LEVERAGING ACADEMIC-INDUSTRY COLLABORATION FOR SOFTWARE TESTING INNOVATIONS & THEORY BUILDING

Jasbir S. Dhaliwal
Systems Testing Excellence Program
FedEx Institute of Technology
University of Memphis

First STEPs

- 25 years ago: “A Framework for the Validation of Knowledge Based Systems” – Nuclear Power Stations (the Homer Simpson Scenario)

- 2005: FedEx CIO – complex, global, real-time environments with high business impact and skills shortage

- Need for Theory to Guide Best Practices in Industry: “who do I listen to?”

- Need for Integrative Perspectives (ISSRE, ICST, ……) within academia and across the subdisciplinary and practitioner divides

Dark Art?: Practitioner Considerations in Testing

- Should we outsource development or testing – which is more critical as a core competency?
- How do we get our outsourced testing vendor to innovate so that we remain on the cutting edge?
- What tools should we buy for automated testing?
- How do we test the quality of program designs?
- Are our test plans adequate- we keep missing critical defects?
- How do we undertake a systematic assessment of our testing methodology?
- The CIO wants a ROI computation for investments in testing?
- How do I get my developers and testers to get along?
- What exactly does a tester do in agile scrums?
- How long should a sprint be?

Illuminating Science?: Considerations of Research

- Test instrumentation and pattern matching for automatic failure identification
- A fitness function to find feasible sequences of method calls for evolutionary testing of object-oriented programs
- On combining multi-formalism knowledge to select models for model transformation testing
- A JML compiler based on AspectJ
- Quality of automatically generated test cases based on OCL expressions
- Statistical sampling based approach to alleviate log replay testing

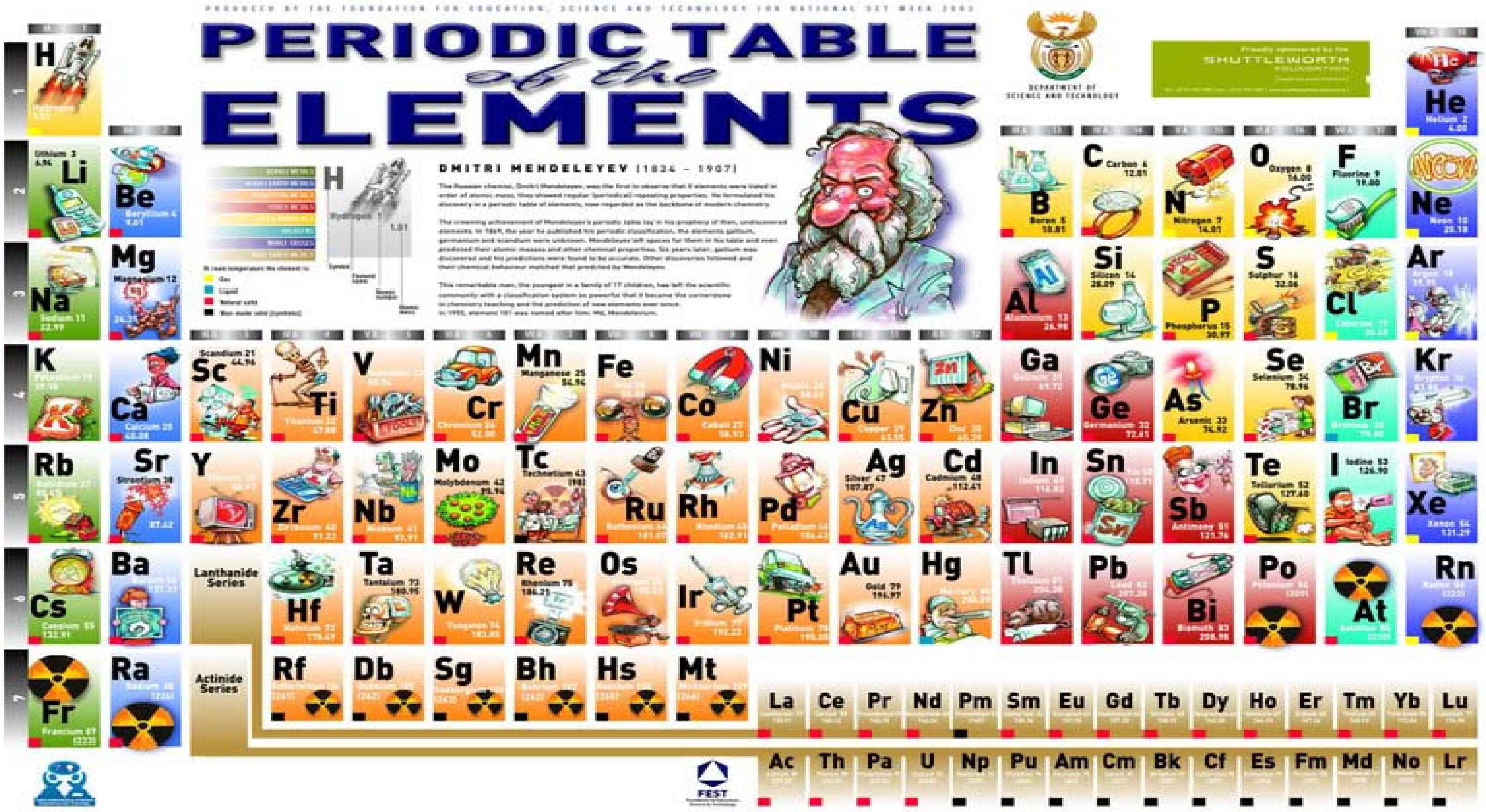
SCIENCE- ART GAP!

