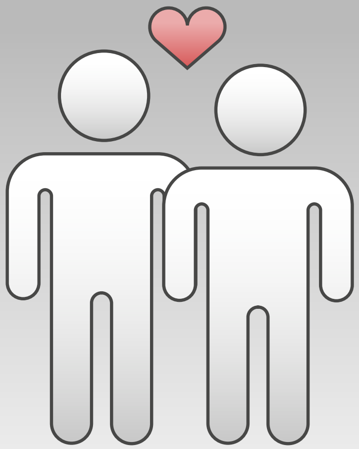
# Lock-in and the Lifecycle of Dependencies



Choosing, Using and Losing

Adrian Cockcroft

What is the return on investment (ROI) for each phase?

Choosing

Using

Losing

What is the ROI for each phase?

How has ROI changed with advances in technology and practices?

- Choosing
- Using
- Losing

Choosing

Using

Losing

# Choosing

## Investments

Negotiating, learning, experimenting

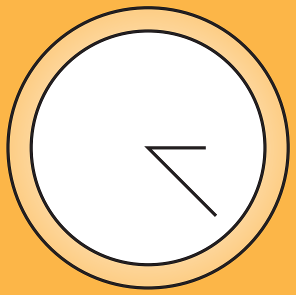
Hiring experts, building

Installing, customizing

Developing, training

# Choosing

How much
time elapses?

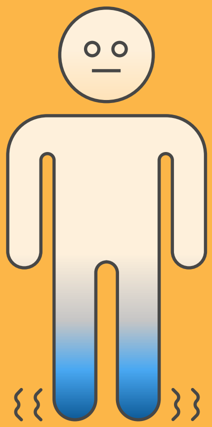


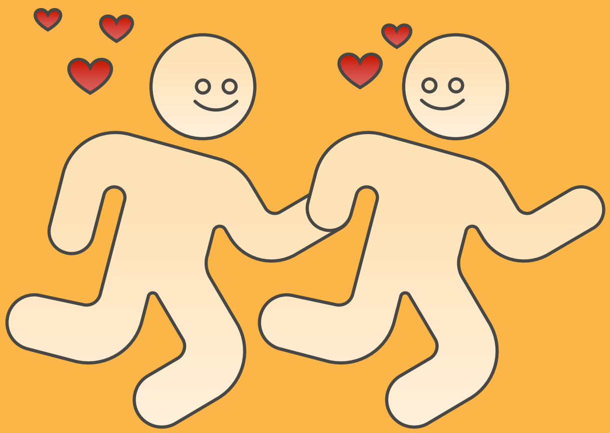
“The best decision is the right decision. The next best decision is the wrong decision. The worst decision is no decision.”

—Scott McNealy

# Choosing

Analysis Paralysis vs. Snap Judgement





## Choosing

### Making a commitment

Whenever development is frozen, and the operations team takes over, the key is turned in the lock

# Choosing—What Changed?

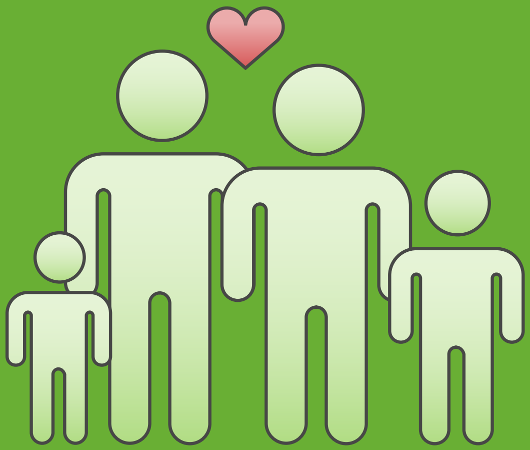
| Old World | New World |
|---|---|
| Monolith—all in one | Microservice—fine grain |
| Proof of concept install | Web service/Open source |
| Enterprise purchase cycle | Free tier/free trial |
| Months | Minutes |
| $100K–$Millions | $0–$1000s |

