COCOMO® III Workshop

Brad Clark, PhD

32nd International Forum on COCOMO® and System/Software Cost Modeling

October 17, 2017

Topics

- Required Security Implementation results
- New Nominal
- Model Definition Manual - Beta
- Next steps
  - Working group review of the model
  - Rosetta stone COCOMO II ➔ COCOMO III
  - Model performance analysis
  - Data collection
    - Form
    - Plan

Required SW Security (SECU)

- New security cost driver added to COCOMO III
- Two draft standards were used to provide security information used in the assessment
  - Lifecycle Requirements: ISA-62443-4-1 Secure Product Development Lifecycle Requirements
  - Component Requirements: ISA-62443-4-2 Technical Security Requirements for Industrial Automation And Control Systems (IACS) Components
- Two workshops were held to assess how development lifecycle and software component requirements would be handled by COCOMO III
  - Participants (12) were asked to identify where the model addressed security lifecycle and component requirements.
  - 3 Votes for a model parameter required to be considered for model modification

ISA: International Society of Automation

SW Components Security Req’ts -1

- FR-1 Identification and authentication control
  - Human user identification and authentication
  - Software process and device identification and authentication
  - Account management
  - Identifier management
  - Authentication management
- Addressed by:
  - Product Size
  - Complexity (enhanced definition)

SW Components Security Req’ts -2

- FR-2 Use control
  - Authorization enforcement
  - Wireless control
  - Use control for portable and mobile devices
  - Session lock
  - Remote session termination
- Addressed by:
  - Product Size
  - Complexity (enhanced definition)

SW Components Security Req’ts -3

- FR-3 System integrity
  - Communication integrity
  - Malicious code protection
  - Software and information integrity
  - Input validation
  - Error handling
- Addressed by:
  - Inconclusive, no strong results
  - Mentioned:
    - Size
    - Use of Software Tools
    - Automated Analysis
    - Execution Testing and Tools
    - Software Architecture Understanding

SW Components Security Req’ts -4

- FR-4 Data confidentially
  - Information confidentiality
  - Information persistence
  - Use of cryptography
- Addressed by:
  - Product Size
  - Complexity (enhanced definition)

SW Components Security Req’ts -5

- FR 5: Restricted data flow
  - Network segmentation
  - Zone boundary protection
  - Person-to-Person communication restrictions
- Addressed by:
  - Product Size

- FR 6: Timely response to events
  - Audit log accessibility
  - Continuous monitoring
- Addressed by:
  - Product Size
  - Platform

SW Components Security Req’ts -6

- FR-7 Resource availability
  - Denial of service protection
  - Resource management
  - Control system backup, recovery and reconstitution
  - Network and security configuration settings
  - Least functionality
- Addressed by:
  - Software Architecture
  - Complexity (enhanced definition)

Development LC Security Req'ts -1

- Practice 1: Security management
  - Identification of responsibilities
  - Security expertise
  - Code signing
  - Development environment security
  - 3rd party embedded component security
  - Process verification
- Addressed by:
  - Process Capability
  - Applications Experience
  - Use of Software Tools
  - Development Flexibility

Development LC Security Req'ts -2

- **Practice 2: Specification of security requirements**
  - **Product security requirements (authentication, authorization, encryption, auditing and other security capabilities)**
  - **Product security context (product's intended operating environment including physical environment)**
  - **Threat model (analysis that identifies potential security issues and how they will be addressed)**
  - **Security requirements review**
- **Addressed by:**
  - **Development Flexibility**
  - **Process Capability**
  - **Peer Reviews**

Development LC Security Req'ts -3

- **Practice 3: Secure by design**
  - **Secure design principles**
  - **Defense in depth design (layers of security)**
  - **Security design review**
  - **Assessing & addressing security-related issues**
- **Addressed by:**
  - **Software Architecture Understanding**
  - **Applications Experience**
  - **Process Capability**
  - **Peer Reviews**