- Unit of analysis problem: micro versus macro
- Balance between BASIC science versus artful APPLICATION
- Scientific funding agencies may not fully appreciate pragmatic requirements (NSF, NSERC versus SSHRC)
- Chasing the next big thing for grants – is this good for science?
- Wisdom from practitioner conferences largely based on sample-size-of-one personal experiences (ART) not deep empirical analysis
- Wisdom from scientific conferences largely based on mathematically complete and **elegant** theoretical models and precise micro measurements (SCIENCE)
- We certainly need both and need them to work together!
- Art and/or Science: both require a strong theory.

Big Picture Theory: Types of Testing

- Functional Testing
- Stress Testing
- Dark Testing
- White/Black/Grey Testing
- Regulatory Testing
- Risk-based Testing
- Exploratory Testing
- Infrastructure Testing
- Database Testing
- Requirements Testing
- Alpha/Beta Testing
- Configuration Testing
- Coverage Testing
- Load Testing
- Boundary Testing
- Localization Testing
- Unit Testing
- Integration Testing
- Systems Testing
- Performance Testing
- User Acceptance Testing
- *Please add …… 143??*

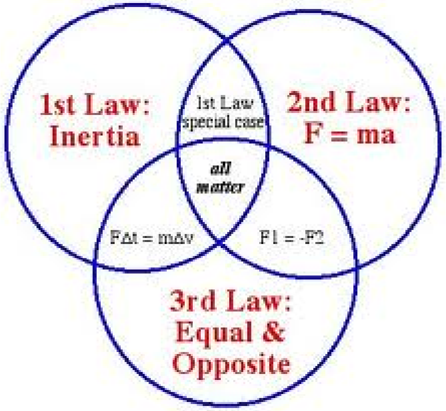
Elements of Test Types


PERIODIC TABLE of the ELEMENTS
DMITRI MENDELEYEV (1834 - 1907)
Lanthanide Series
Actinide Series
FEST

What about Biology and Physics

- Darwinian Evolution is at the core of Biology: Do our V+V and maturity models (TMMi??) suffice?

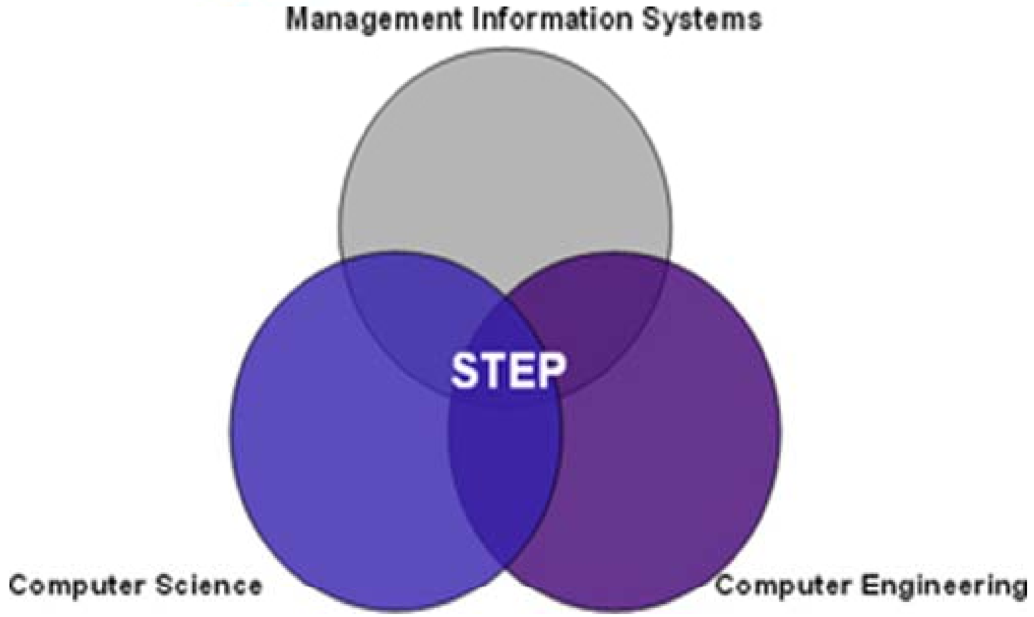
- Laws of Physics: Newtonian Principles



Newton's Laws

- What are the core principles of T&E that can be universally applied?

Systems Testing Excellence Program



A Partnership for Advancing the Science of Testing:

an interdisciplinary collaboration of Computer Science, Information Systems, and Computer Engineering scholars and deep practitioners: what is missing in our literature?

