# Evaluating the Impact of New Tools and Technologies Using Simulation

David M. Raffo, Ph.D., Portland State University
Tim Menzies, Ph.D., Portland State University

# Agenda

- Motivation
- Learned Defect Detectors Highlights
- Process Simulation Highlights
- Model Overview
- Three Scenarios and Results
- Conclusions

# Motivation

- Good new technologies are wasted
  - unless there is a compelling business case to use them
- Without such a case:
  - Managers not convinced
  - No reallocation of scarce resources
- Good technology: data mining defect detectors
  - increased PDs (probability of detection)
  - Lower PFs (probability of false alarm)
  - Lower inspection effort (more time for other, more specialized methods
- This talk:
  - The business case
  - Developed via process simulation

# Data mining defect detectors



- Data miners learn detect detectors from static code measures ( McCabe and Halstead)at the module level.
  - Not perfect: widely deprecated (Shepherd, Fenton, and others)
  - Adequate as partial indicators (but watch that false alarm rate)

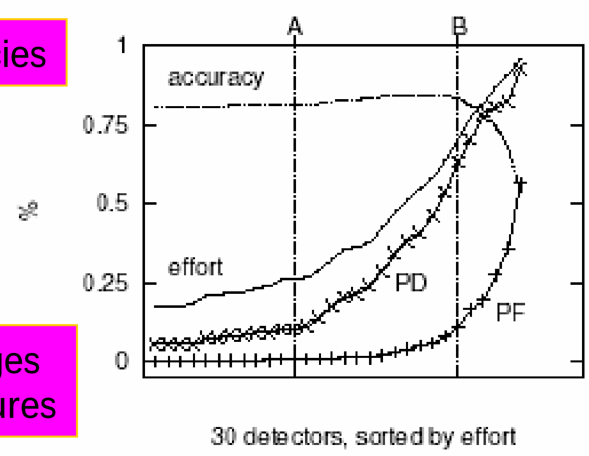
has defect

| No | Yes | |
|---|---|---|
| A | B | detector silent |
| C | D | detector triggered |

accuracy= (a+d)/(a+b+c+d)
pd = detection (or recall)
= d/(b+d)
pf = false alarms = c/(a+c)
prec = d/(c+d)
Effort = (C.loc + D.loc)/
(ABCD.loc)

