



**Cognizant**

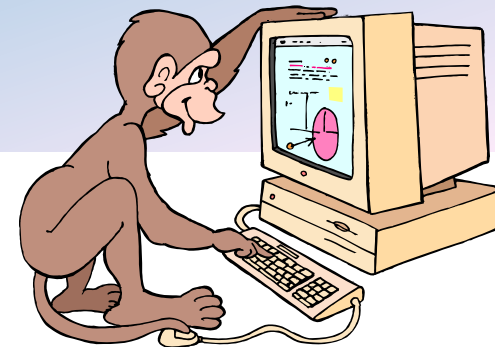
Passion for building stronger businesses

# Adventures with Test Monkeys

**John Fodeh**

**Cognizant Technology Solutions**

**john.fodeh@cognizant.com**



SAST 15-årsjubileum



**Cognizant** | **Testing Services**  
Passion for building stronger businesses



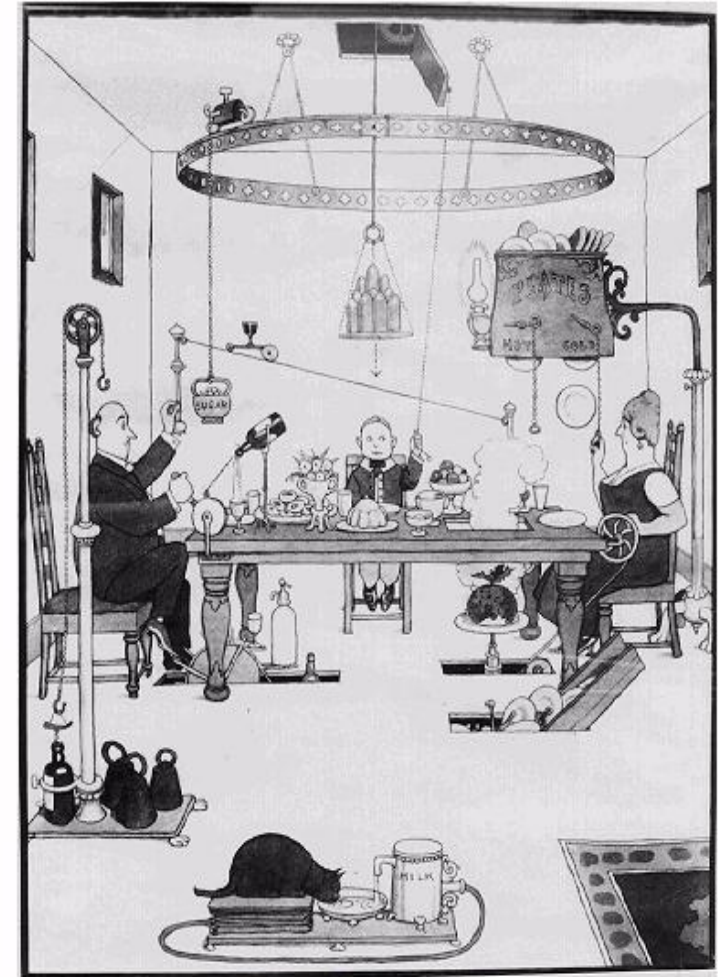
# Contents



- Traditional Test Automation
- Test Monkeys
- Advanced Test Monkeys
- Added Value

# The Promise of Test Automation

- Replacing repetitive and tedious manual testing
- Ensuring the consistency and repeatability of tests
- Performing tests that are difficult to run manually
- Accelerating test execution



A Vision of Automation

\* Image source: Rosemont Engineering

# Traditional Test Automation



## **Typically**

Automation of regression tests

## **Purpose**

Verifying the behaviour of the application under test across multiple builds

# Characteristics of Traditional Test Automation (1/2)

- Static
  - Same sequence of actions executed each time
  - Same code coverage obtained
- Simple
  - Short sequences of actions and verifications
  - Return to known start state after test
  - Clean up after test

# Characteristics of Traditional Test Automation (2/2)

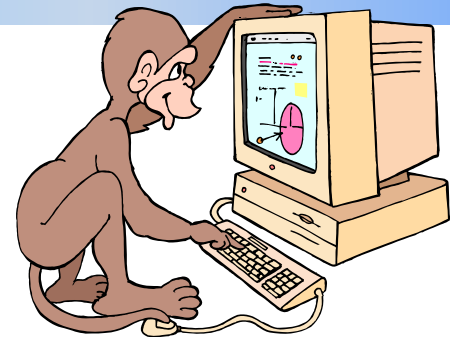
- Synchronized
  - Test execution halted while waiting for “Visual Cues” in the application
- Vulnerable
  - Affected by changes in application and requirements
  - A reasonable level of stability in the application is needed
  - Costly to maintain