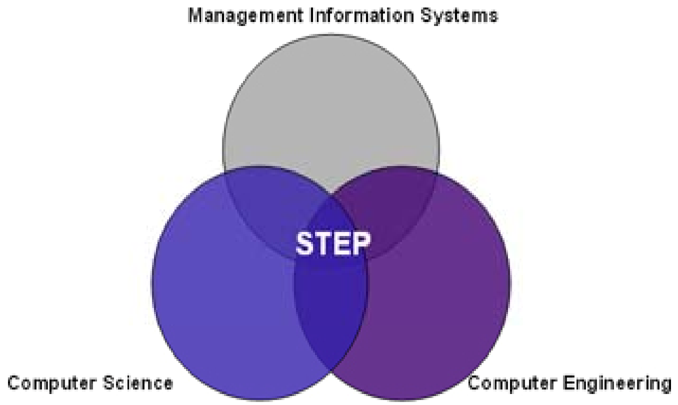
FedEx
INSTITUTE OF
TECHNOLOGY
THE UNIVERSITY OF
MEMPHIS

Systems Testing Excellence Program

FedEx Institute of Technology
The University of Memphis

Advancing the Science of Testing

System Testing Excellence Program



FedEx®

DISA
A Combat Support Agency

SYNTEL
Consider IT Done®

Since 1884
DeHart Capital, LLC

Mphasis
an HP company

CISCO

NRI 未来創発
Dream up the future.

Cook Systems International Incorporated

AutoZone®

Thomas&Betts

Is Testing an Art or a Science?

- What attributes or aspects make it a science?
- What attributes or aspects make it an art?
- Measurement, control, causality, rationality, holistic, decomposition, structure ………
- Positive tension between the **inductive** and **deductive** impulses of our discipline
- Lets look at definitions!

Evolution in the Definition of Testing

- “Testing is the process of executing a program or system with the intent of finding errors”: *Myers (1979) in Art of Systems Testing*

- “Testing is *any activity* aimed at evaluating an attribute of a program or system. Testing is the *measurement* of software quality”: *Hetzel (1983) in Complete Guide to Software Testing*

- “Testing is the concurrent lifecycle process of engineering, using, and maintaining testware in order to measure and *improve* the quality of the software being tested”: *(2002) Craig & Jaskiel in Systematic Software Testing*

- *Our role is expanding – will this make us less scientific?*

Standards & Innovation in Testing

- "Innovate! Follow the standard and do it intelligently. That means including what you know needs to be included regardless of what the standard says. It means adding additional levels or organization that make sense": *IEEE Computer Society Software Engineering Standards Collection (as discussed by Craig & Jaskiel (2006)*

- Food for thought: Does this suggest a contingency approach to a science of test and evaluation?

- Contingent upon **context**, perspective and organization – art??

- ***Optimal*** selection of test methods and techniques from a comprehensive basket based on contextual characteristics.

- Risk-based testing involves massive judgmental heuristics

Expanding Roles of Testers

- Testers As bug specialists
- Testers As execution watchdogs
- Testers As process analysts
- Testers As documentation experts
- Testers As quality assurors
- Testers As service providers
- Testers As team players/leads
- Testers As certification authorities
- Testers As user representatives
- Testers As automation experts
- Testers As user representatives
- Testers As designers of experiments
- Testers As risk managers
- Testers As master communicators
- Testers As improvement evangelists
- Testers As security analysts
- Testers As localization forces
- Testers As regulatory auditors
- Testers As exploratory adventurers
- Testers As test case writers and optimizers

Will these new roles make us more or less scientific?

Back to the Future in the evolution of software testing

- Little or no testing: White elephant systems - let users do the testing
- SDLC: testing as a final stage by specialist testers –we got compartmentalized but recognized as a profession
- Prototyping: testing by users
- CMMi and quality approaches: testing is a parallel process to development and to be done by independent assessors
- Agile and X-treme methods: testing by developers or quasi-testers
- A science or art of testing has to answer the following questions: What is the right way? Who is best trained to do testing?

Constructs for the Art/Science of Testing

- Validation: Mapping between a representation and its world state - Art
- Verification: Demonstration of consistency, completeness and correctness in relation to an available set of specifications (from a prior stage) - Science
- Reliability: Low degree of measurement error (related to replicability) - Science
- Evaluation: determination of quality of model and its output in relation to known optimal sources and outcomes (who evaluates?) – Science + Art
- Utility: actual and perceived benefits in relation to users and purpose of a system (usefulness) – Science + Art
- Usability: ease of use and acceptance by a user community in relation to human engineering and understandability of results - Science and Art

Theoretical Perspectives for Guiding the Art and Science of Testing

- Rationalism: logical assessment of underlying premises central to testing
- Empiricism: Observation and measurement as absolute forms of validation
- Positive Economics: Predictive behavior assessment is the ultimate form of validation
- Unanism: Inter-subjective agreement is the basis for validation and testing
- Pragmatism: Functional evaluation (does it work?) is the critical criteria
- Representationlism: Mapping between software artifact and its source world state
- Popper: We cannot prove anything we can only negate circumstances

We need to build a pragmatic art of testing that has strong empirical scientific foundations to provide the basis for industry best practices

- Bug finding/fixing and verification to specs have hijacked the true mission of testing - there is a need to refocus our art/science on validation
- Bug finding/fixing and verification to specs can be automated and outsourced away but not validation
- The theoretical heart of a science of testing lies with a focus on validation based on representational theory

STEP Research Project: Reconsidering sequential precedence

- Testing is divided into two distinct stages: Evaluation Testing and Substantive Testing.
- Evaluation testing is done a priori to assess process quality, contextual factors and risk (internal control) of development/data/test cases, etc. (ARTful JUDGEMENTS)
- Substantive testing is done at end using smaller samples to ensure reliability, responsibility, correctness (PRECISE SCIENCE).

STEP Research Project: Efficiency Versus Effectiveness Paradigms of Testing

- Efficiency: ratio of input to output - precise
- Effectiveness: how well goals are met - artful
- Do development and testing share similar efficiency and effectiveness frameworks?
- Need to consider testing within the efficiency paradigm – metrics, metrics, metrics
- Can effectiveness-driven testing be successfully undertaken in an efficiency-driven development environment?