Stable accuracies

Massive changes in other measures

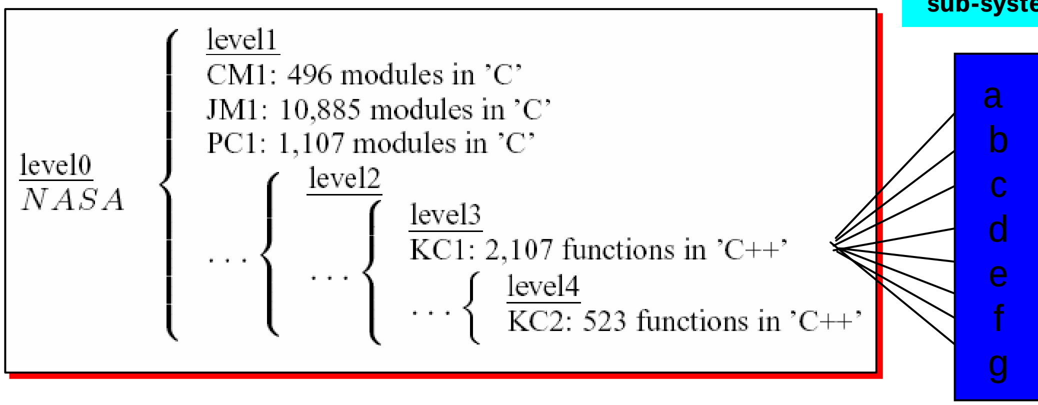


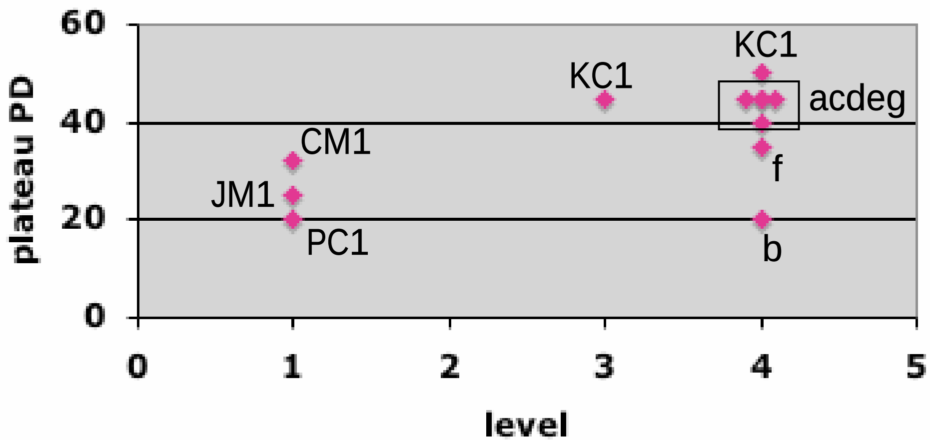
# Results

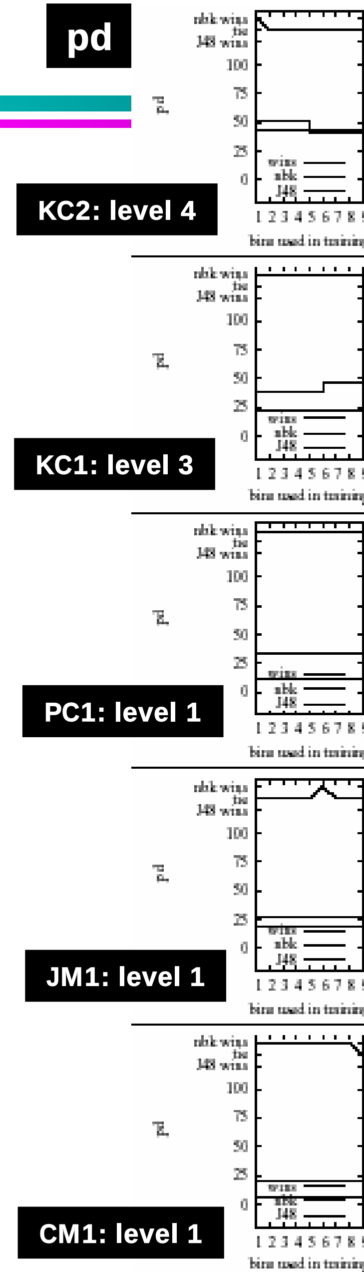


- NBK will suffice (in 85% cases NBK same or better than J48)
- Early plateaus (50-200 examples are enough)
- Not shown: low PFs
- Stratification improves PD?

**7 level 4 sub-systems**




- Suggestive, not conclusive evidence for "stratification improves PD"

# But, so what?

Is any of the above useful?

# Introducing - Process Simulation

- One area that can help companies improve their processes is ***Process Simulation***.
- Process Simulation supports organizations to address
  - Strategic management
  - Process Planning
  - Control and operational management
  - Technology adoption
  - Understanding
  - Training and learning
  - Quantitative process management and other CMMI-Based Process Improvement