# Monkey Testing



**Monkey Testing refers to the process of randomly exercising a software program by means of an automated test tool**

# Test Monkey

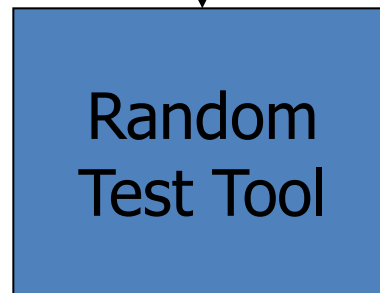


## Action List

```
#script file: Test4.asc  
KEY L;  
KEY 5, LONG;  
CP_KEY FOCUS_UP;  
#KEY BOX_SIZE_UP  
KEY 8;  
TRACKBALL 0, 64;  
KEY SHIFT, R;
```

## Log file

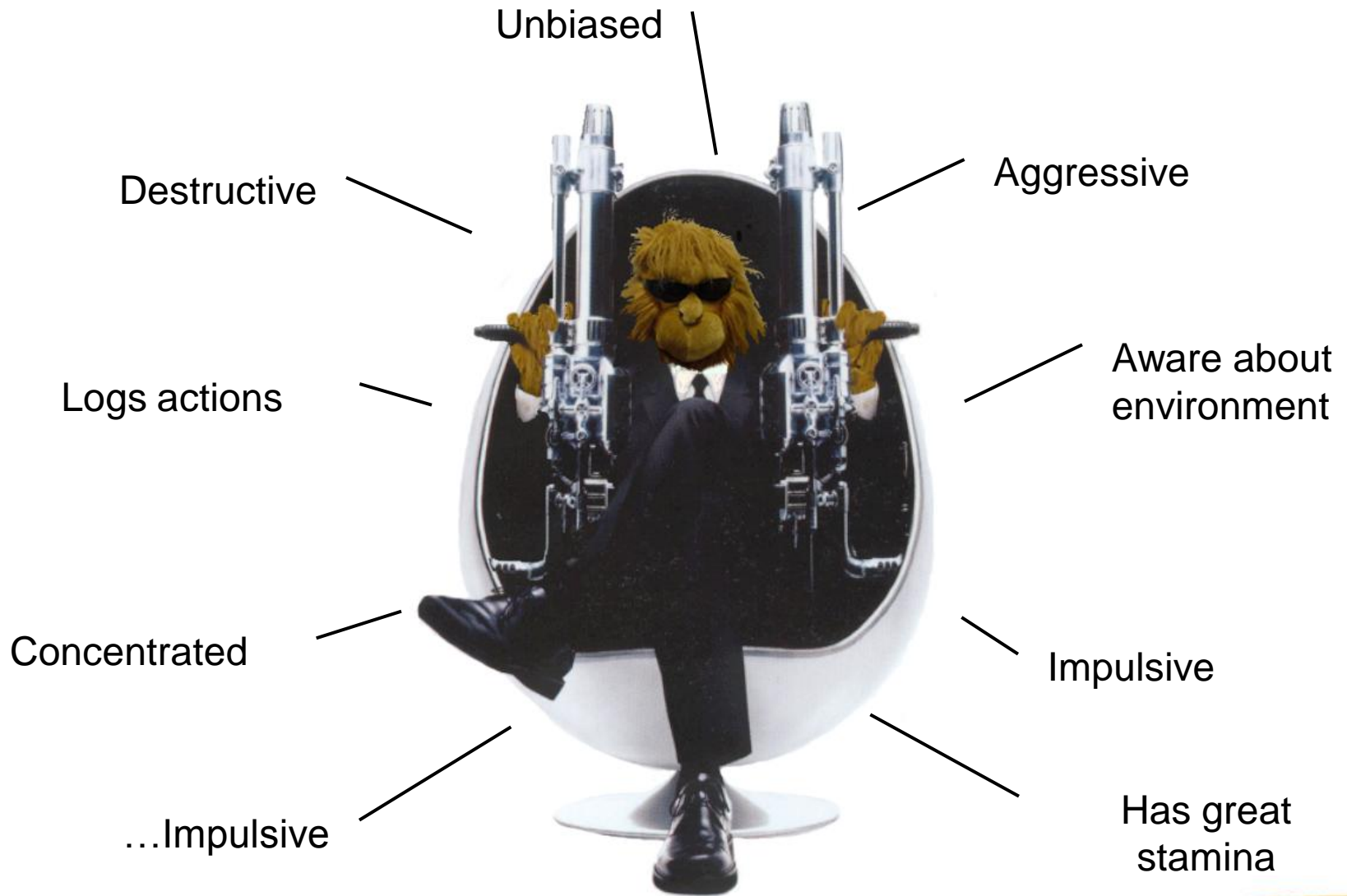
```
#script file: Test4.asc  
CP_KEY FOCUS_UP;  
TRACKBALL 0, 64;  
KEY L;  
KEY 5, LONG;  
KEY 8;  
CP_KEY FOCUS_UP;
```