Development LC Security Req’ts -4

- Practice 4: Secure implementation
  - Security implementation review
  - Assessing & addressing security-related issues
- Addressed by:
  - Process Capability
  - Peer Review
  - Applications Experience

Development LC Security Req’ts -5

- **Practice 5: Security verification and validation testing**
  - **Security requirements testing**
  - **Threat mitigation testing**
  - **General vulnerability testing**
  - **Penetration testing**
- **Addressed by:**
  - **Execution Testing and Tools**

Development LC Security Req’ts -6

- **Practice 6: Security defect management**
  - **Receiving notifications of security-related issues**
  - **Reviewing security-related issues**
  - **Assessing & addressing security-related issues**
  - **Disclosing security-related issues**
- **Addressed by:**
  - **Process Capability**

Development LC Security Req’ts -7

- **Practice 7: Security update management**
  - **Dependent component or operating system security update documentation**
  - **Security update delivery**
  - **Timely delivery of security patches**
- **Addressed by:**
  - **Process Capability**

SW Component Requirements

- Identification and authentication control: identify and authenticate all users and other software applications
- Use control: enforce assigned privileges of user or another software application
- System integrity: ensure integrity of the application to prevent unauthorized manipulation
- Data confidentially: ensure confidentiality of information and communication channels and prevent unauthorized disclosure
- Restricted data flow: segment the control system via zones and conduits to limit the unnecessary flow of data
- Timely response to events: respond to security violations by notification and reporting in a timely manner
- Resource availability: ensure availability of the application against degradation or denial of essential services

SECU Rating

- **These rating scales describe the level of security based on unauthorized disclosure or attack. The higher the rating, the more complex the implementation for the security requirements (previous slide).**

Rating	Very Low	Low	Nominal	High	Very High	Extra High
SECU			Prevent unauthorized disclosure using casual eavesdropping or casual exposure	Prevent unauthorized disclosure using simple means with generic skills	Prevent unauthorized disclosure using sophisticated means but moderated resources	Prevent unauthorized disclosure using sophisticated means and extended resources

- **Other cost drivers in this model account for additional effort required for security practices.**

Secure LC Impact on Drivers

- Process Capability
  - Security management
  - Specification of security requirements
  - Secure by design
  - Secure implementation
  - Security defect management
  - Security update management
- Applications Experience
  - Security management
- Use of Software Tools
  - Security management
- Development Flexibility
  - Security management
- Development Flexibility
  - Specification of security requirement
- Peer Reviews
  - Specification of security requirements
  - Secure by design
  - Secure implementation
- Software Architecture Understanding
  - Secure by design
- Applications Experience
  - Secure by design
  - Secure implementation
- Execution Testing and Tools
  - Security verification and validation testing

Topics

- Required Security Implementation results
- New Nominal
- Model Definition Manual - Beta
- Next steps
  - Working group review of the model
  - Rosetta stone COCOMO II ➔ COCOMO III
  - Model performance analysis
  - Data collection
    - Form
    - Plan

The New "Nominal"