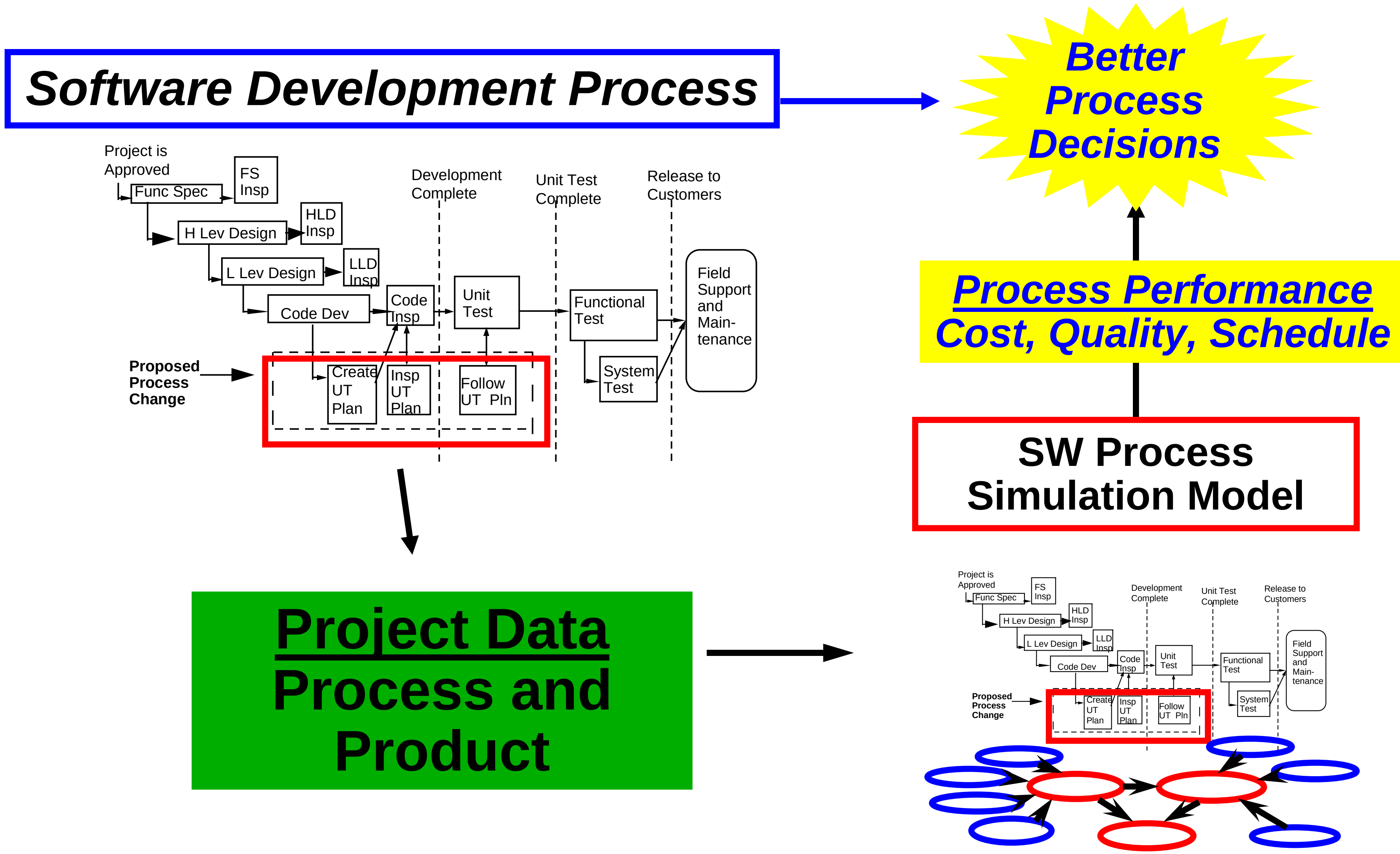
# Features of Process Simulation and PTAM

- **Based on extensive research**.
- **Graphical user interface** and models software processes
- **Utilizes SEI methods** to define SW Processes
- **Integrates metrics** related to cost, quality, and schedule into understandable performance picture.
- **Predicts project-level impacts** of process improvements in terms of cost, quality and cycle time
- **Support business case analysis** of process decisions - ROI, NPV and quantitatively assessing risk.
- **Designed for Rapid Deployment**

# Importance/Benefits – Enduring Needs

- NASA Project Level
  - Software Quality Assurance Strategy Evaluation for NASA Projects
  - Independent Bottoms-Up NASA Project Cost Estimation (Going where COCOMO cannot – KSC project)
  - NASA Contractor Bid Evaluation (NASA IV&V integrated part of Planning and Scoping/Cost Estimation strategy)
  - Software Assurance Replanning
  - Cost/Benefit Evaluation of new technologies and tools