Application  
Under Test

Utilities





# Basic Features



A test monkey should:

- Select randomly from input range
- Enter input to AUT through user interface
- Detect “life signs” of AUT
- Have robust logging facility. Logs can be replayed to reproduce defects
- (Re-) start and initialize AUT
- Have limited application knowledge but general environment awareness

# Implementing Test Monkeys

- Possible to implement in various environments and platforms
  - Applied to Windows applications through Capture/Replay tools
  - Applied to embedded systems through external interface (e.g. RS232)
- Suitable for
  - Medium to large event-driven systems
  - Complex application environment/settings
  - Systems with many running/interacting applications

# Using Test Monkeys

- Indicator of application reliability
  - Incorporated into daily “smoke” test
  - Establish confidence in application
- System test tool
  - Evaluation of release candidates
  - Previously failed scripts are run again as part of regression tests
- Possible to focus test on specific areas of interest
  - Combinations of sequential and random scripts
  - Random scripts with weight on specific actions

Clear

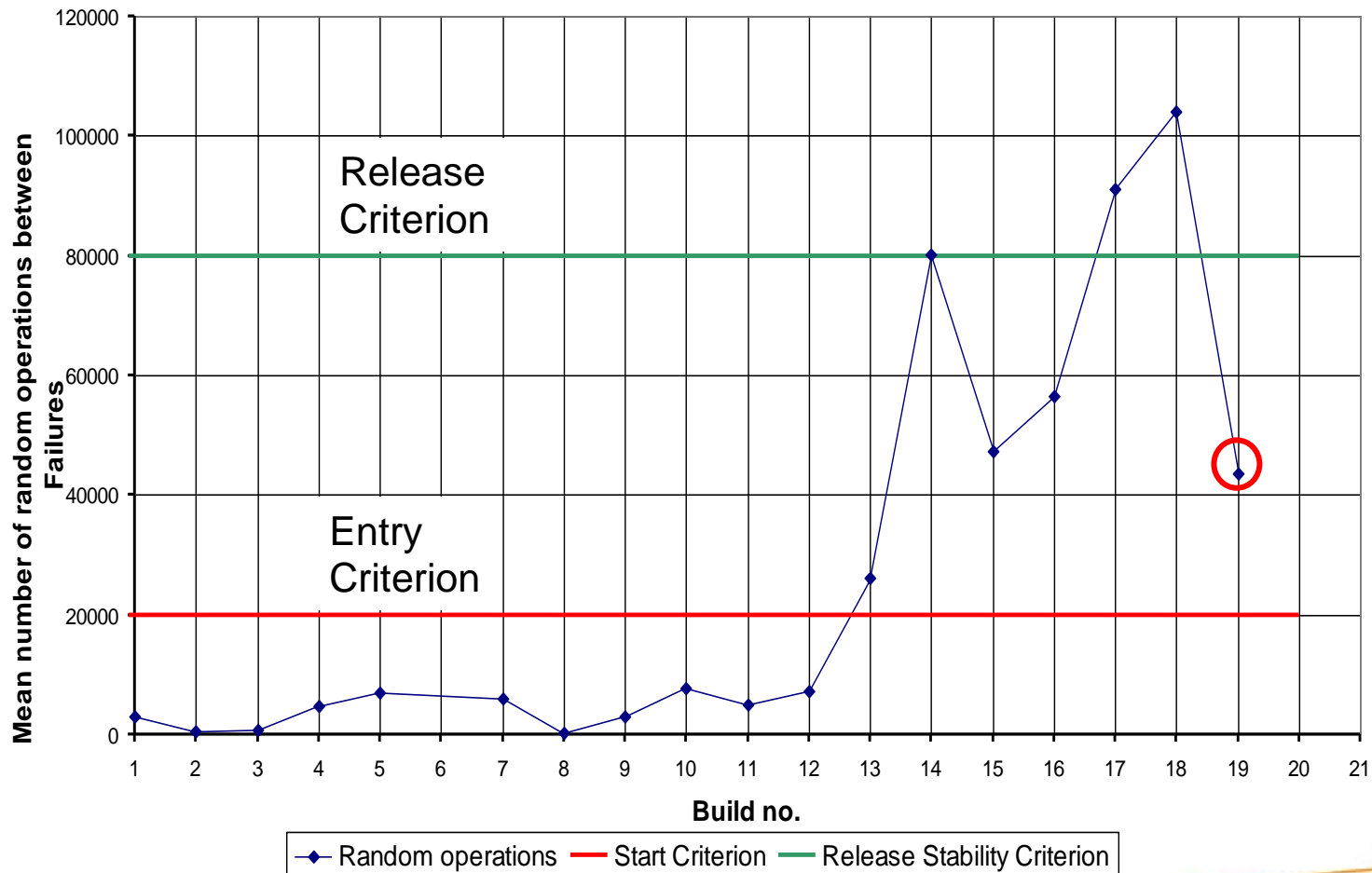
Title