- Cost Driver definition refinement

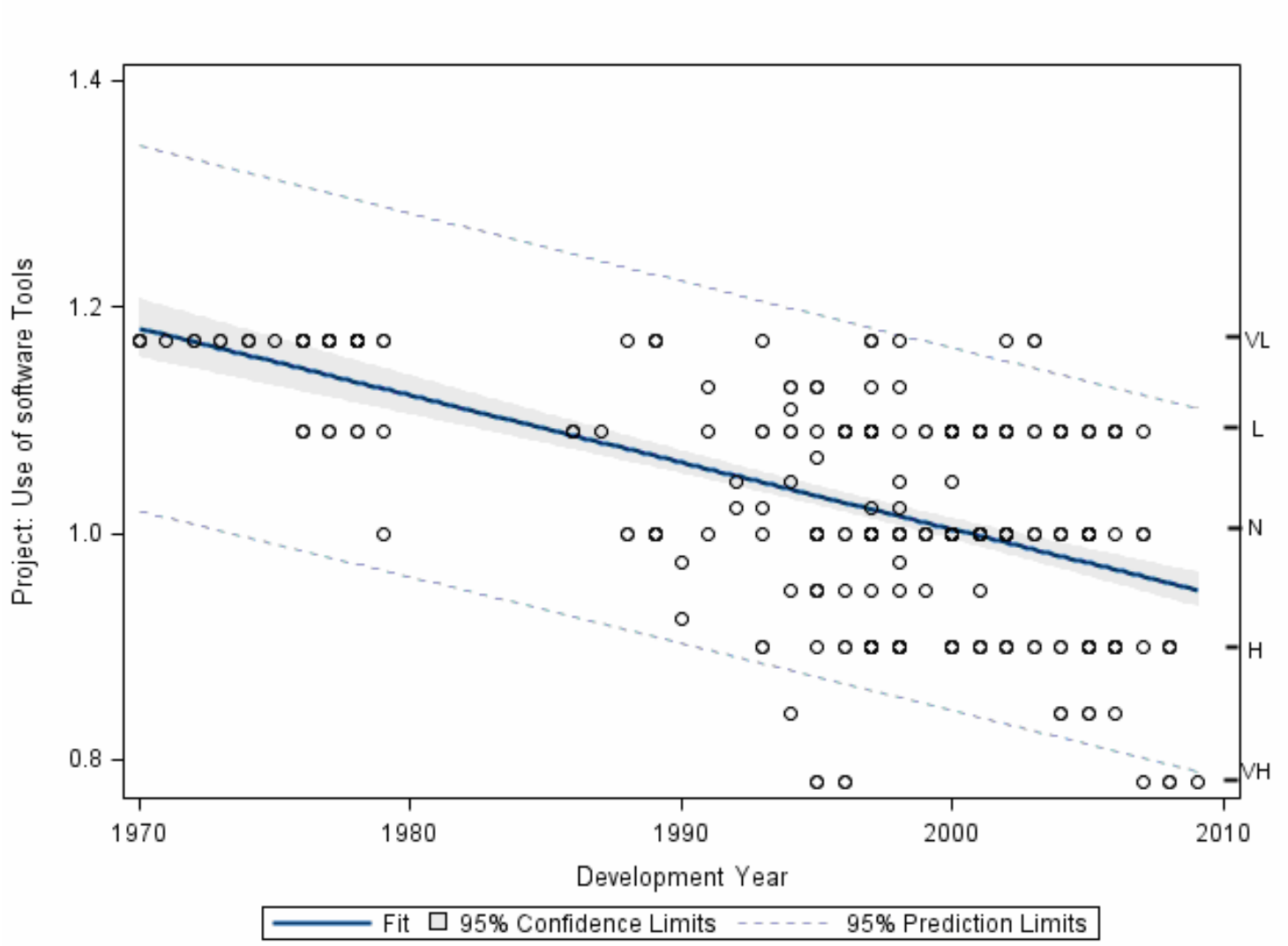
- The New "Nominal
  - Study of productivity trends over the past 40 years reveals a shift in "nominal" ratings
  - Following slides show the shift in ratings for selected cost drivers, i.e., the new "nominal"