# How it works


Software Development Process
Better Process Decisions
Project is Approved
Func Spec
FS Insp
H Lev Design
HLD Insp
L Lev Design
LLD Insp
Code Dev
Code Insp
Development Complete
Unit Test
Unit Test Complete
Functional Test
System Test
Release to Customers
Field Support and Main-tenance
Proposed Process Change
Create UT Plan
Insp UT Plan
Follow UT Pln
Process Performance
Cost, Quality, Schedule
SW Process Simulation Model
Project Data
Process and Product

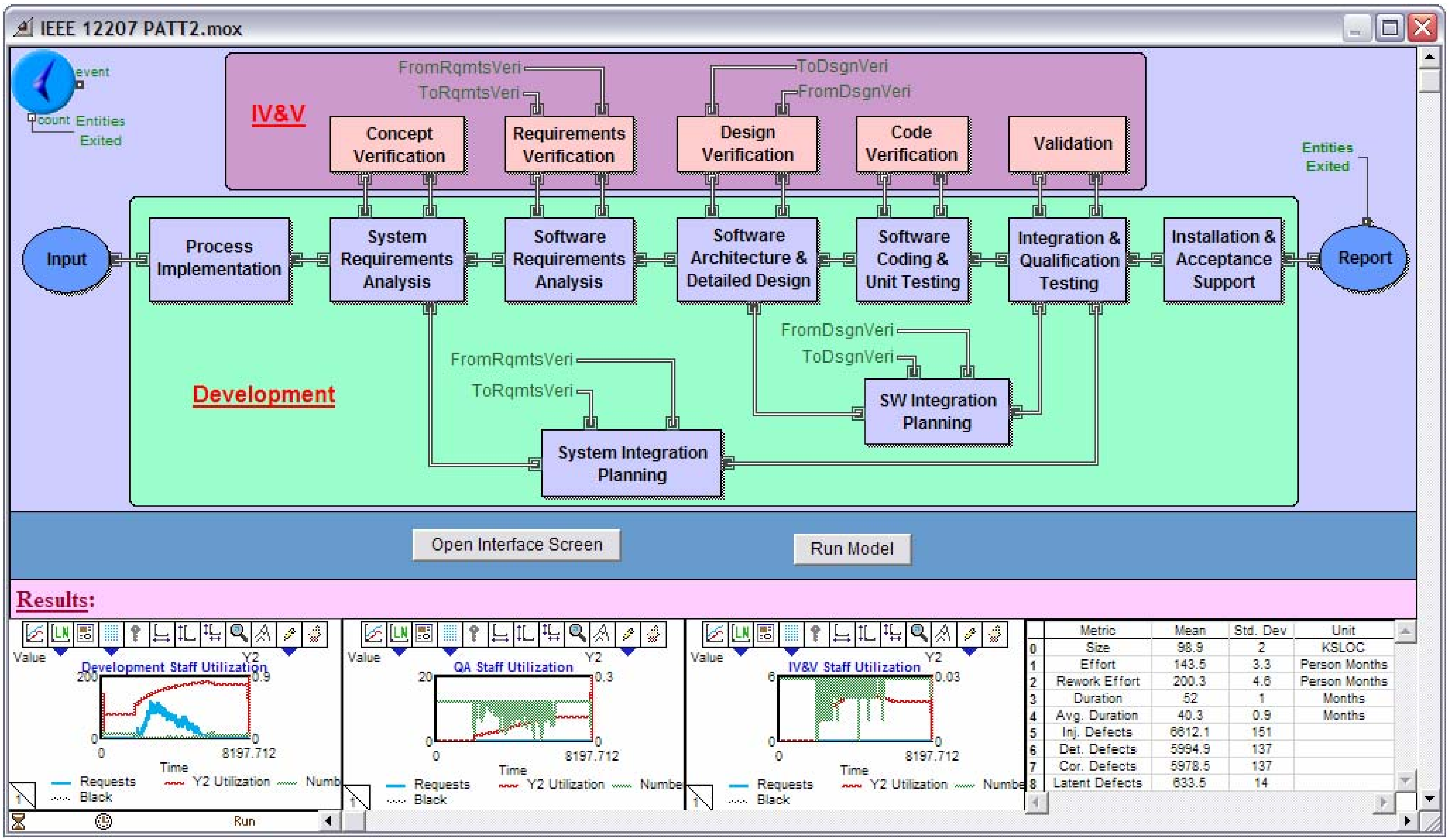
# Goal

- In this presentation, we assess the impact of a new technology (i.e. Learned Defect Detectors) on a “typical” large-scale NASA project in terms of overall cost, quality and schedule performance
- Goal: To determine when the new technology might be ***useful*** and when they might be ***useless*** by providing a business case to support the adoption of these tools.

