# The Formula for Success in System Testing

Nathan Petschenik
STS Consulting
www.stsv.com
732-257-9033 x 12
nathan@stsv.com

# The Formula for Success in System Testing

**System Testing Success =**

**(Technical Excellence)**

**+**

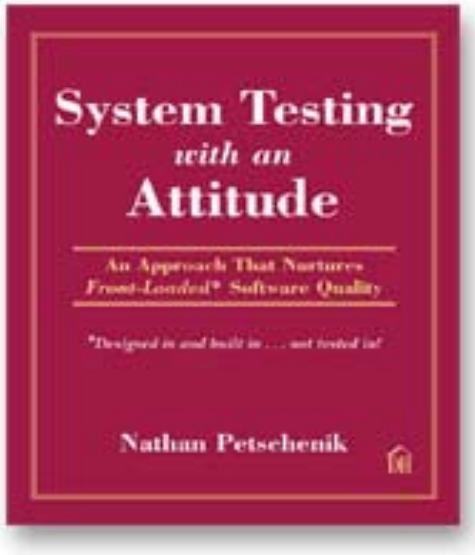
**(Nurturing Front-Loaded Quality)**

***where:***

**(Technical Excellence) >0**

**(Nurturing Front-Loaded Quality) >0**

# Main Source



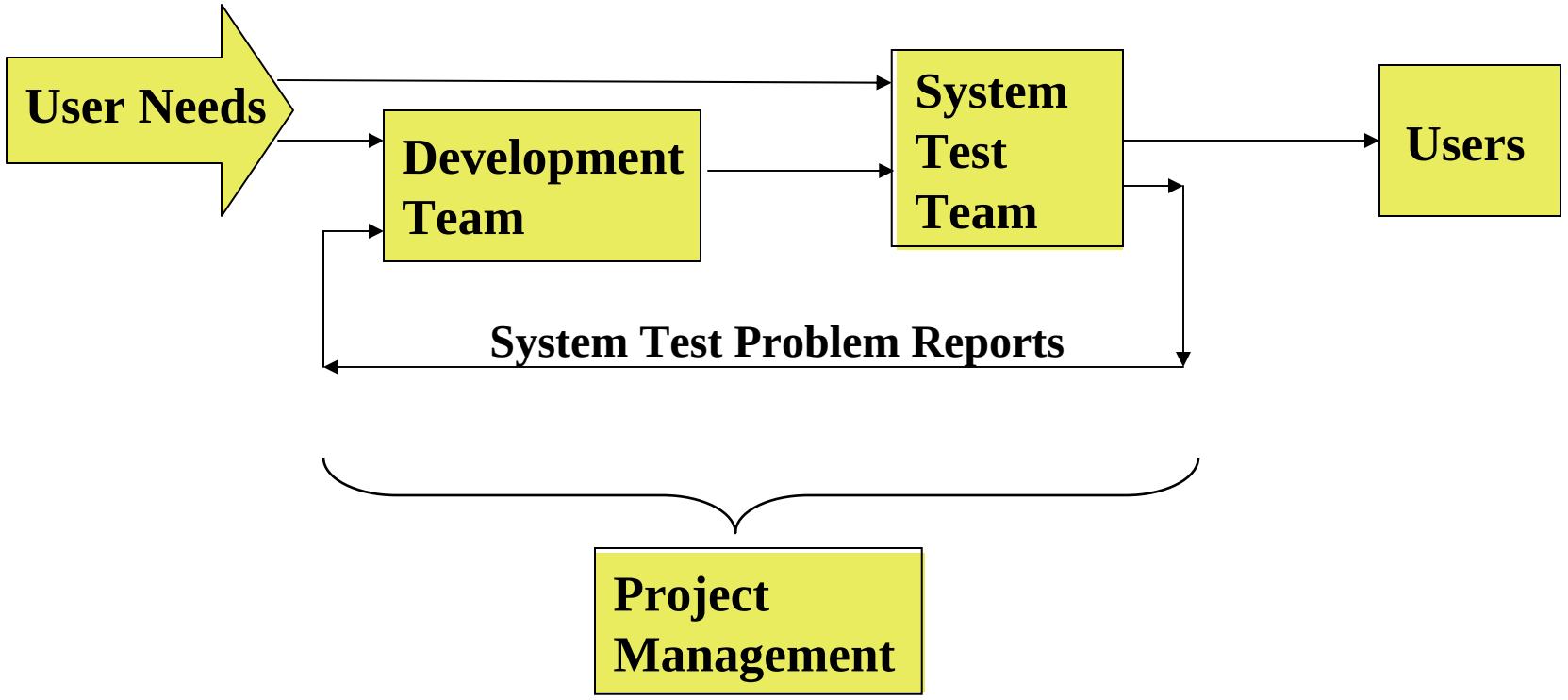

## System Testing with an Attitude

*An Approach That Nurtures*
*Front Loaded Software Quality*

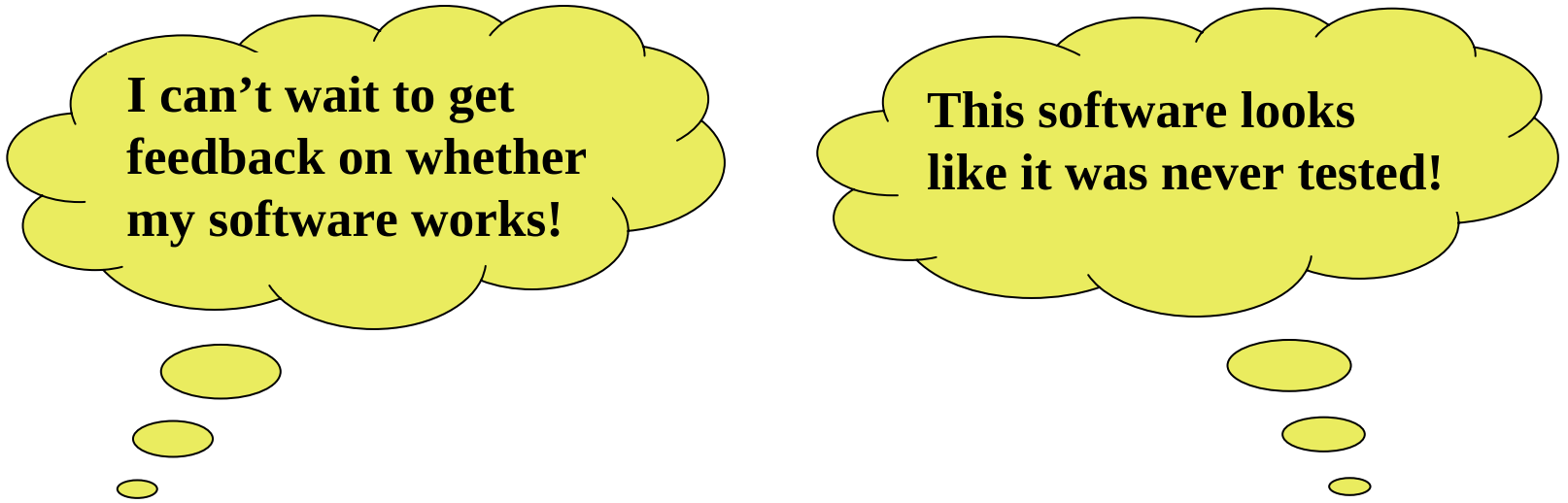
by Nathan Petschenik

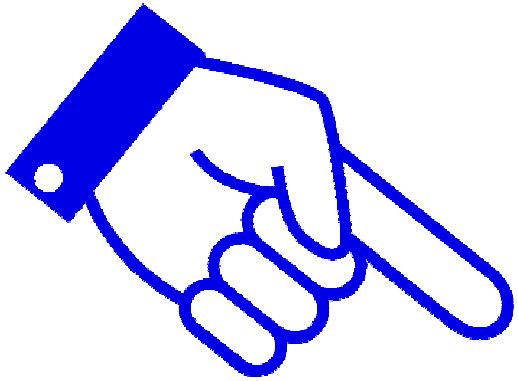