Impact of Productivity Trends

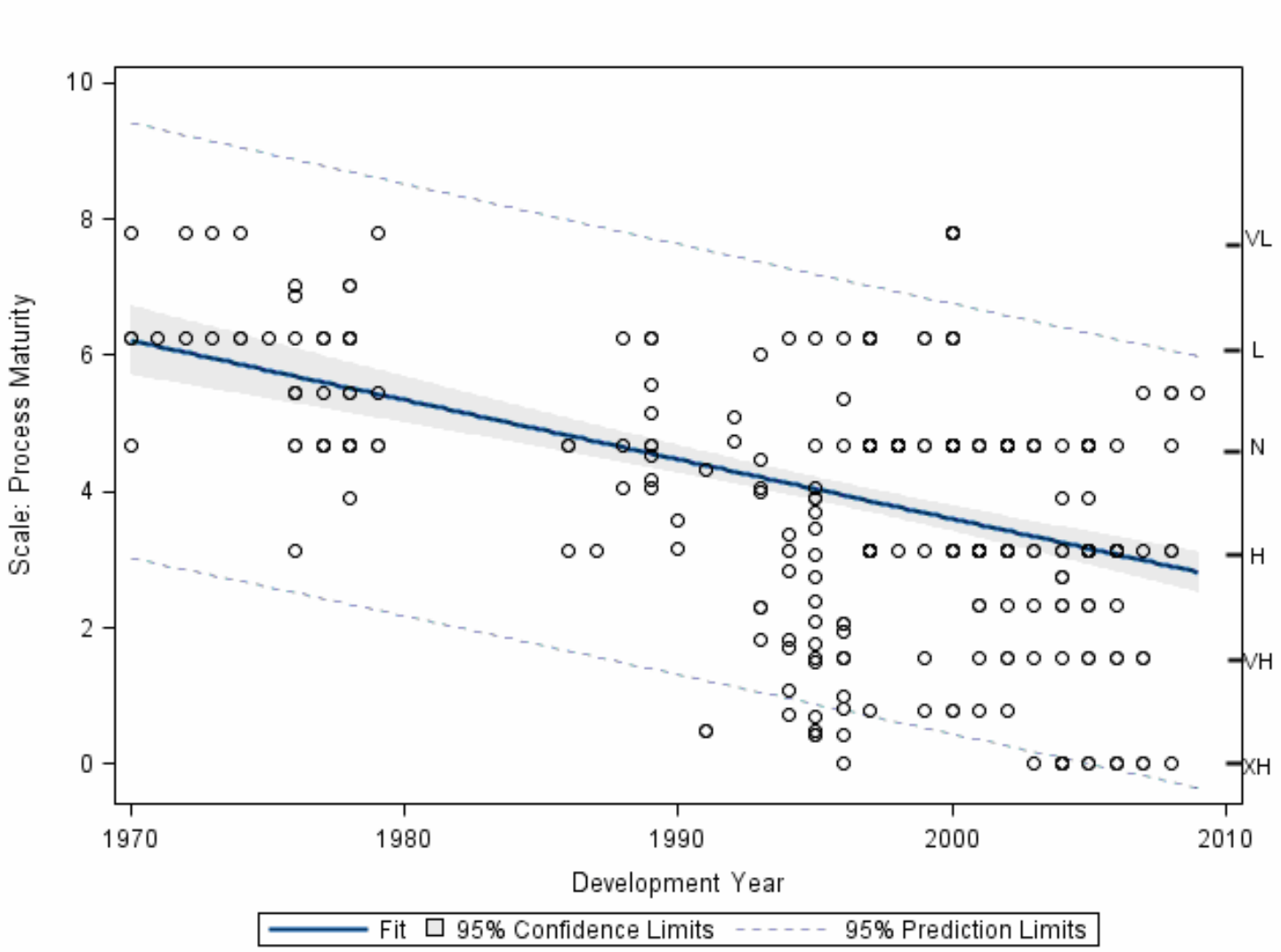
Kendall's Rank Correlation Coefficients between the Completion Year and COCOMO II Cost Drivers (sorted by degrees of correlation)

Cost driver Kendall's	τ	p-value
TOOL Use of Software Tools	-0.37	2.20E-16
PMAT Process Maturity (PCUS)	-0.30	1.22E-13
STOR Main Storage Constraint	-0.29	1.31E-11
TIME Execution Time Constraint	-0.26	6.62E-10
PLEX Platform Experience	-0.17	1.98E-05
PVOL Platform Volatility	-0.18	2.04E-05
APEX Applications Experience	0.17	4.88E-05
LTEX Language and Tool Experience	-0.15	2.84E-04
~~DATA Database Size~~	0.13	1.81E-03
RELY Required Software Reliability	-0.10	1.42E-02
CPLX Product Complexity	-0.10	1.58E-02
PREC Precedentedness of Application	-0.09	2.13E-02
ACAP Analyst Capability	0.08	4.87E-02