# Business Case Questions

- What is the impact of applying new tools and technologies?
- What is the economic benefit or value of the tool or technology? What is the ***Return on Investment***?
- Under what conditions does the tool or technology perform best? Under what conditions does it perform poorly?
- What performance standards does the tool need to achieve in order to have a positive performance impact on the project/organization?
- Are there alternative ways to apply the tool or technology that enable it to provide a more positive impact?

# NASA Model – Includes IV&V Layer with IEEE 12207 Software Development Lifecycle

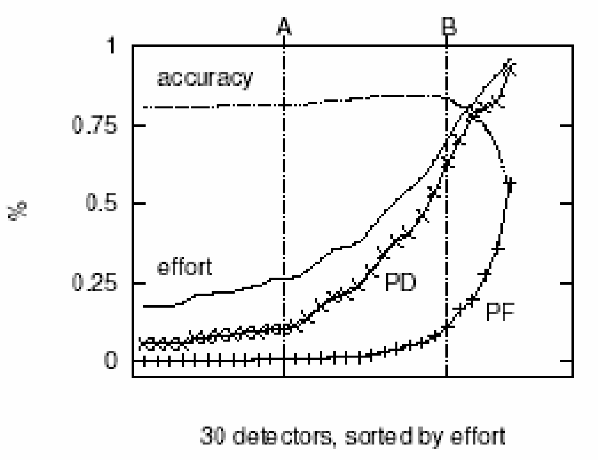


# IV&V Layer – Select Criticality Levels for IV&V Techniques using pull-down menus

Notebook - IEEE 12207 PATT2.mox

| | | Concept Verification | | Requirements Verification | | Design Verification | | Code Verification | | Validation | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ID | IV&V Technique | Consequence | Error Potential | Consequence | Error Potential | Consequence | Error Potential | Consequence | Error Potential | Consequence | Error Potential |
| 1.1 | Management and Planning of Independent Verification and Validation | None | None | None | None | None | None | None | None | None | None |
| 1.2 | Issue and Risk Tracking | None | None | None | None | None | None | None | None | None | None |
| 1.3 | Final Report Generation | None | None | None | None | None | None | None | None | None | None |
| 1.4 | IV&V Tool Support | None | None | None | None | None | None | None | None | None | None |
| 1.5 | Management and Technical Review Support | None | None | None | None | None | None | None | None | None | None |
| 1.6 | Criticality Analysis | None | None | None | None | None | None | None | None | None | None |
| 1.7 | Identify Process Improvement Opportunities in the Conduct of IV&V | None | None | None | None | None | None | None | None | None | None |
| 2.1 | Reuse Analysis | None | None | | | | | | | | |
| 2.2 | Software Architecture Assessment | None | None | | | | | | | | |
| 2.3 | System Requirements Review | None | None | | | | | | | | |
| 3.1 | Traceability Analysis – Requirements | | | None | None | | | | | | |
| 3.2 | Software Requirements Evaluation | | | None | None | | | | | | |
| 3.3 | Interface Analysis – Requirements | | | None | None | | | | | | |
| 3.4 | System Test Plan Analysis | | | None | None | | | | | | |
| 4.1 | Traceability Analysis – Design | | | | | None | None | | | | |