ISBN: 0-932633-46-3
Dorset House, New York

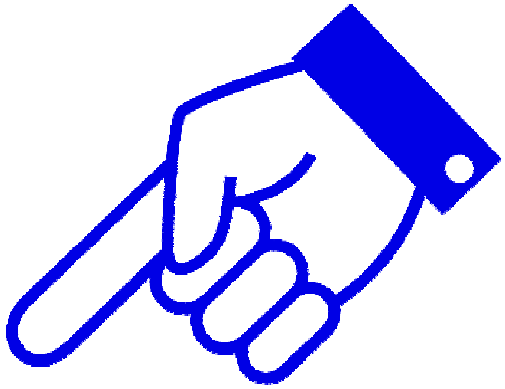
# Model of System Testing



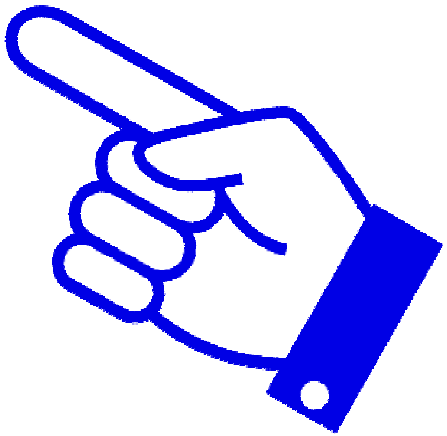
# Misunderstanding Between Developers and Testers







***How could you have missed <u>that</u> problem?***



# Role Awareness Seminars

- Objectives:
  - Build awareness of system testing issues
  - Clarify expectations
  - Change attitudes
  - Influence behavior
- Technique:
  - Small groups of developers, system testers, and project managers
  - 12 True/False Questions
  - Open discussion of answers

# Sample Questions from Role Awareness Seminar

The role of the System Test Team is:

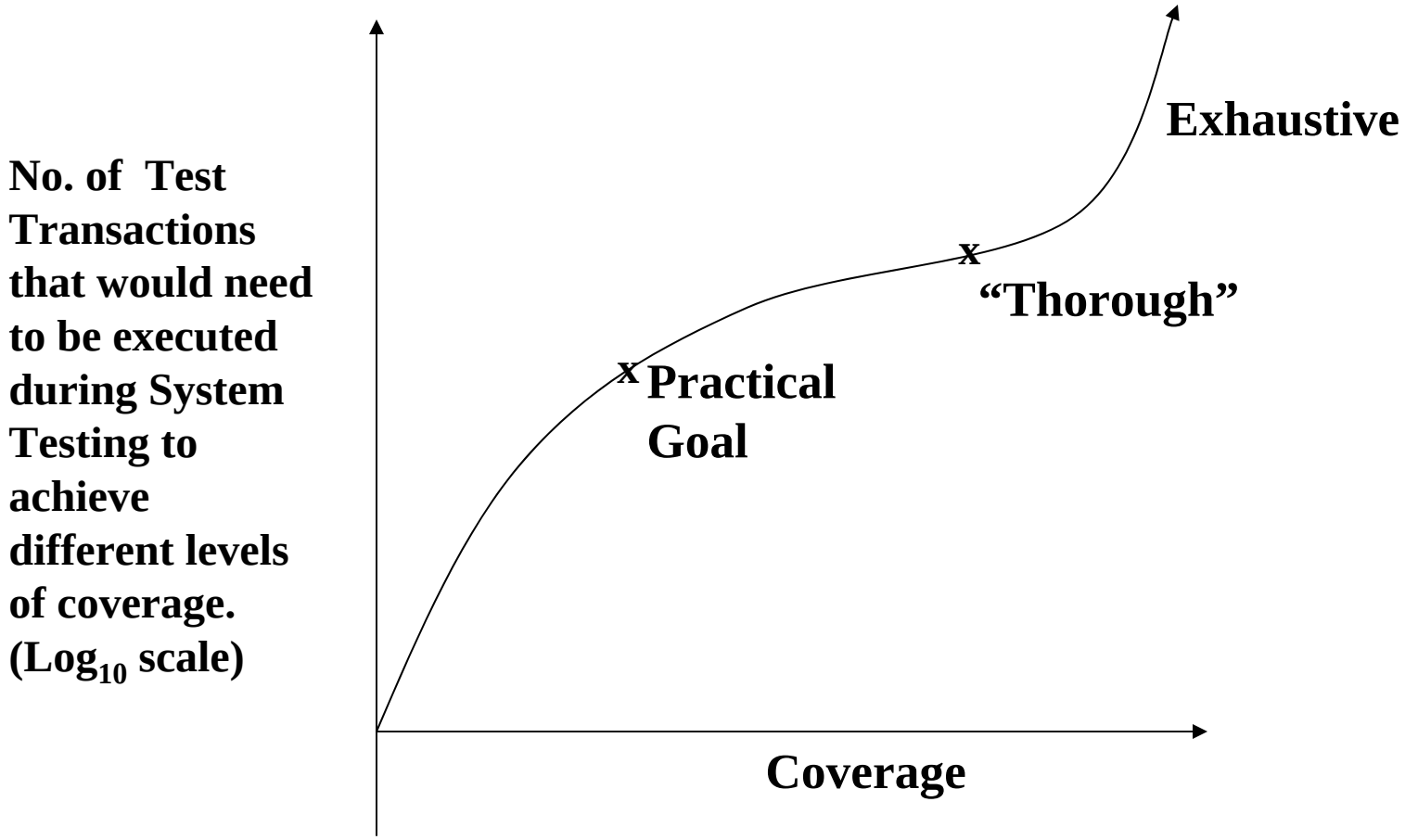
to supplement the testing done by the developers so as to achieve *exhaustive* coverage of the system.

T ( )  F( )

to test and retest the system *thoroughly* during the system test phase of the project.

T ( )  F( )

# Levels of System Testing



# Selecting Test Cases for System Testing

System testers select test cases using factors not considered in white box and black box coverage methodologies

- System Testers are looking for “important” problems that must be fixed before the system is provided to users
- System Testers are looking for the types of problems that may have been missed in the testing performed by developers prior to System Testing

# Practical Priorities in System Testing

**Rule A: Testing the system's capabilities is more important than testing its components.**

**Rule B: Testing old capabilities is more important than testing new capabilities.**

**Rule C: Testing typical situations is more important than testing unlikely situations.**

# Sample Question from Role Awareness Seminar

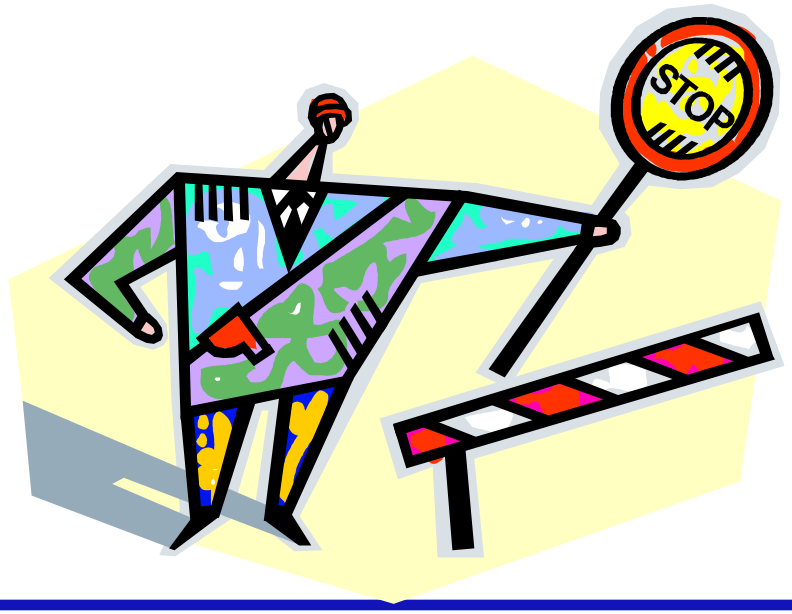
The role of the System Test Team is:

to help developers complete their testing.