STEP Research Project: Is Systems Theory Adequate As the Basis for a Science of Testing

- A system is an organized whole implying decomposition and organizing logic
- Basis for separating unit testing from integration testing
- As design is decomposition and there are good and bad designs (tight cohesion and loose coupling), are the decompositions of good design also sound for good testing
- Example: normalization in relational database design and blowing “bubbles” in data-flow diagrams
- Timing of decomposition: apriori? If developers are doing unit testing, is a comprehensive modular structure developed beforehand
- Modularization for facilitating development, scalability and maintenance may not be yielding testable systems
- Should we change modularity concepts to yield more **testable** systems?

- Is this a good approach to: How do we test designs?

STEP Research Project: Decoupling for Testing Complex Systems

- Complexity of global systems is making regression testing harder
- Can we re-modularize portions of code for decoupling purposes
- Formalize and compartmentalize business rules or interfaces as a separate component from application code for testing purposes.
- Testing business rules requires different methods and testers *(IEEE Software – July 2012)*

Validation based on representation theory

(more than bug-finding and verification) is at the heart of a theory for testing:

If we make software systems more “natural” - they will be valid.

Our criteria for representational artifacts could be :

- Domain Correspondence
- Range Correspondence
- Event/Action Correspondence
- Operational Correspondence
- Grain Size Correspondence
- Construct/Primitives Correspondence
- Semantic Correspondence (Expressive Power)
- Meta-knowledge correspondence
- Abstraction Level Correspondence

Creating A Literature

- Interpersonal Conflict Judgments between Developers and Testers in Software Development. ***Journal of Database Management***.
- The Business Rules Approach and Its Impact on Software Testing. ***IEEE Software***.
- Alignment within the Software Development Unit: Assessing Structural and Relational Dimensions between Developers and Testers. ***Journal of Strategic Information Systems.***
- Empirical Investigation of Client Managers’ Responsibilities in Managing Offshore Outsourcing of Software Testing Projects. ***IEEE Transactions on Engineering Management.***
- Mitigating Vendor Silence in Offshore Outsourcing to India: An Empirical Investigation of Testing. ***Journal of Management Information Systems***.
- TESTQUAL: Conceptualizing Software Testing as a Service. ***E-Services Journal***.
- Implementing Quality Gates throughout the Enterprise IT Production Process. ***Journal of Information Technology Management***.
- Alignment within the corporate IT unit: An analysis of software testing and development. ***European Journal of Information Systems***.
- Organizing Software Testing for Improved Quality and Satisfaction. ***Journal of Information Technology Management***.
- Client Communication Practices in Managing Relationships with Offshore Vendors of Software Testing Services. ***Communications of the Association for Information Systems.***
- Governance Mechanisms for Software Testing. **Journal of Organizational and End-User Computing**

