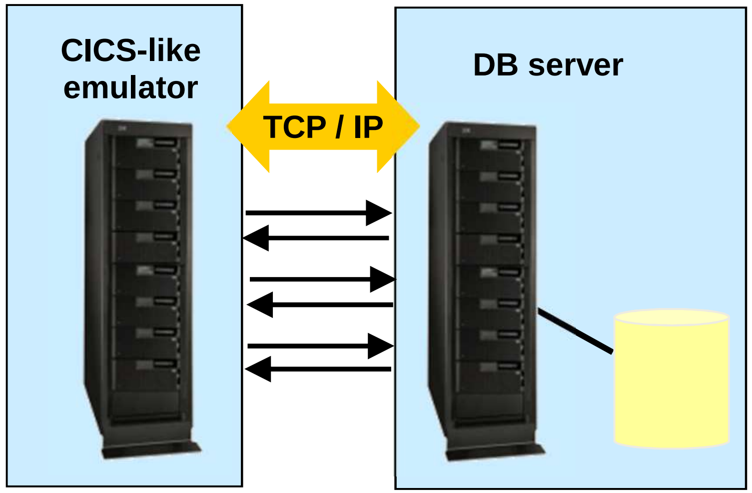
# Some applications originally designed with co-located data are not good offload candidates

- Large insurance company rehosted portion of application as POC
  - Found TCP/IP stack consumed considerable CPU resource, and introduced security compromises and network latency
- European bank tried rehosting CICS workload to Linux while maintaining VSAM and DB2 data on System z
  - Induced latency resulted in CICS applications no longer meeting its SLA