T ( )  F( )

# My Entrance Criterion to System Test

*Developers must believe that the software is ready for users on the day that the software enters the system test phase*



# Obstacles to Meeting My Entrance Criterion

- Staff?
- Schedule?
- Lack of a stable integration testing environment?
- Lack of training in testing?
- Tools?
- Holes in Requirements?

# Results of Breaking Down Real and Perceived Barriers to Quality

- Better developer testing
- Higher quality software to System Test
- Higher quality software to Users
- Productivity increases
- Non-tangible benefits

# Ways for a System Test Leader to Nurture Front Loaded Quality

- Conduct Role Awareness Seminars (and stimulate action on follow-up items)
- Formalize Entrance Criteria to System Test
- Infiltrate the SDLC
- Champion the need for static testing techniques throughout the SDLC
- Be the collector and interpreter of measurements
- Become a role model for accountability

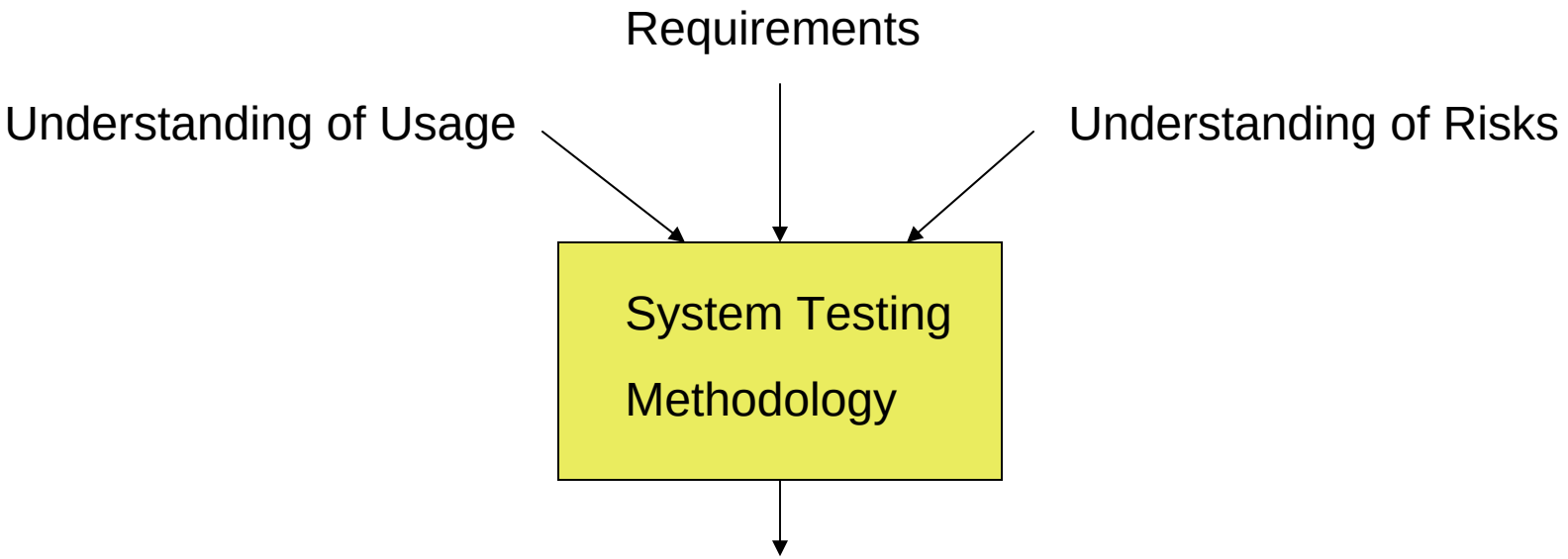
# What does it mean for a System Test Team to Achieve Technical Excellence?

- Team consistently prevents important problems from reaching users
- Team members share a common approach, process, and/or methodology for addressing the technical challenges of system testing

# Elements in the Science of System Testing

1. What system tests need to be developed?
2. How do you design system tests?
3. How do you make a test repeatable?
4. How do you solve cycle acceleration problems?
5. How do you test transactions-in-progress?
6. What should you measure?

# 1. What system tests need to be developed?




- Architecture of System Tests
- Coverage responsibilities of each System Test in the Architecture (Test Objective)
- Relative Priorities for developing System Tests in the Architecture

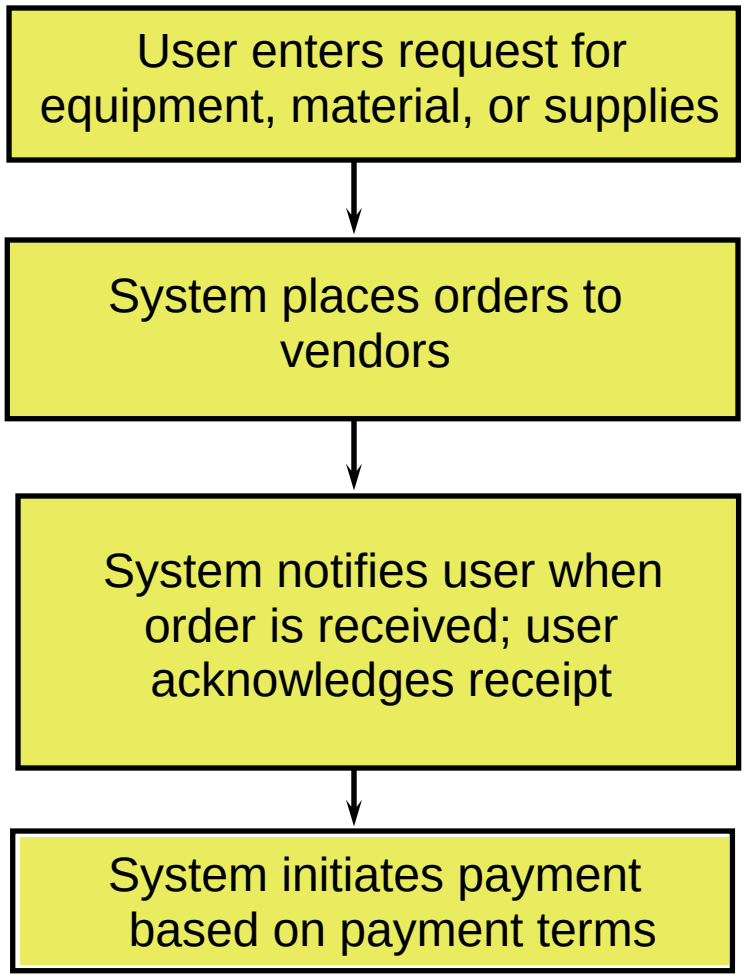
# Understanding Usage: Typical Business Flow