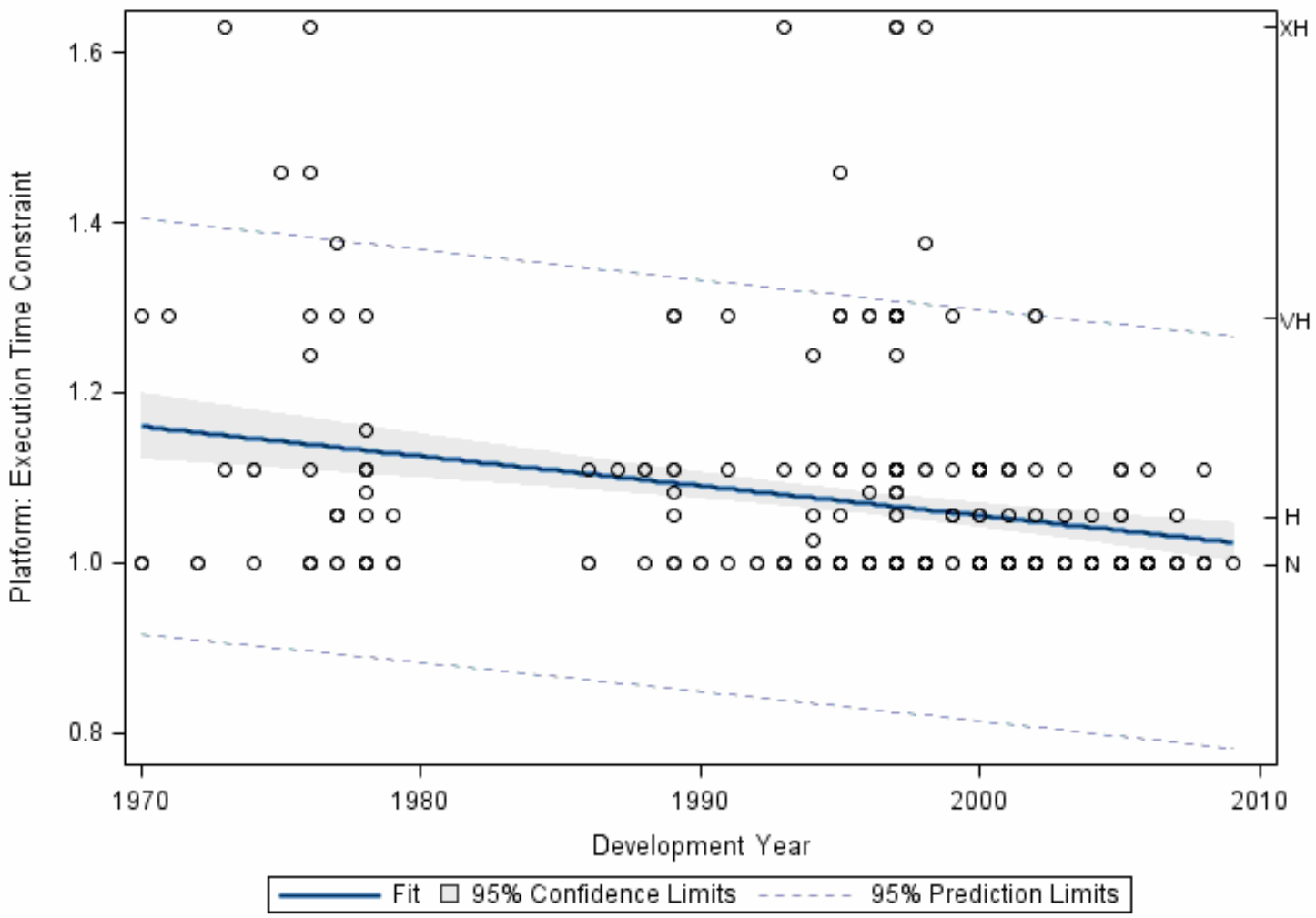
Use of Software Tools