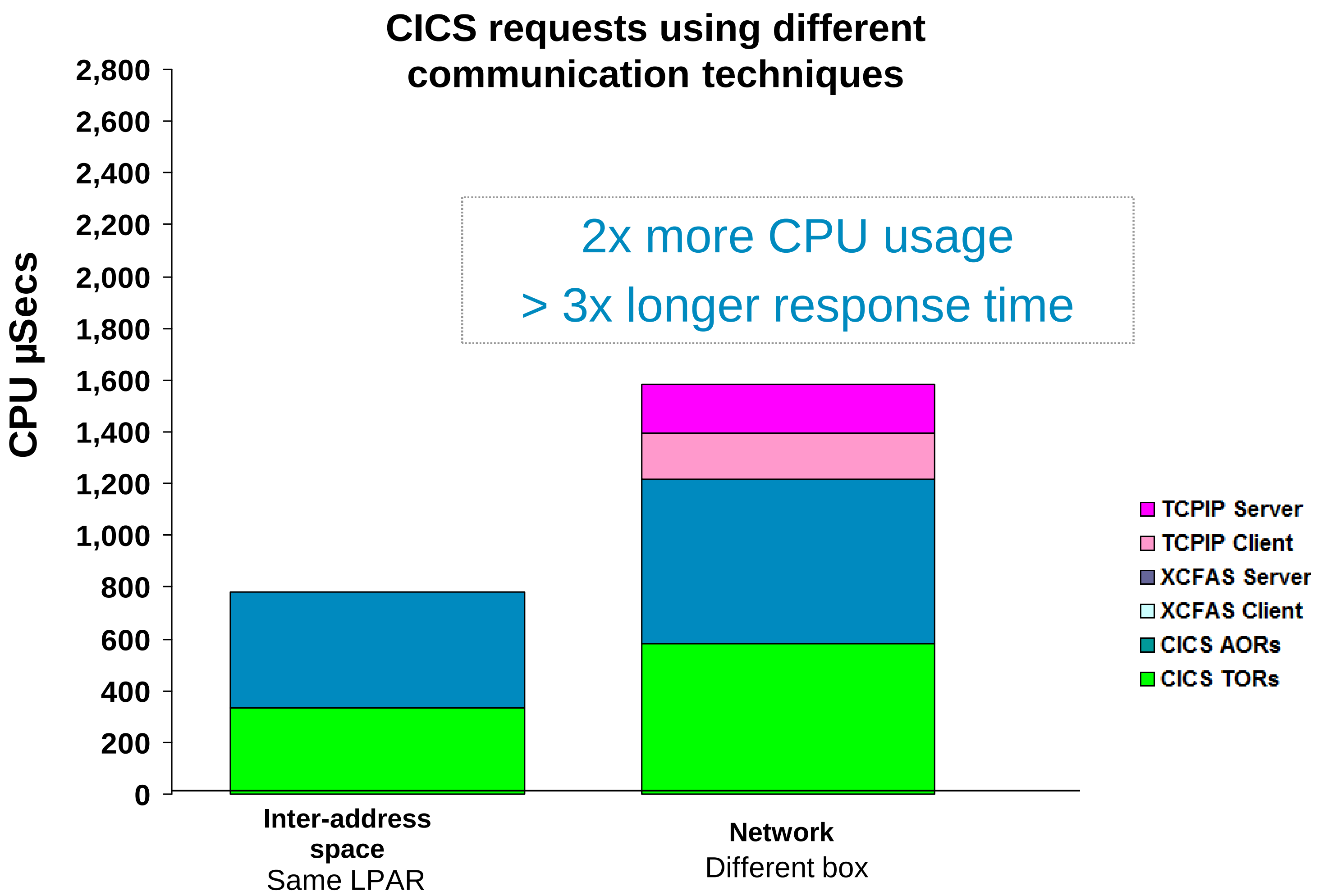
**Single z/OS LPAR**




**Distributed architecture**



# Co-locating in the same address space is more efficient




Source: http://hurgsa.ibm.com/projects/t/tp_performance/public_html/OS390CICS/reports/CICS%20TS%20V4.2%20Performance.ppt
and email with z/OS Communications Server development team

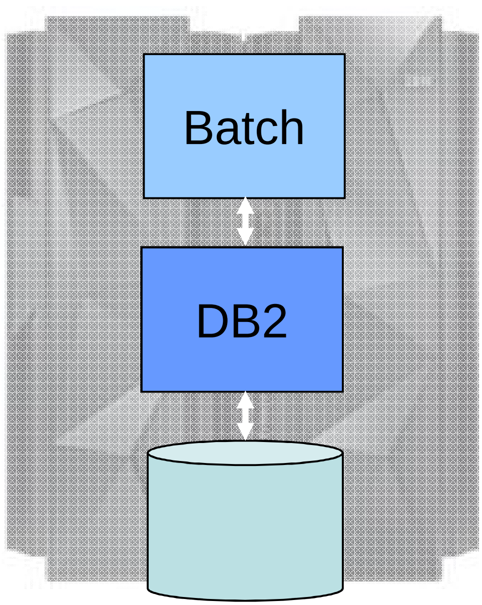
# Moving Batch applications off the mainframe can have serious consequences

- Customer was facing large one-time charges for mainframe growth
- Rehosting vendor committed to a quick partial migration to avoid mainframe growth