Figures

Text

Generate Slide

# Metrics – Application Reliability



# Added Value (1/2)

- Early testing
  - Does not require a stable application
  - Valuable feedback
- Cost effective
  - Cheap to develop
  - Limited maintenance
  - Finds severe errors
- Negative testing
  - Find combinations and sequences not considered during design



# Added Value (2/2)

- Long and complex test runs
  - Cover wide areas of the system
  - Detect initialization problems and memory/resource leaks
  - Compound chained situations
- Stability indicator
  - Entry criterion for functional testing
  - Evaluate release readiness





# Limitations



- Miss obvious defects
- Do not emulate real use situations
- Long tests runs can be difficult to debug
- For a reliable statistical basis, long and multiple test runs are needed

Monkey Testing is a supplement  
to your manual and automated  
testing - Not a replacement!

# Advanced Test Monkeys

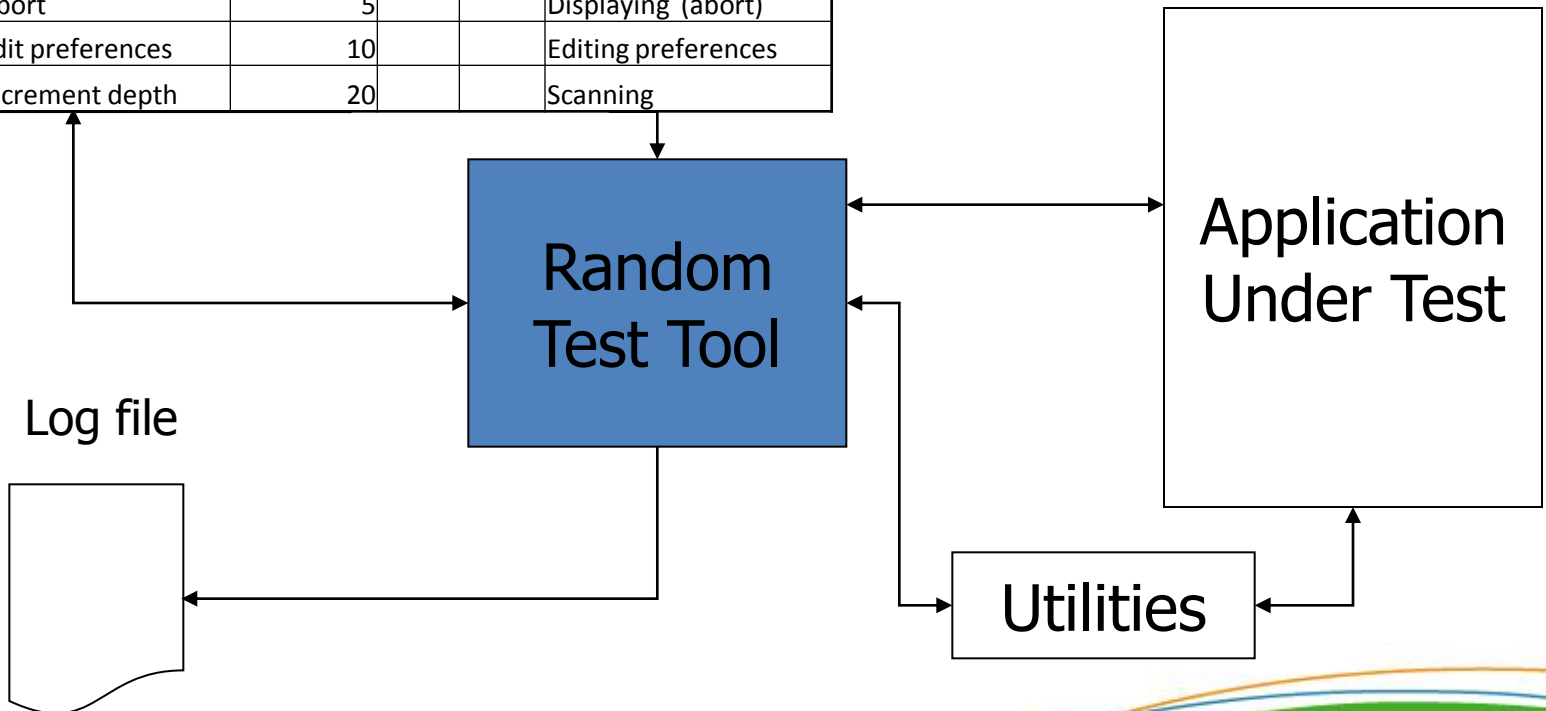
- Wide application knowledge
  - Application modeled in state table
  - Illegal input can be used to test error handling and recovery
  - Emulate real and complex use scenarios
- Effective in finding defects
- Higher development and maintenance cost