# Assumptions

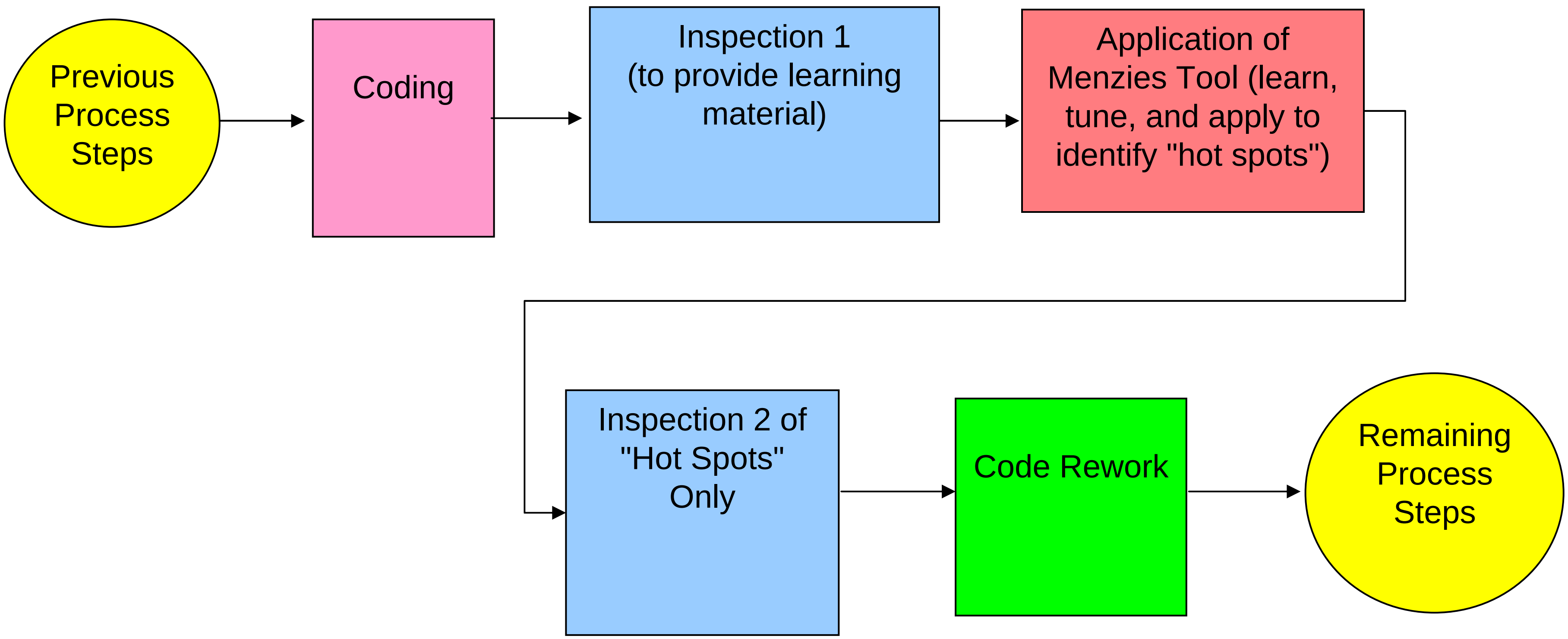



- Project Size is 100 KSLOC.
- Software process follows the IEEE 12207+IV&V model. True for many DoD and NASA projects.
- %LOC Inspected=PD+5% to 10%; and %LOC is proportional to Effort
- PF = 10%-30%.
- PD=40 to 70%.
- The PD rate assumes, in turn, that defect detectors are learned from data divided below the sub-system level.
- Standard manual inspections find 40% to 60% of the total defects.
- Perspective Based inspections find 80% to 90% of latent defects
- Defects uniformly distributed throughout code

# Scenario I - Applying LDD to V&V

- Learned defect detectors are applied during project V&V.
  - Inspections are conducted on 11.5% of code to learn defect detectors
  - LDDs then applied to remaining code to identify high-risk portions of the system
  - Explored the impact of using higher PD combined with higher PF
  - Explored the impact of using regular inspections(weak training set) vs Perspective Based inspections (strong training set) for LDDs.