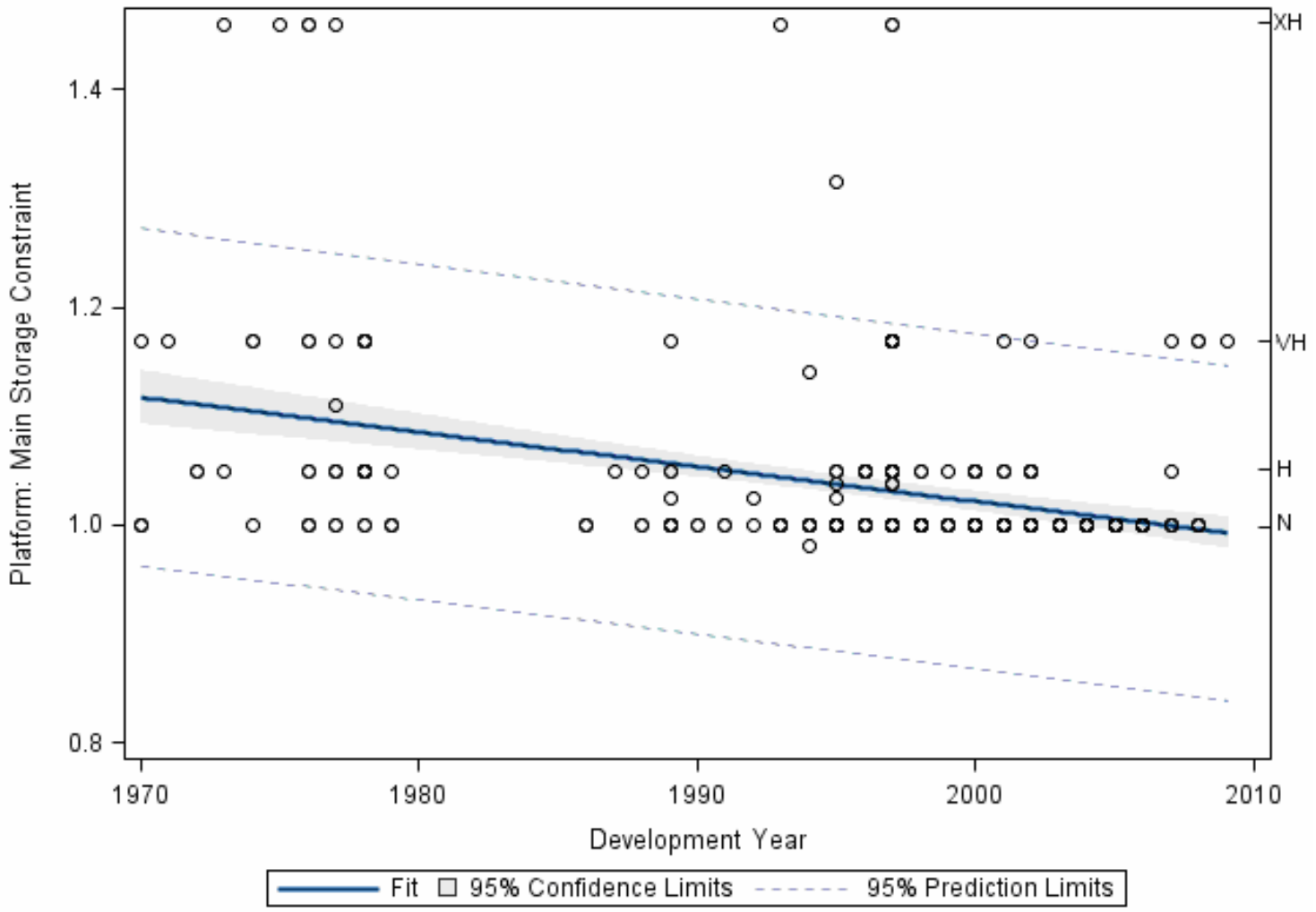
Process Maturity (Now PCUS)



