

# Software Testing for Continuous Delivery

Seminar 13: UBST & OPs

Dr. Byron J. Williams

October 16, 2019

# What is Site (system) Reliability Engineering?

- Assigned Reading:  
“What is Site Reliability Engineering?”
  - <https://landing.google.com/sre/interview/ben-treynor.html>

“Fundamentally, it’s what happens when you ask a software engineer to design an operations function.”

— Ben Treynor, VP Engineering, Google

# OP (Operational Profile)

- **Definition:** a list of disjoint set of operations and their associated probabilities of occurrence
- A quantitative way characterization of the way a software system is or will be used
- Obtained via measurement, survey, & expert opinion
- Operations: multiple possible test cases or multiple runs
  - Each operation corresponds to an individual sub-domain in domain partitions, thus representing a whole equivalence class.



John D. Musa (RE)  
Pioneer

## Telemetry

### Enable Crash Reporter

- ☒ Enable crash reports to be sent to a Microsoft online service. This option requires restart to take effect.

### Enable Telemetry

- ☒ Enable usage data and errors to be sent to a Microsoft online service.

Source: IEEE Software, <http://www.computer.org/portal/web/csdl/doi/10.1109/MS.2009.132> & Google Images

\*RECAP\*

CIS 4930/6930

# Comparing UBST with CBT

	UBST / BBT	CBT / WBT
<b>Perspective</b>	external behavior (functional)	internal implementation (structural)
<b>Stopping Criteria</b>	reliability goals	coverage goals
<b>Scale</b>	large software (as a whole)	small objects (looking inside)
<b>Timeline</b>	later (e.g., acceptance testing)	earlier (e.g., unit testing)
<b>Tester</b>	IV&V	developers themselves

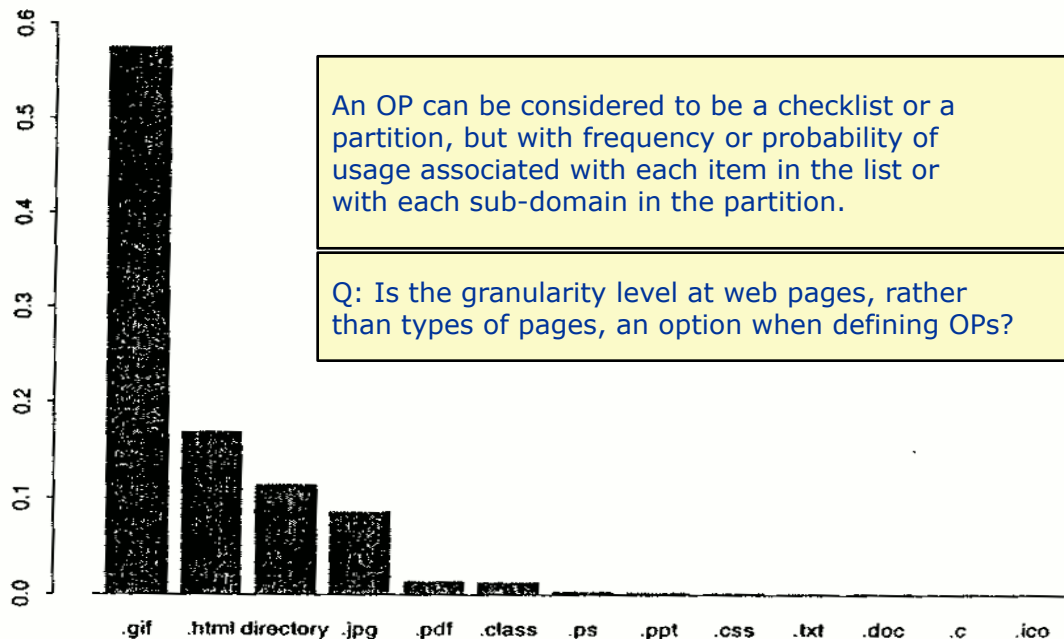
**\*RECAP\***

# OP: Example

- The usage of [www.seas.smu.edu](http://www.seas.smu.edu) (School of Eng. And Applied Science at Southern Methodist Univ.) over 26 days
  - ↳ Operations: requests (hits)
  - ↳ Probability: file types & % of total

File type	Hits	% of total	File type	Hits	% of total
.gif	438536	57.47%	.css	2008	0.26%
.html	128869	16.89%	.txt	1597	0.21%
directory	87067	11.41%	.doc	1567	0.21%
.jpg	65876	8.63%	.c	1254	0.16%
.pdf	10784	1.41%	.ico	849	0.11%
.class	10055	1.32%			
.ps	2737	0.36%	<b>Cumulative</b>	<b>753709</b>	<b>98.78%</b>
.ppt	2510	0.33%	<b>Total</b>	<b>763021</b>	<b>100%</b>