Choosing

Using

Losing

## Using



### Investments

Cost of setup

Cost of operation

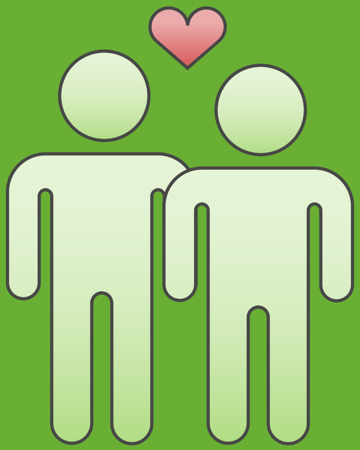
Capacity planning

Scenario planning

Incident management

Tuning performance and utilization

# Using



## Returns

Service capabilities

Availability, functionality

Scalability, agility

Efficiency

## Using - What Changed?

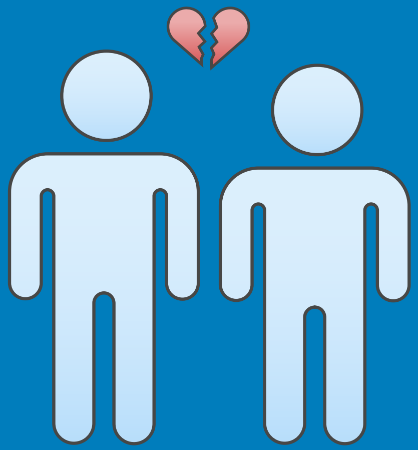
| Old World | New World |
|---|---|
| Frozen installation | Continuous delivery |
| Ops specialist silo | Dev automation |
| Capacity upgrade costs | Elastic cloud resources |
| Low utilization | High utilization |
| High cost of change | Low cost of change |

Choosing

Using

Losing

# Losing



## Investments

Negotiating time

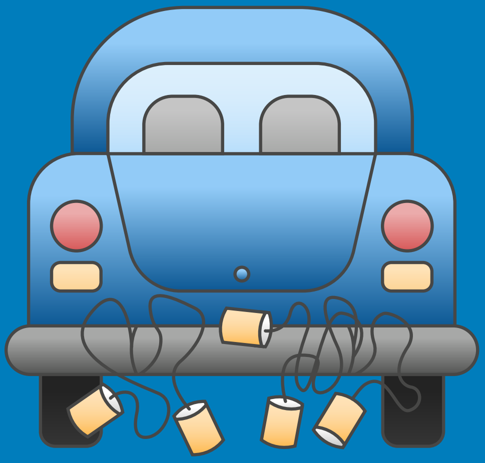
Contract penalties

Replacement costs

Decommissioning effort

Archiving, sustaining legacy

# Losing



## Returns

Reduced spending

More advanced technology

Better service, agility, scalability

Choose again, the cycle continues…

## Losing—What Changed?

| Old World | New World |
|---|---|
| Monolithic—all or nothing | Microservices—fine grain |
| Frozen waterfall projects | Agile continuous delivery |
| Long term contracts | Pay as you go |
| Local dependencies | Remote web services |

| **Old World**<br>Monolithic on-prem waterfall lock-in | **New World**<br>Agile cloud-native micro-dependencies |
|---|---|
| Years | Weeks |
| Millions of dollars | Hundreds of dollars |
| 100s of dev years | A few dev weeks |
| Lock-in | Refactoring |
| Lawyers and contracts | Self service |

# Bottom line

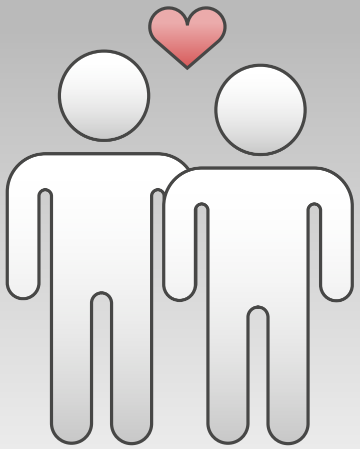
ROI for choosing, using, losing has changed radically. Stop talking about lock-in, it’s just refactoring dependencies

The cost of each dependency is far lower

Frequency of refactoring is far higher

Investment and return is much more incremental

# Lock-in and the Lifecycle of Dependencies



Choosing, Using and Losing