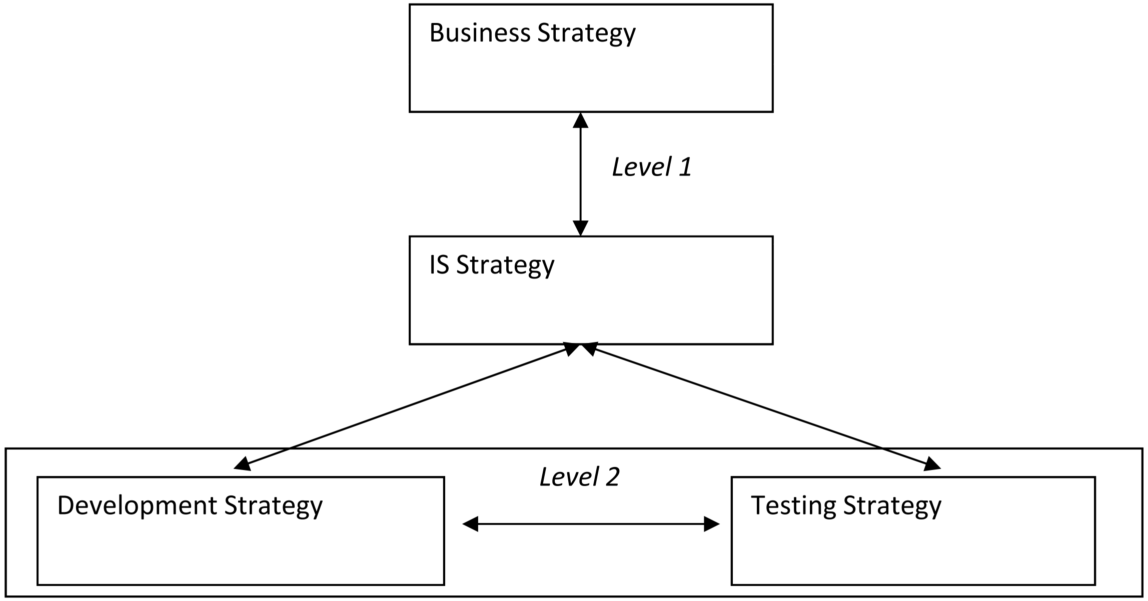
Systems Testing Excellence Program

STEPing Back Reflections

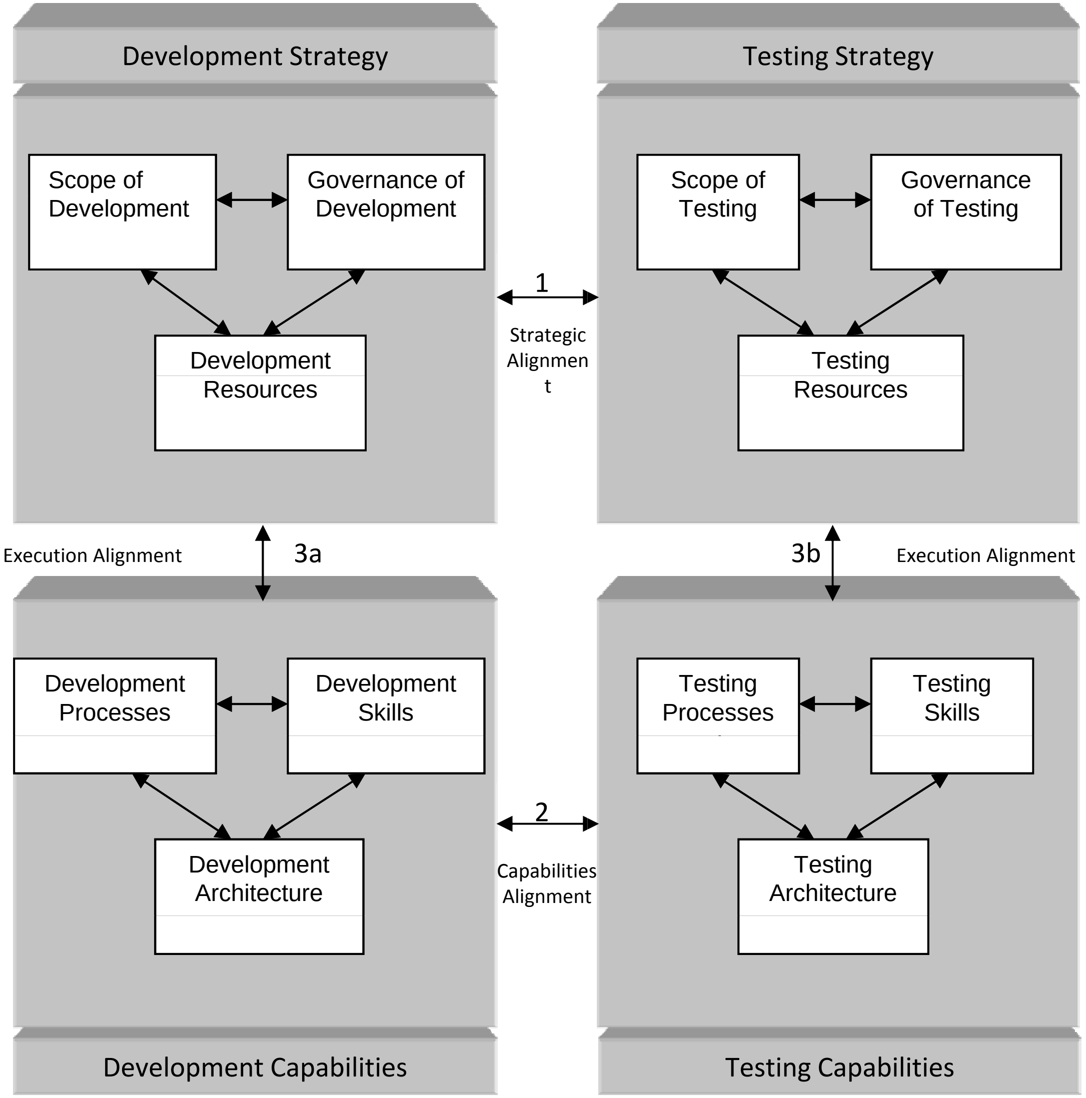
FedEx Institute of Technology Memphis

- Start with real world practical issues
- Use external benchmarks for quality
- Support and emphasize sub-disciplinary collaboration
- Make journal publication mandatory beyond practical recommendations
- Taking academics out is more efficient than bring practitioners in for research – reverse is true for teaching
- Develop diverse collaborations for contextual complexity (FedEx vs Microsoft and DoD vs FedEx scenarios)
- Undergraduate Minor, Graduate Certificate, PhD Concentrations, and Theory Based Industry Training Programs
- Integrative Research and Training Partnering for Longer Term