# Changes to the Process



# Scenario I - Results Summary

- Model recommendations for specific scenarios
- General Rule:

  Insp Effect* %Code_Inspected*95%<= E_LDD* TS_IE

Where:

Insp Effect – Probability of detection of V&V inspections

%Code_Inspected - % of code inspected during V&V

E_LDD – Probability of Detection for LDDs

TS_IE – Probability detection of Training Set inspections

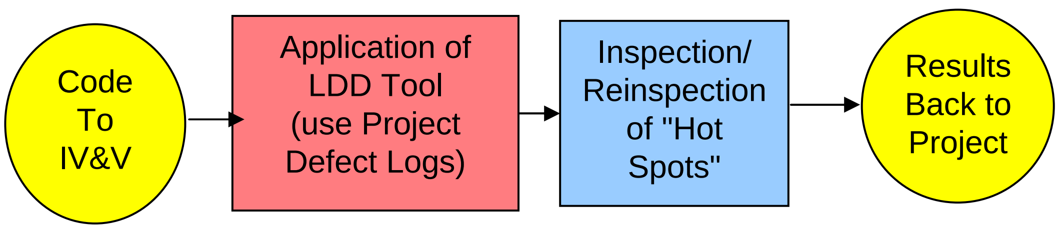
# Scenario I - Results Summary

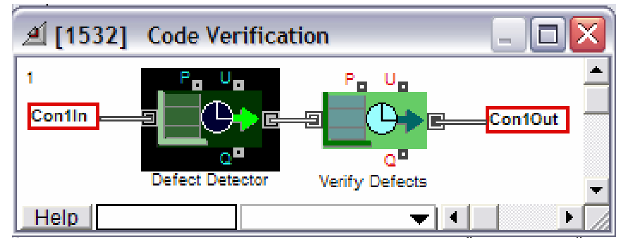
- LDDs are ***Useful*** (Significant benefits) in a V&V setting when:
  - 53% or less of the code is inspected during V&V (manned vs unmanned missions) using regular inspections and LDD PD =50%
  - Using high PD mode and Perspective based inspections
  - Project inspections are poor
- Applying LDDs to V&V are ***Useless*** when:
  - Project inspections are good or high quality
  - More than 53% of the code is inspected by V&V (typical for manned missions)