Step 1 – Requisition

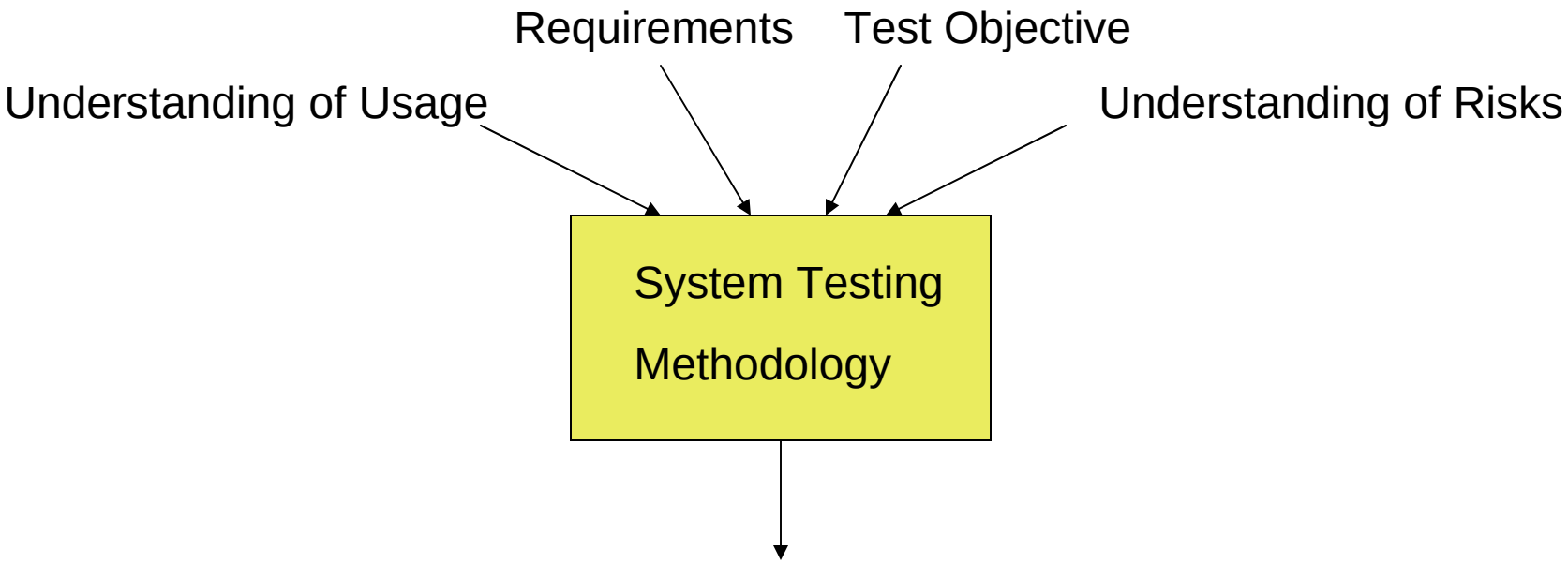
Step 2 - Order Placement

Step 3 - Receipt of Goods

Step 4 - Payment



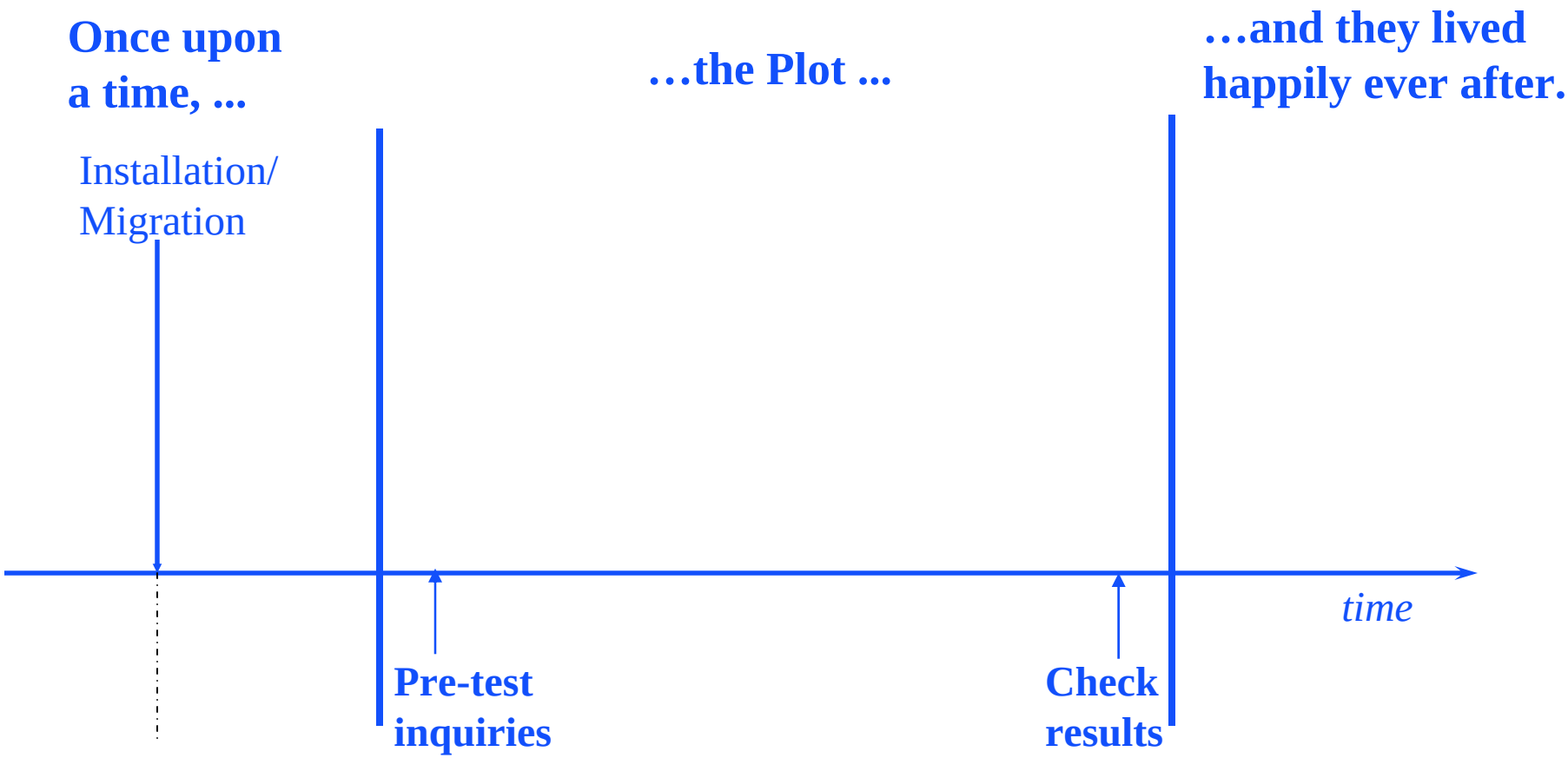
# 2. How do you design System Tests?