# OP: Graphical Form Example



**Figure 8.1** An operational profile (OP) of requested file types for the SMU/SEAS web site

# OP: Basic Concepts

- Profiles: Disjoint alternatives and their associated probabilities
  - Occurrence or weighting factors
  - Representation: graphs and/or tables
  - Often sorted in decreasing probabilities
- Principle:
  - **The number of test runs for each operation in the OP is proportional to its probability**
- General observations
  - Uneven distribution: basis for UBST (otherwise uniform sampling adequate)
  - Very low probability of usage → cutoff threshold

# OP Usage

- Usage of OP in UBST
  - Pure random sampling rare
    - Requires dynamic (on-the-fly) decisions; Might interfere with system functions (because the system is under constant usage monitoring → significant overhead)
  - More often: pre-prepared test cases → “pseudo” randomness
  - Other variations
    - Progressive testing: lower threshold step by step
- OP and SRE (Software / Site Reliability Engineering)
  - SRE assumes OP-based UBST (availability)
  - OP sometimes directly used in reliability evaluations and improvement



# UBST: General Issues

- General steps
  - Information collection
    - Usage scenarios, usage frequency, etc.
  - OP construction
  - UBST under OP
  - Analysis (reliability!) and follow-up (feedback)
- Linkage to development process
  - Construction: Requirements/specification and in later phases
  - Usage: Testing techniques and SRE

# UBST Benefits

## ■ Primary benefits

- Overall reliability management
- Focus on high leverage parts → productivity and schedule gains:
  - Same effort on most-used parts
  - Reduced effort on lesser-used parts
- Reduction of 56% system testing cost or 11.5% overall cost (Musa study)

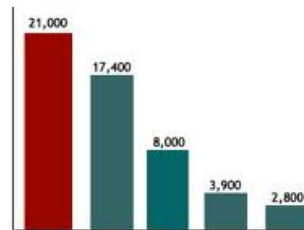
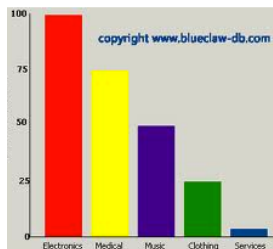
## ■ Gains vs. savings situations

- Savings situations
  - Reliability goal within reach
  - Not to over test lesser-used parts
- Gains situation: more typical
  - Re-focusing testing effort
  - Constrained reliability maximization

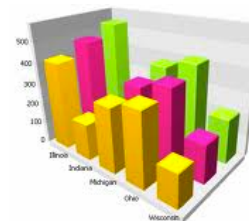
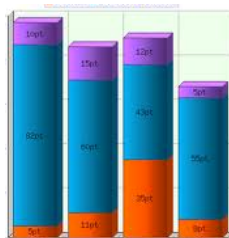
# UBST: Other Benefits

- Introducing new product
  - Highly-used features quickly
  - Lesser-used: subsequent releases
- Better communications/customer relations
  - Customer perspective & involvement → closer ties to customers
  - More precise requirements/specification
  - After all, UBST
- High return on investment (Musa AT&T study)
  - OP cost for an “average” product: ~1 staff-month
    - Defining “average”: 10 developers, 100KLOC, 18 months
    - Sub-linear increase for larger ones
  - Cost-benefit ratio is reported to be about 1:10

# OP: Development



- One OP or more than one?



One OP for each stakeholder role (homogeneous group), e.g., customer, user, DBA, etc.

# OP: Development (Cont'd)

## ■ Approaches

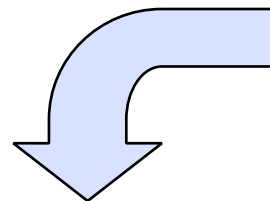
- Measurement
  - Accurate can be costly; limitations for new products
- Customer survey
  - Less accurate and costly; sampling challenges over the customer population
- Expert opinion
  - Least accurate and costly; a rough starting point

## ■ Sample OP developments

- Top-down/Musa-1
  - Customer profiles (end-users, DBA, etc.) & functional profiles
  - OP (combining customer & functional profiles)
- Musa-2
  - Operational mode
  - Graphical (joint probability = product of indiv. prob.)