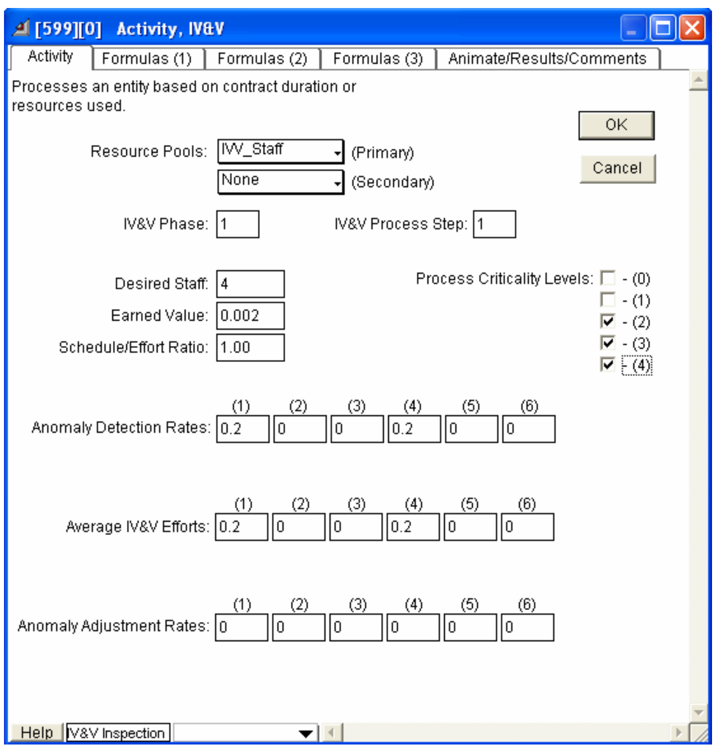
# Scenario II - Applying LDD to IV&V

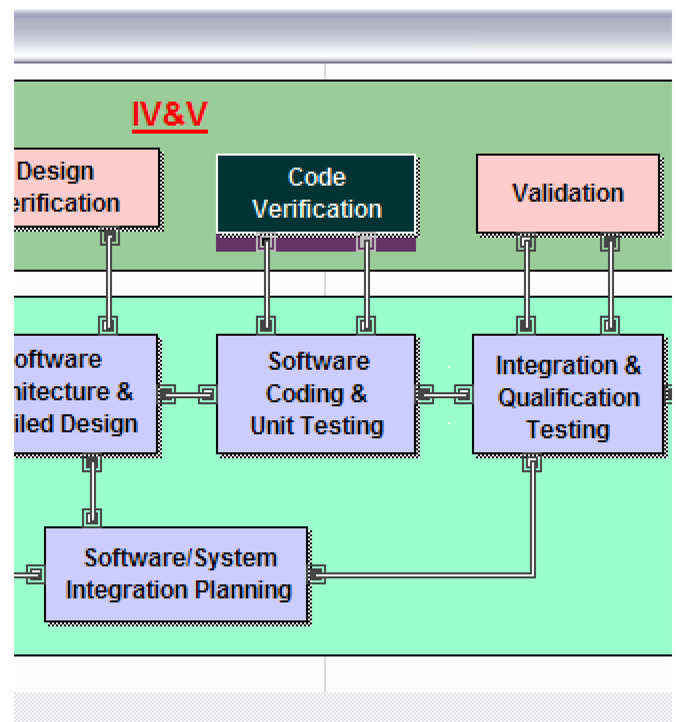
- Learned Defect Detectors (LDD) applied to IV&V (Shedding light on blind spots)
  - Project generated training sets (regular inspections)
  - Investigated the Impact of applying LDD to different project types (varied amount of code that is reinspected (100%-25%))
  - Varied the effectiveness of reinspection (2%-10%)

# Changes to the Process – IV&V



# Scenario II - Results

- Clear recommendations for specific scenarios
- Results (Excellent Application):
  - Low Risk = 1.2 PM with no defects detected
  - Improves quality if any defects are found (detection capability > 0)
  - Receive added assurance even if detection capability is 0
  - For Manned Missions, (100% reinspection), break-even on total project effort if IV&V reinspection effectiveness = 2%
  - Significantly improves cost, quality and schedule if reinspection effectiveness is >= 5%

# Scenario II - Results

- Significant up side potential when LDDs are used to identify high risk portions of the code that were not previously inspected during project level V&V (unmanned missions).
- At 50% code inspected by V&V, 4%-7.5% reduction in delivered defects
- At 25% code inspected during V&V, reductions in delivered defects range from 15%-24%. Effort savings range from 18 PMs to 29 PMs.

# Conclusions

- Learned Defect Detectors ***are useful*** when they ***increase*** the overall detection capability of the Coding phase.
- General Rule:
- Insp Effect* %Code_Inspected*95%<= E_LDD* TS_IE
- This occurs when:
  - Less than 53% of code is inspected during V&V or V&V has week inspections
  - Used as IV&V technique identifying blind spots and augmenting regular high-quality V&V
  - V&V has weak inspections

# Conclusions

- Learned Defect Detectors ***are useless*** when they ***decrease*** the overall detection capability of the Coding phase.
- This occurs when:
  - Used to frivolously cut costs by replacing high quality code inspections.

# Conclusions – Broader Impacts

- Identify the conditions under which application of a new technology **would be** beneficial and when applying this technology **would not be** beneficial.
- We can define ***performance benchmarks*** that a new tool or technology needs to achieve.

# Conclusions – Broader Impacts

- We can ***diagnose problems*** associated with implementing a new tool or technology and ***identify new ways*** to apply the technology to the benefit of the organization (and the vendors)
- Finally, we can do all this ***before*** the technology is purchased or applied and therefore can save scarce resources available for process improvement.

# The End

# Contact Information

David M. Raffo, Ph.D.

Associate Professor, Portland State University
Visiting Scientist, Software Engineering Institute

President, Vaiyu, Inc.

raffod@pdx.edu

503-725-8508

503-939-1720