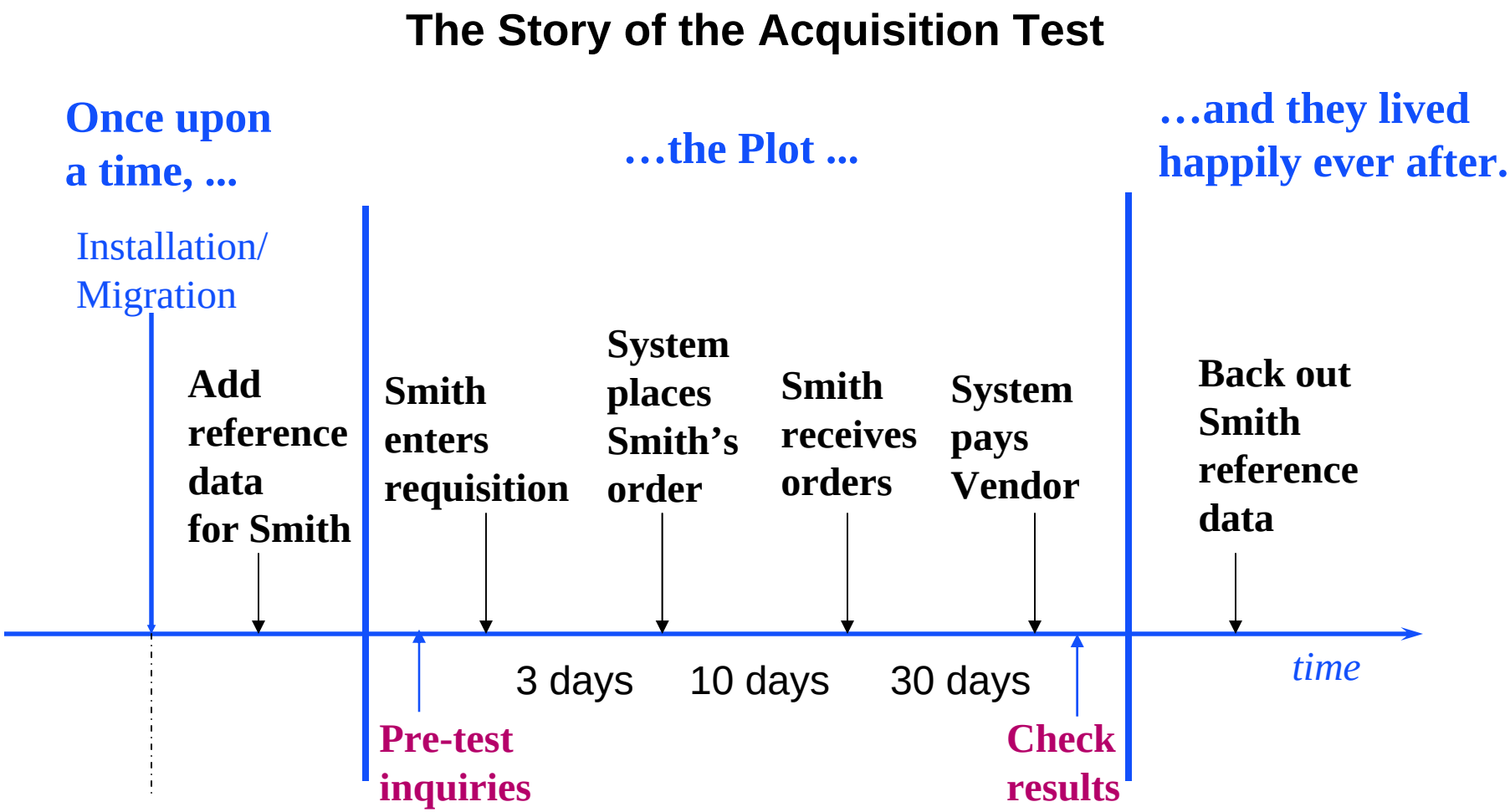

- Overall Design of System Test
- Test Design Documentation

# 2. How do you design System Tests? (continued)

**Template for Designing the Story of the Test**



# 2. How do you design System Tests? (continued)

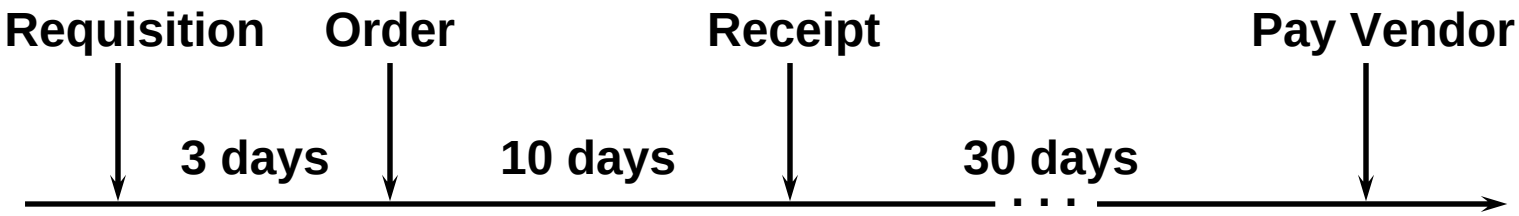