Levels of Alignment



Alignment model for testing and development



Alignment model for testing and development

Building the Future of Testing

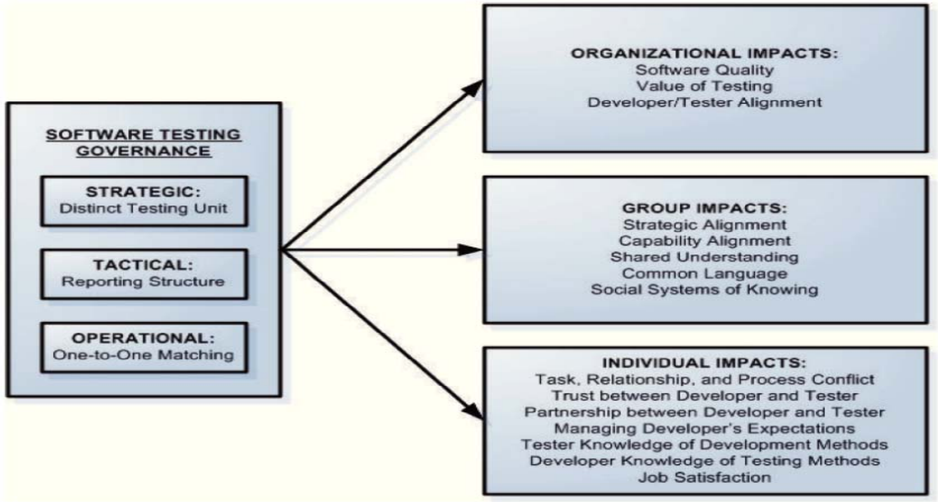

SOFTWARE TESTING GOVERNANCE
STRATEGIC: Distinct Testing Unit
TACTICAL: Reporting Structure
OPERATIONAL: One-to-One Matching
ORGANIZATIONAL IMPACTS: Software Quality
Value of Testing
Developer/Tester Alignment
GROUP IMPACTS: Strategic Alignment
Capability Alignment
Shared Understanding
Common Language
Social Systems of Knowing
INDIVIDUAL IMPACTS: Task, Relationship, and Process Conflict
Trust between Developer and Tester
Partnership between Developer and Tester
Managing Developer's Expectations
Tester Knowledge of Development Methods
Developer Knowledge of Testing Methods
Job Satisfaction

Figure 1. A Theoretical Framework

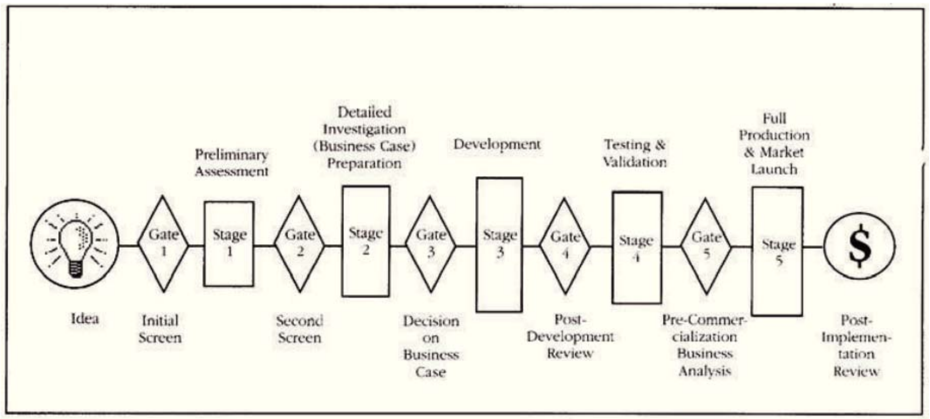
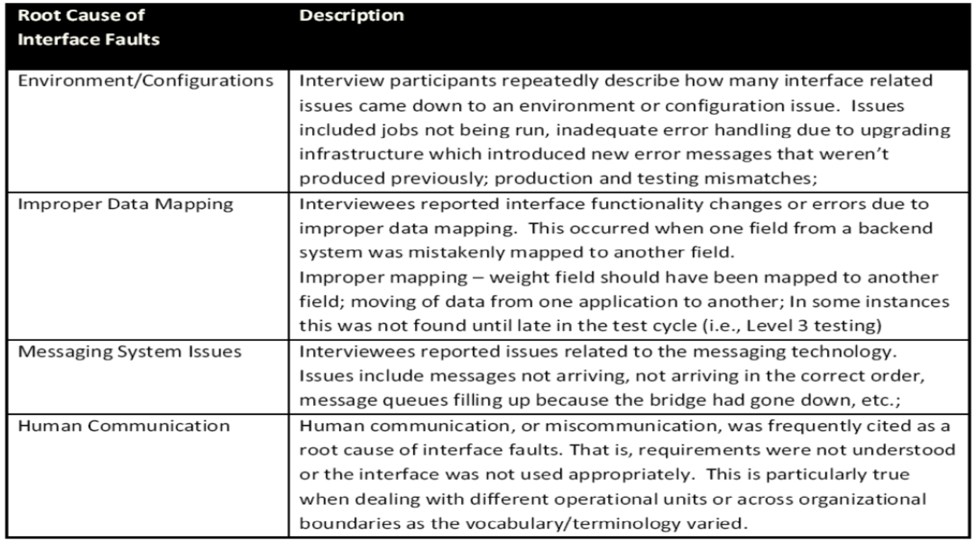


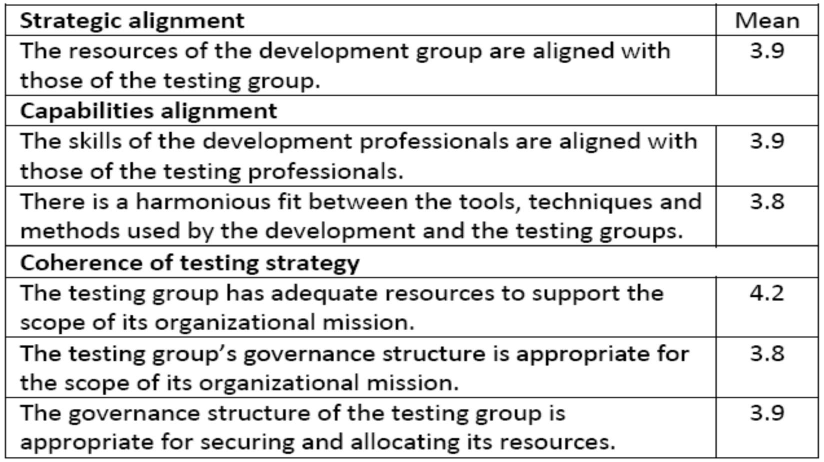
Figure 1 - Overview of a Stage-Gate System