*Before:*

**1** Mainframe CPU usage units

**1** Units of elapsed job time




**System z**

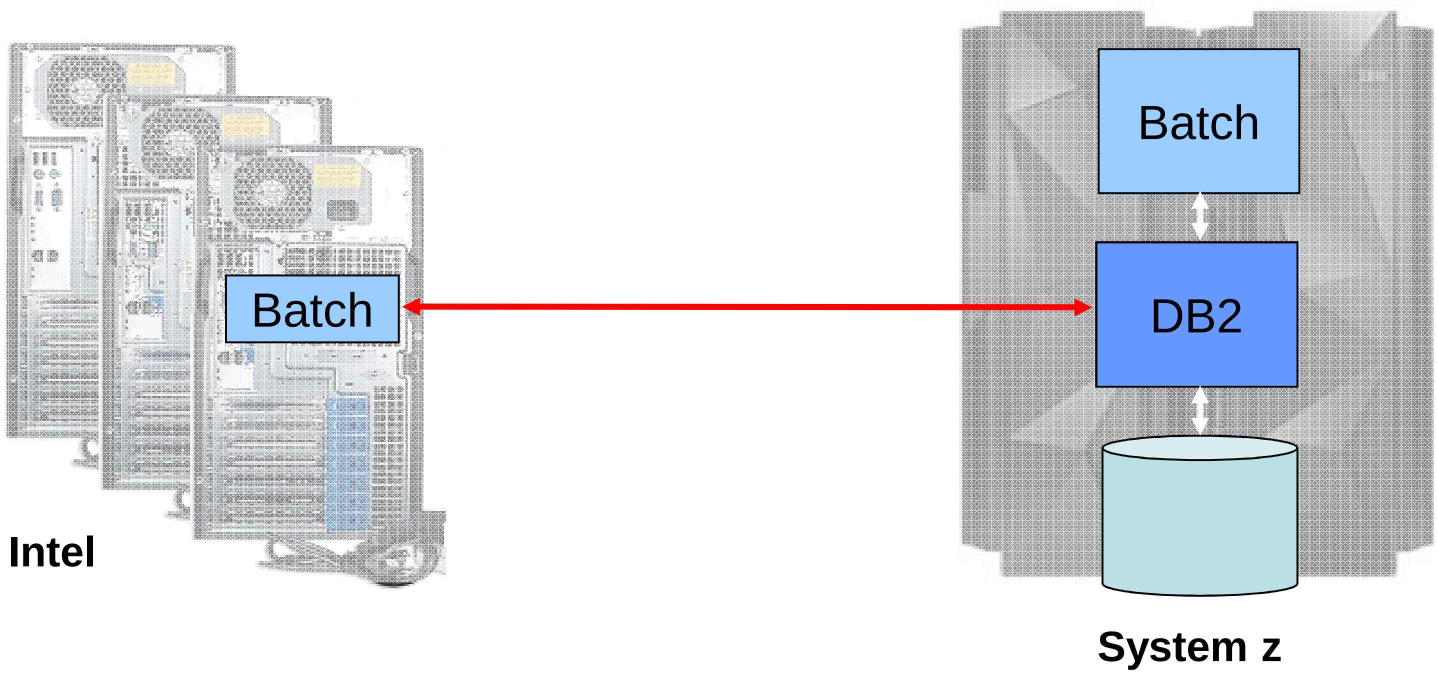
# Moving Batch applications off the mainframe can have serious consequences

- Additional DRDA processing doubled mainframe CPU usage even though the application was now running on Intel
- Additional network latency dramatically increased elapsed job time (10-25x)