▷ Musa-1: customer → user → sys. modes  
→ functional → operational

Customer Type	Weight
corporation	0.5
government	0.4
education	0.05
other	0.05



## OP Development: Musa-1

User Type	User Profiles by Customer Type					Overall User Profile
	ctype	com	gov	edu	etc	
	weight	0.5	0.4	0.05	0.05	
End user		0.8	0.9	0.9	0.7	0.84
DbA		0.02	0.02	0.02	0.02	0.02
Programmer		0.18	--	--	0.28	0.104
Third party		--	0.08	0.08	--	0.036

▷ Musa-2: initiator → representation → list → rate → probability

## OP Development: Musa-2

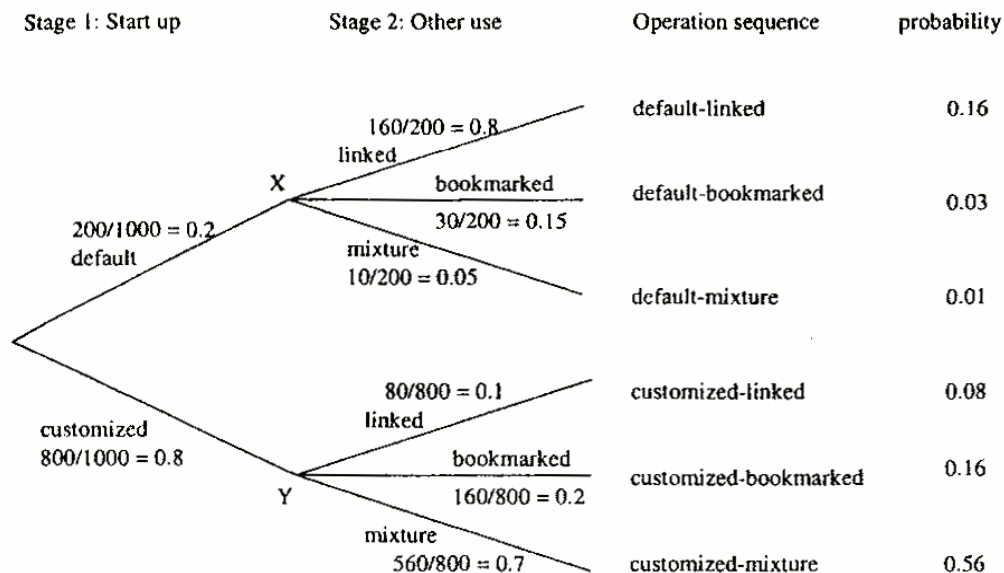


Figure 8.2 A tree-structured or graphical operational profile

# Test Management

## Scenario

Total budget for testing = H hours

# tests during that time = T

Time each test takes = t

% of tests that reveal faults = f

Time required to remove a fault = r

## Estimation:

$$T * t + (f * T * r) = H$$

$$T [t + (f * r)] = H$$

$$T = H / (t + f * r)$$

## Example:

Total budget = 1000 hours

# tests: T = ?

Time for a test = 1 hour

% tests with faults = 20

Fault removal time = 2 hours

# of tests within the budget:

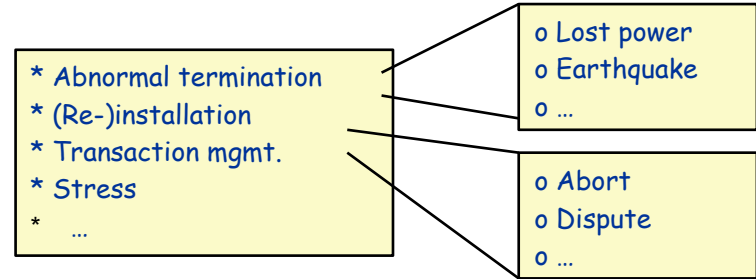
$$\begin{aligned} T &= 1000 / (1 + 0.2 * 2) \\ &= 714 \text{ tests} \end{aligned}$$

How would you manage the (714) tests by using CBT, PBT, and UBST method?

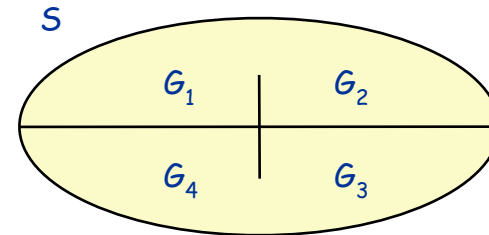


## Test Management (Cont'd)

- How would you manage the tests using checklist method (CBT)?



- How would you manage the tests using partition-based method (PBT)?



- How would you manage the tests using usage-based statistical method (UBST)?

# Test Management (Cont'd)

Suppose I have to manage 714 tests...

File type	Probability	Tests
.gif	0.57	407
.html	0.17	122
directory	0.11	86
.jpg	0.09	70
.pdf	0.01	7
.class	0.01	7
.ps	0.004	3
.ppt	0.003	3

File type	Probability	Tests
.css	0.003	2
.txt	0.002	2
.doc	0.002	2
.c	0.002	2
.ico	0.001	1
Cumulative	0.977	
Total	1.00	

The number of test runs for each operation in the OP is proportional to its probability.