Table 3 – Root Causes of Interface Faults



Root Cause of Interface Faults	Description
Environment/Configurations	Interview participants repeatedly describe how many interface related issues came down to an environment or configuration issue. Issues included jobs not being run, inadequate error handling due to upgrading infrastructure which introduced new error messages that weren't produced previously; production and testing mismatches;
Improper Data Mapping	Interviewees reported interface functionality changes or errors due to improper data mapping. This occurred when one field from a backend system was mistakenly mapped to another field. Improper mapping – weight field should have been mapped to another field; moving of data from one application to another; In some instances this was not found until late in the test cycle (i.e., Level 3 testing)
Messaging System Issues	Interviewees reported issues related to the messaging technology. Issues include messages not arriving, not arriving in the correct order, message queues filling up because the bridge had gone down, etc.;
Human Communication	Human communication, or miscommunication, was frequently cited as a root cause of interface faults. That is, requirements were not understood or the interface was not used appropriately. This is particularly true when dealing with different operational units or across organizational boundaries as the vocabulary/terminology varied.

Table 3: Sources of misalignment



Strategic alignment	Mean
The resources of the development group are aligned with those of the testing group.	3.9
Capabilities alignment	
The skills of the development professionals are aligned with those of the testing professionals.	3.9
There is a harmonious fit between the tools, techniques and methods used by the development and the testing groups.	3.8
Coherence of testing strategy	
The testing group has adequate resources to support the scope of its organizational mission.	4.2
The testing group's governance structure is appropriate for the scope of its organizational mission.	3.8
The governance structure of the testing group is appropriate for securing and allocating its resources.	3.9