*After:*

**2** Mainframe CPU usage units

**10-25** Units of elapsed job time



# Large systems with centralized management deliver better labor productivity

Large US Insurance Company

**HP Servers + ISV**

Production Servers

HP 9000 Superdome RP4440

HP Integrity RX6600

Dev/Test Servers

HP 9000 Superdome RP5470

HP Integrity RX6600

Claims per year **327,652**

**$0.12 per claim**

**$0.79 per claim**

**Mainframe support staff has 6.6x better productivity**

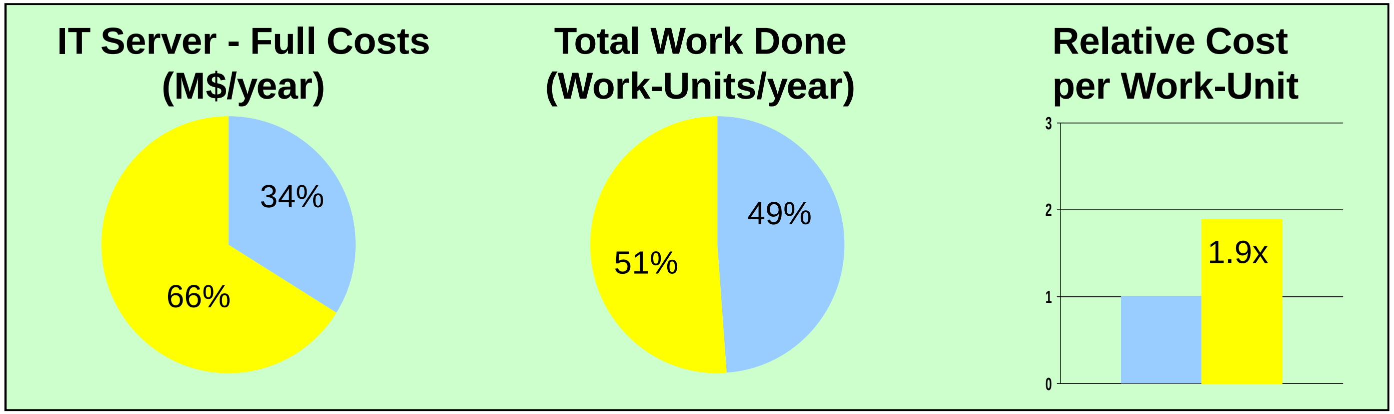
**IBM System z CICS/DB2**

Total MIPS **11,302**

MIPS used for commercial claims processing prod/dev/test **2,418**

Claims per year **4,056,000**

# Cost per unit of work is much lower for the mainframe than for distributed platforms

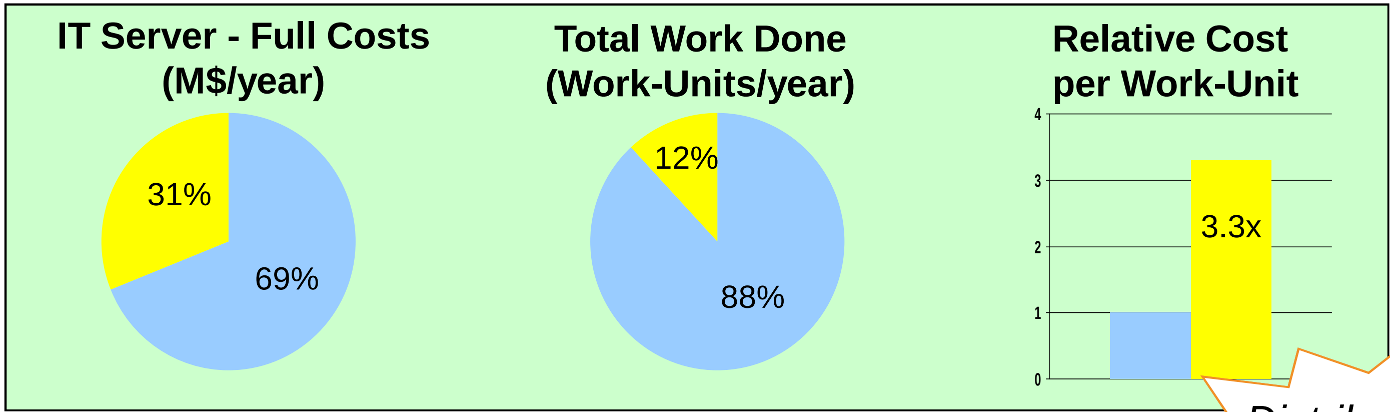



Case 1:
Very large retail bank – total of service delivery plus application development

Mainly UNIX distributed (>5000 servers)

*Mainframe*

*Distributed*




Case :
Medium retail bank service delivery only. Mainly mainframe 2,500 MIPS, 13M txns/day.
Limited distributed servers Windows + some UNIX
(~350 servers, 12% util'n)

*Distributed cost is **2-3x** more than mainframe!*

Data from 3Q06 Scorpion studies

# The IBM Eagle team helps customers understand mainframe costs and value

- **Worldwide** team of senior technical IT staff
- **Free of Charge** Total Cost of Ownership (TCO) studies
  - Help customers evaluate the lowest cost option among alternative approaches
  - Includes a one day on-site visit and is **specifically tailored to a customer's enterprise**
- Studies cover POWER, PureSystems and Storage accounts in addition to System z
  - For both IBM customer and Business Partner customer accounts
- Over 300 customer studies since formation in 2007
- Contact: eagletco@us.ibm.com