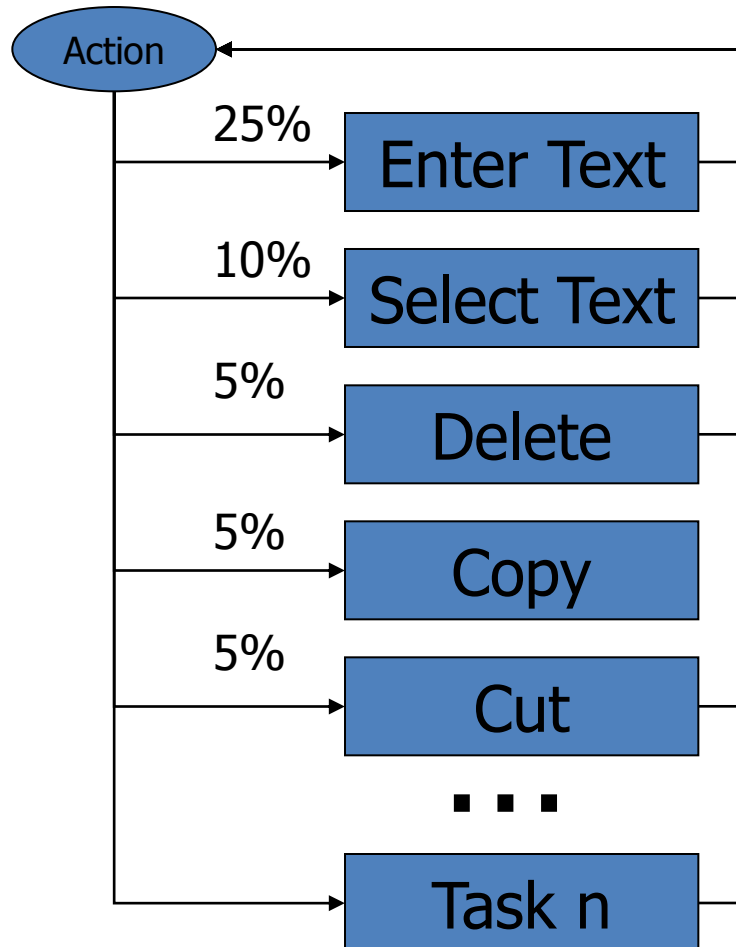
# Test Monkey

State table

Start state	Action	Probability	Input range	End state
Scanning	Adjust frequency	45	3	20 Scanning
	Adjust brightness	20	1	100 Scanning
	Abort	5		Displaying (abort)
	Edit preferences	10		Editing preferences
	Increment depth	20		Scanning



