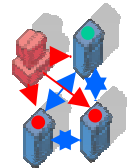




# Taurus Software: Lessons from migrations

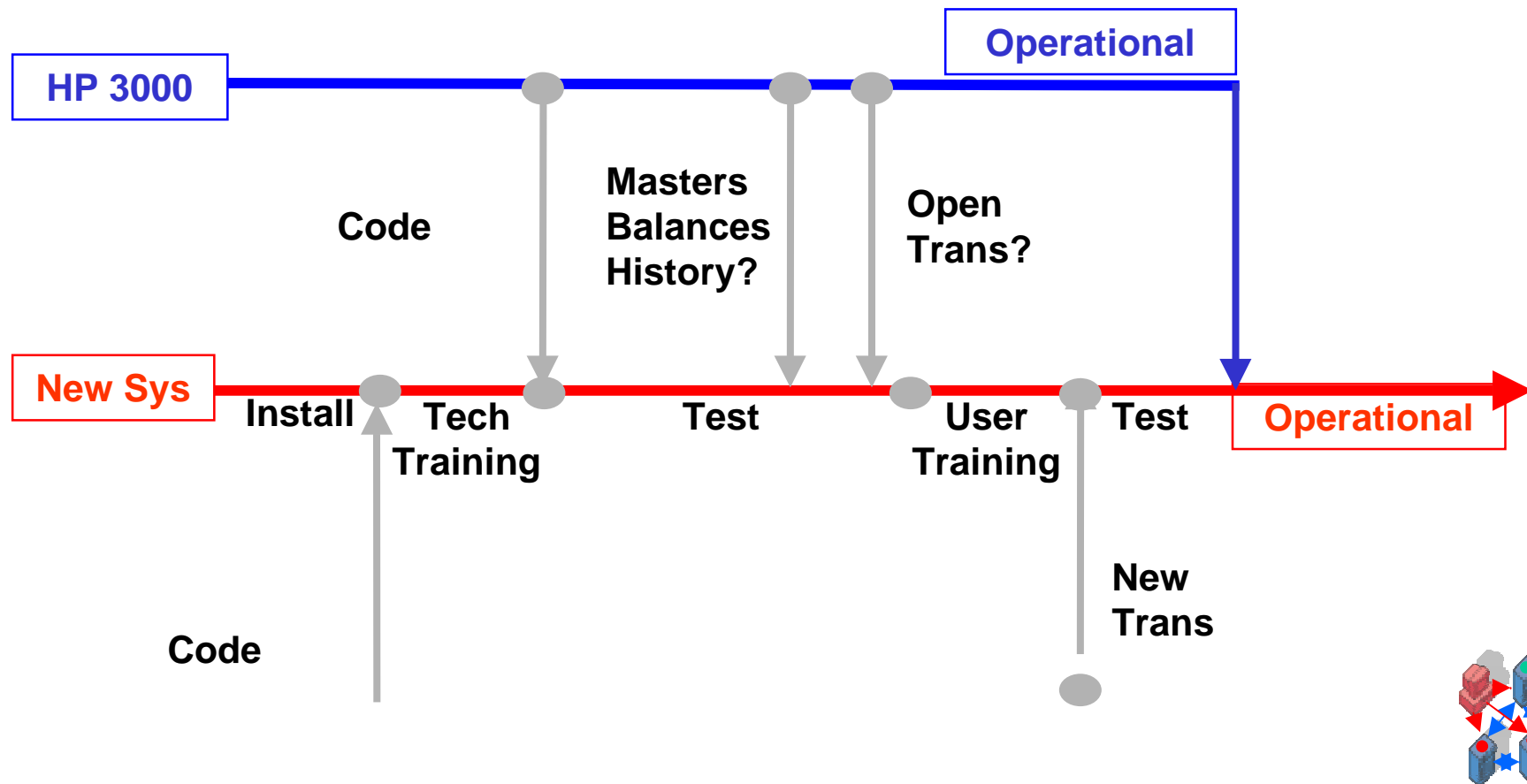
I wish I knew then  
what I know now...





# Taurus Software: Lessons from migrations

## The general process





## Your Migration Type?

### Migration Type

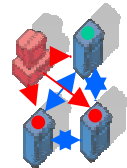
Big Bang (weekend)

Big Bang (weeks)

Phased (months)

### Risk?

High? Low?





## High risk IT projects



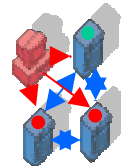
### Predictors of low risk

Strategic

Familiar

Reversible

Sunk costs





# Taurus Software: Lessons from migrations



## Not Strategic

### Executive Commitment

Bet the company?

Support: visible, strong; actionable,

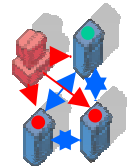
### Team Pride

Members have vested interest in success

Failure is not a option

Altruism

### Communications Plan





## Not Leveraging Sunk Costs

### Capital

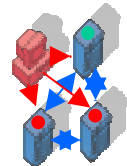
Hardware; Infrastructure

### Labor

The team(s)

### Corporate memory: aka Tribal Lore

Business Processes; Undocumented stuff





# Taurus Software: Lessons from migrations



## Not familiar

### Hardware; Infrastructure

Start learning now; document as you go

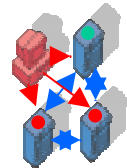
### New Application

User interface; help desk; maintenance plan

### Skill up versus Skill Out

Business Processes

Tribal Lore: aka Informal Business Rules





## Not familiar

### Dirty Data

60-80% Rule of fields

### IMAGE

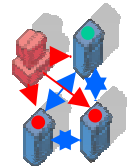
Is IMAGE strongly typed?

Examples: in an hour

### REDEFINES

Examples: in an hour

Tribal Lore: aka Informal Business Rules







## Not Reversible

### Not Phased (Big Bang)

Consider phasing when:

- Project has over 40 Activities

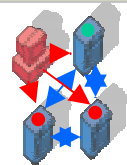
- The new Business Process is not fully understood

- When everything will change (target, interface, processes)

- When you need training on the new

- When you can't size dirty data problem

Always have a Fire Escape





# Taurus Software: Lessons from migrations



## Not Reversible

Test at dataset/module level

Parallel test/run

Tie pay to performance

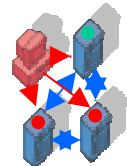
Vendor Warranties

Turnkey is not turn away

Fixed price is not fixed success

Kick the tires

Always have a Fire Escape





## Not Reversible

### Late Tangibility

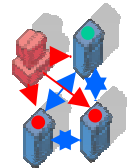
Must have test system

What about training?

100% testing by experts

Transactions?

Acceptance criteria





# Taurus Software: Lessons from migrations



Now I know what I need to know

Think strategically

Leverage sunk costs

Get familiar

Look for the fire escape