# 3. How do you make a test repeatable?

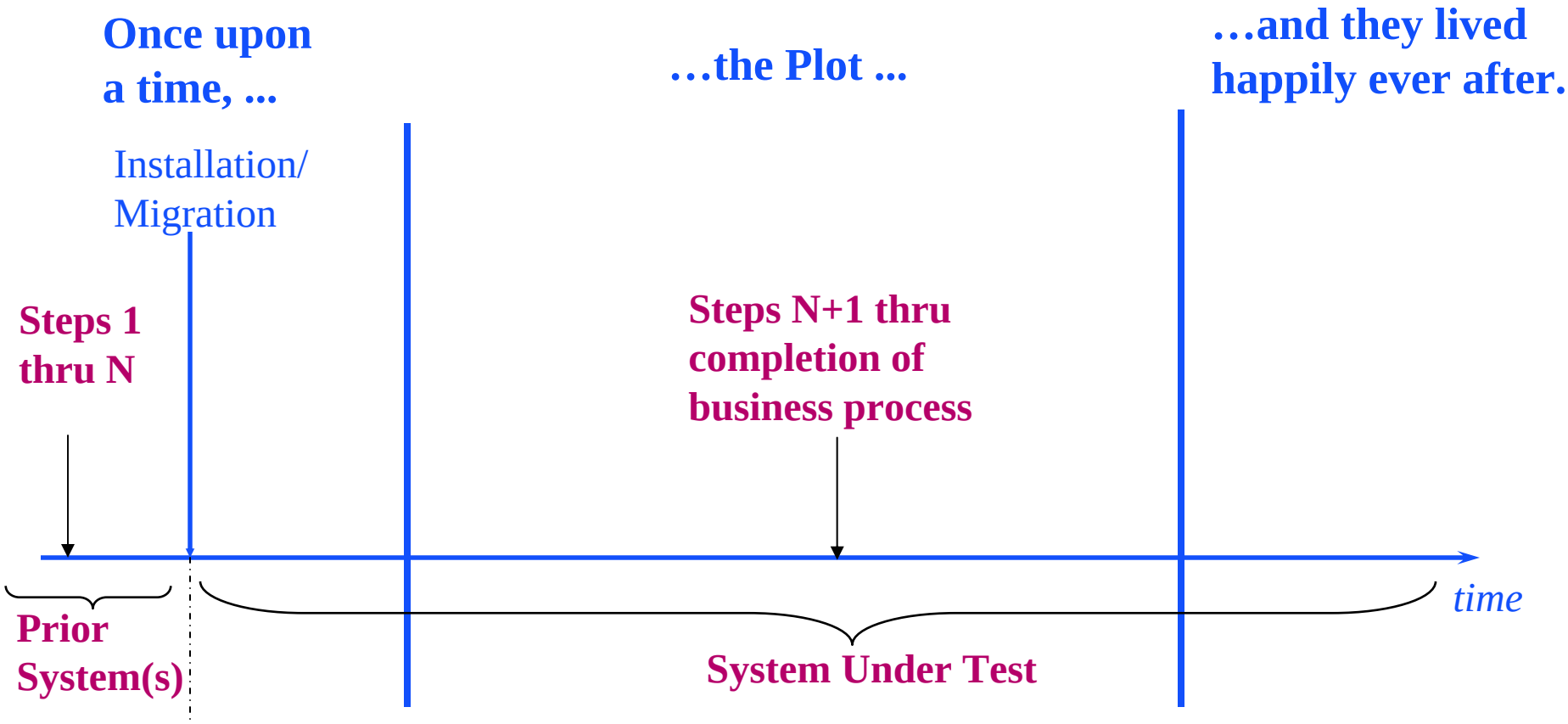
- Why is repeatability important in System Testing?
- Obstacles to repeatability
  - Remnants Problem
  - Common Sandbox Problem
  - Self-Competition Problem