# Intelligent Randomness

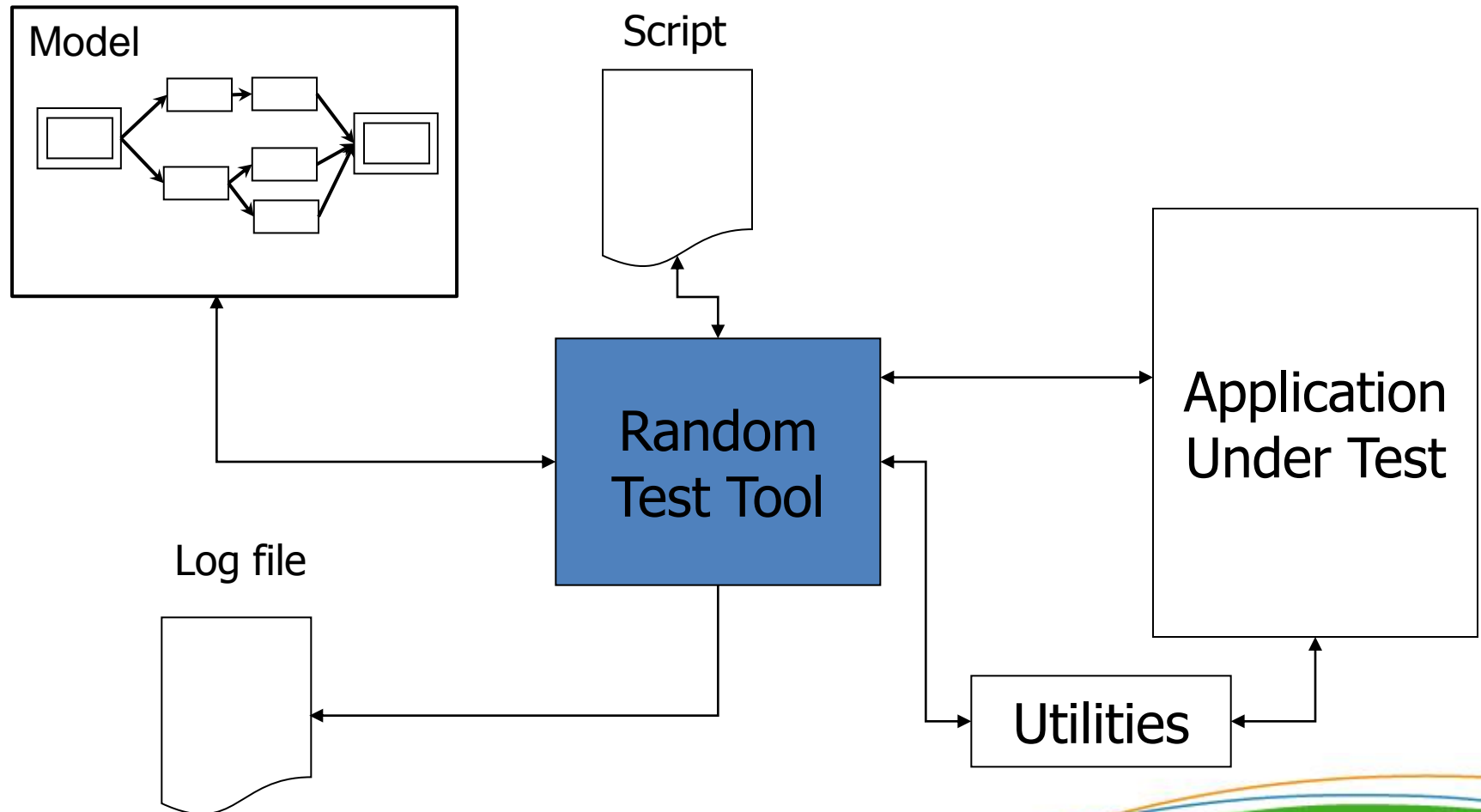


## Probability Tables

- Focus on selected areas
- Weight on specific use scenarios
- Continuous expansion and improvement
- Possibility to exclude certain parts

\* source: Whitmill, Kelly

# Next Step – Model-Based Test Monkeys



# Summary

Test monkeys add great value:

- Cover gaps in traditional automated tests
- Perform more testing in the available time
- Cost effective if utilized correctly

Boost your testing -  
unleash your test monkeys