Systems Testing Excellence Program

FedEx Institute of Technology
The University of Memphis

Building the Future of Testing

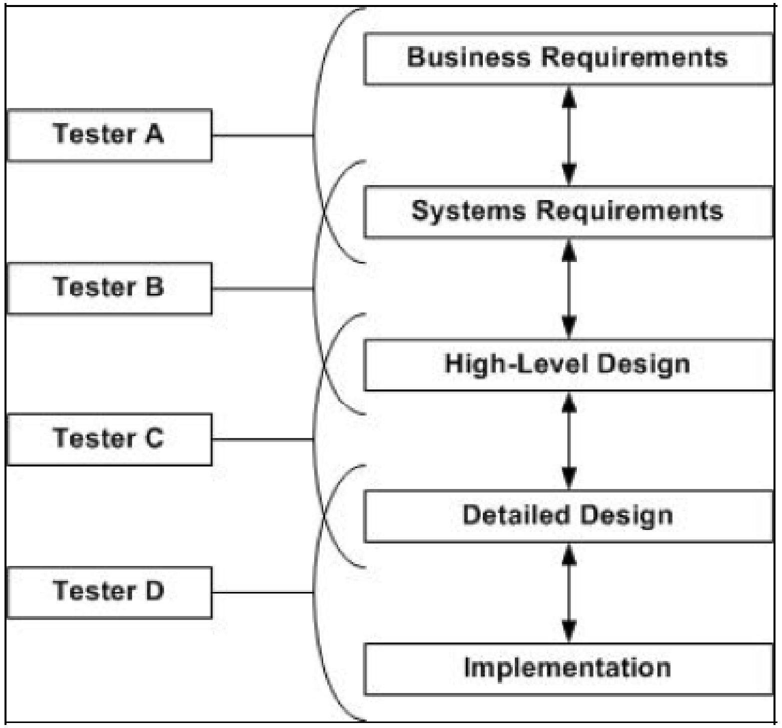


Figure 3. The leapfrog model

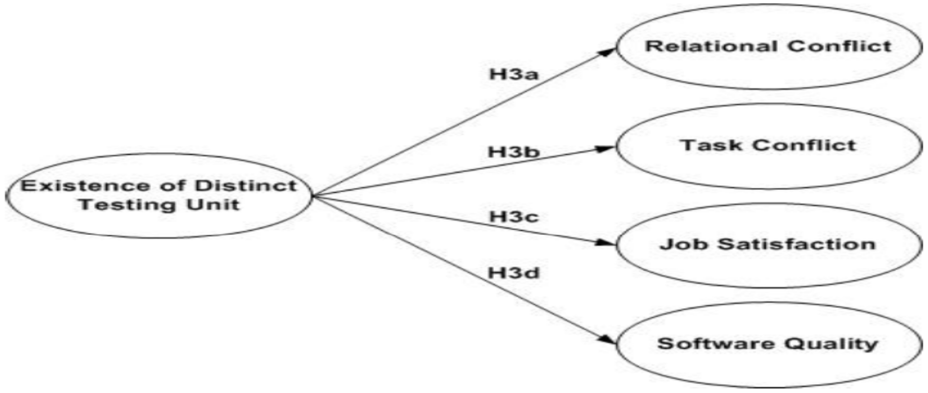
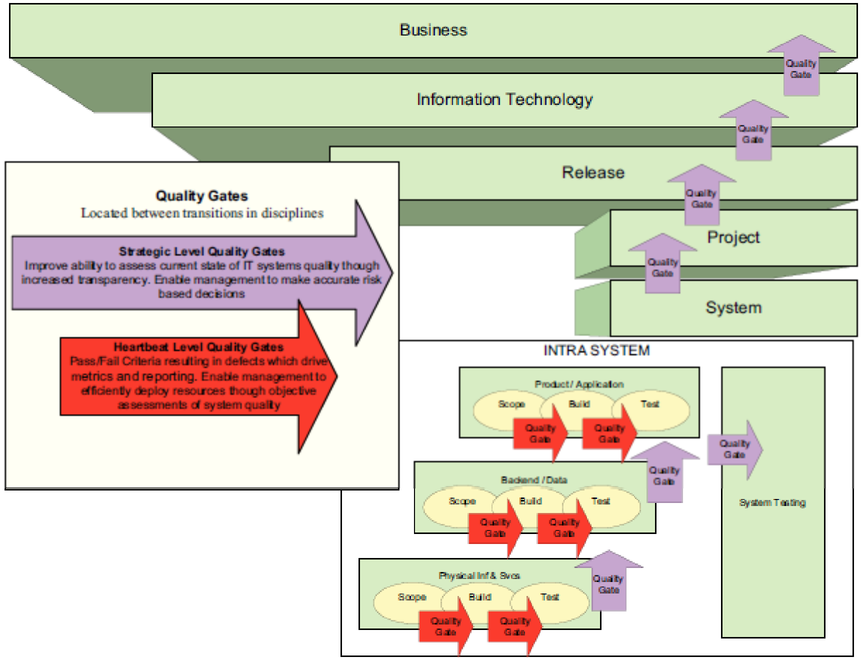


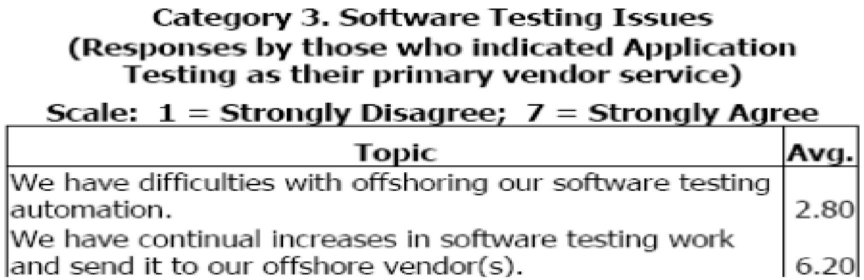
Figure 3. Research Model 3: Impact of the Existence of Distinct Testing Unit



Quality Gates
Located between transitions in disciplines

Strategic Level Quality Gates
Improve ability to assess current state of IT systems quality though increased transparency. Enable management to make accurate risk based decisions

Heartbeat Level Quality Gates
Pass/Fail Criteria resulting in defects which drive metrics and reporting. Enable management to efficiently deploy resources though objective assessments of system quality



Category 3. Software Testing Issues
(Responses by those who indicated Application Testing as their primary vendor service)

Scale: 1 = Strongly Disagree; 7 = Strongly Agree

Topic	Avg.
We have difficulties with offshoring our software testing automation.	2.80
We have continual increases in software testing work and send it to our offshore vendor(s).	6.20

Direct Applications of STEP Research to Testing Practices

Quality Gates for Testing

Facilitates decoupled testing

Stop-Go decisions at 3 levels not at only the highest level

Business Applications

Limits impact of defects

Compartmentalizes/localizes negative consequence of IT failures

Facilitates multi-level quality analysis

Testability of EA Platforms

Common development and testing platform reduces cost, and

Facilitates shared development and testing work

Business Applications

Ensures modules developed are testable and saves resources

Forces modularity on systems design, thereby improving scalability and interconnectivity between systems

Facilitates implementation of complex enterprise systems

Direct Applications of STEP Research to Testing Practices

Strategic Alignment of Development/Testing

Highlight relational and structural components to align development and testing organizations

Business Applications

Provides concepts for strategic uses of testing at highest levels

Helps isolate misalignment leading to poor quality development

Provides basis for optimal allocation of resources between development and testing

Interface Testing

Structures connectivity between systems

Imposes traceability of between system defects

Business Applications

Forces IT design to be modular

Interfaces can be separately managed and modified

Complementarity of interfaces is managed helping interoperability

Direct Applications of STEP Research to Testing Practices

Mining Repositories for Defect Management

Supporting analysis of defect: types, data collection, categorization, communication, source, and fixing

Business Applications

Non-linear programming (NLP) used to isolate defects from reports for efficient defect management

Facilitate automation of defect detection

Optimizes management of test documentation

Testing for Virtualization

To support virtualized environments for development and testing

Business Applications

Ensures virtual containers are defect free

Separates defects from architecture and from application systems

Facilitates traceability of defects in distributed virtualized architectures

Direct Applications of STEP Research to Testing Practices

Business Rule Testing

Validation of business rules can be done at non-technical level
Stronger traceability to requirements

Business Applications

Forces business rules to be tested separately from code speeding development and testing
Changes to business rules can be easily tested and implemented
Ensures better validation to real-world considerations

Testing as a Service

Provides foundation for testing platforms comprising alternative tools and techniques

Business Applications

Helps ensure self-service can be applied to support unit testing
Helps track costs of testing in relation to development expenditures
Strengthens role of tester as supporters of efficient development

Advancing the Art & Science of Testing By Theory Building

step@memphis.edu
step.memphis.edu