# 4. How do you solve cycle acceleration problems?

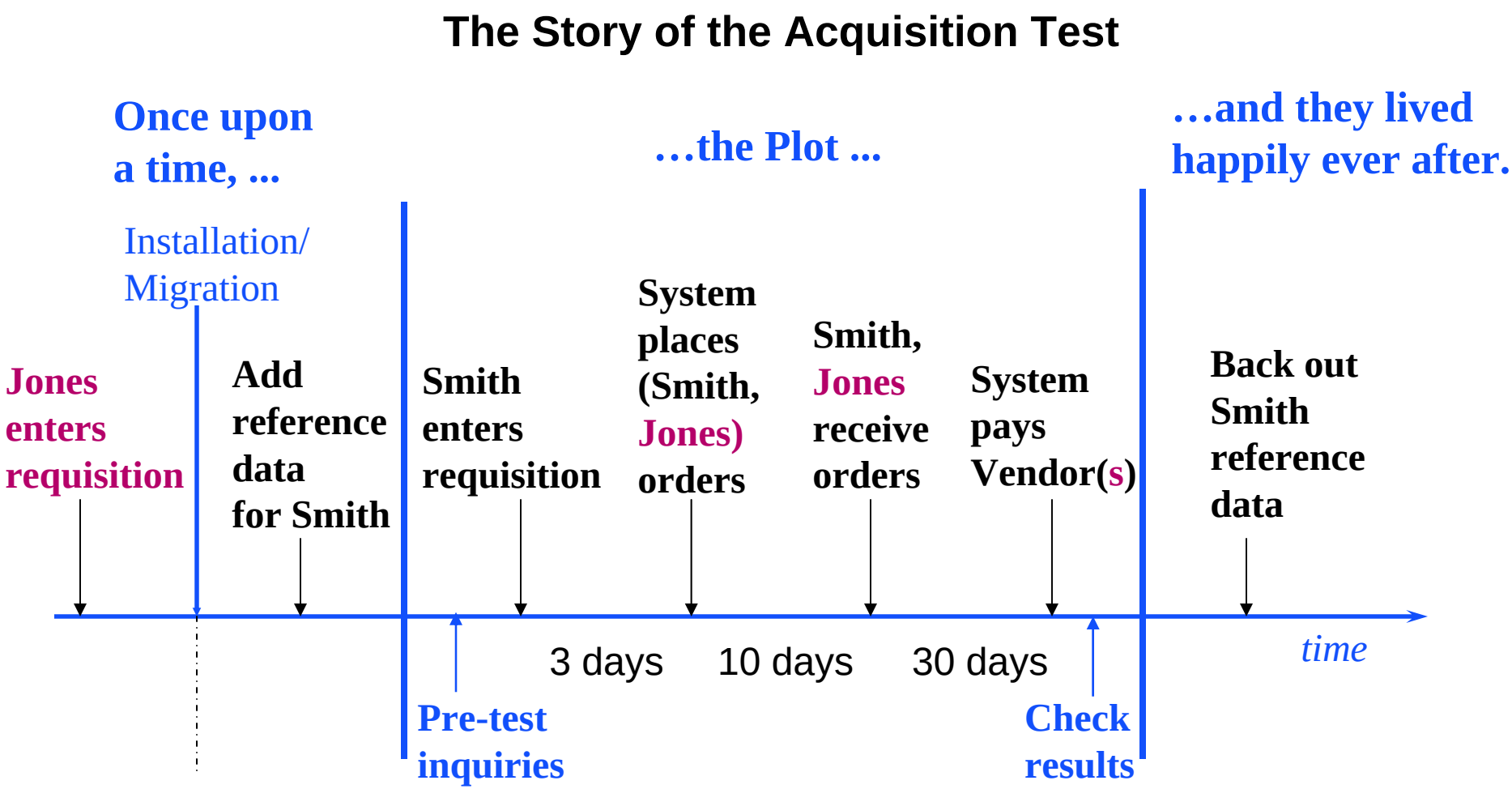
**Typical Chronology for Acquisition Process Business Flow**




# 5. How do you test transactions-in-progress?



# 5. How do you test transactions-in-progress? (cont.)



# 6. What should you measure?

$$\text{System Test Team Effectiveness} = \frac{\text{Defects Avoided X 100}}{\text{Defects Avoided + Defects Missed}}$$

But collection of measurements is a full life cycle activity

# The Formula for Success in System Testing

**System Testing Success =**

**(Technical Excellence)**

**+**

**(Nurturing Front-Loaded Quality)**

***where:***

**(Technical Excellence) >0**

**(Nurturing Front-Loaded Quality) >0**