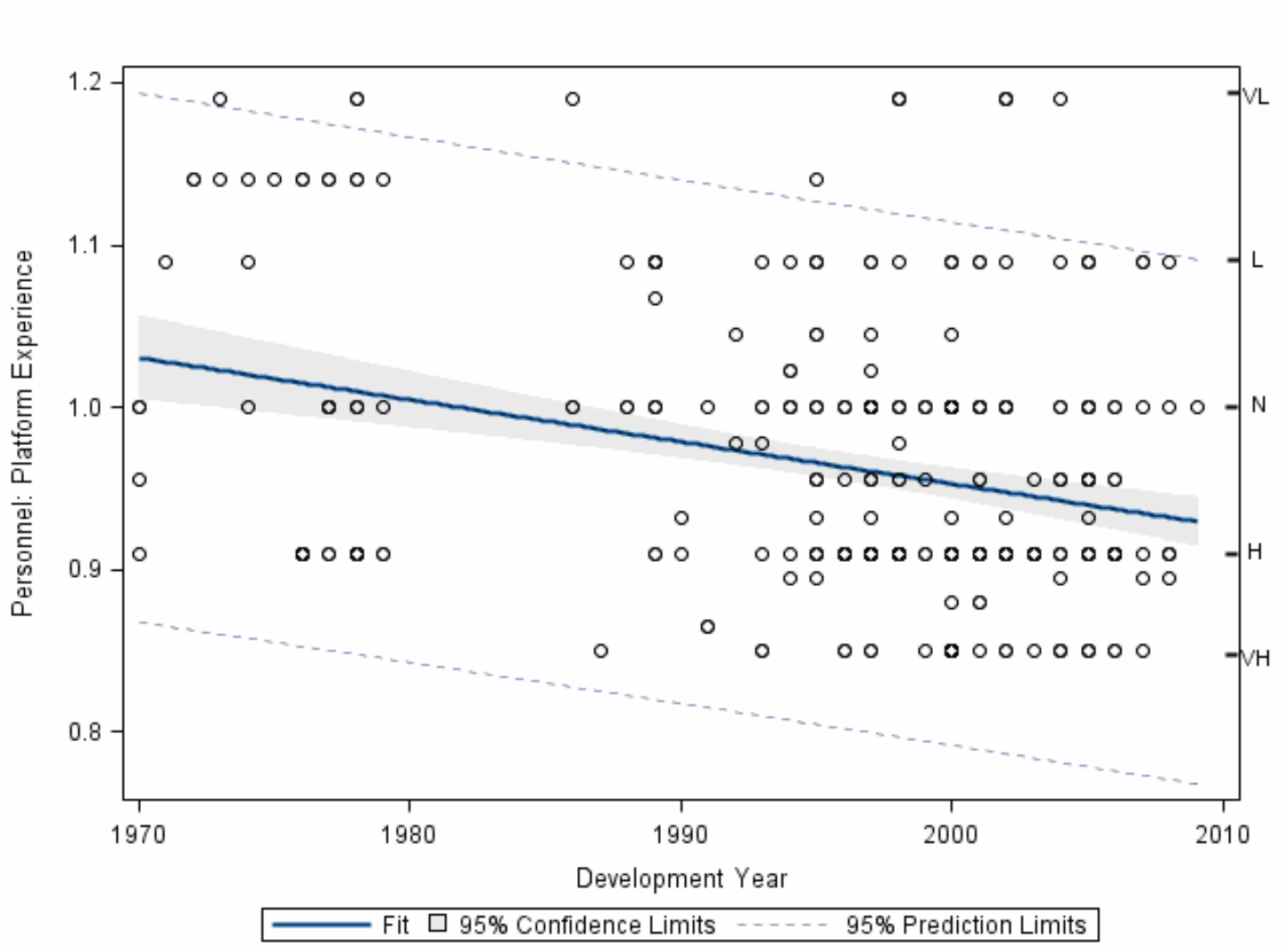
Execution Time Constraint-TIME



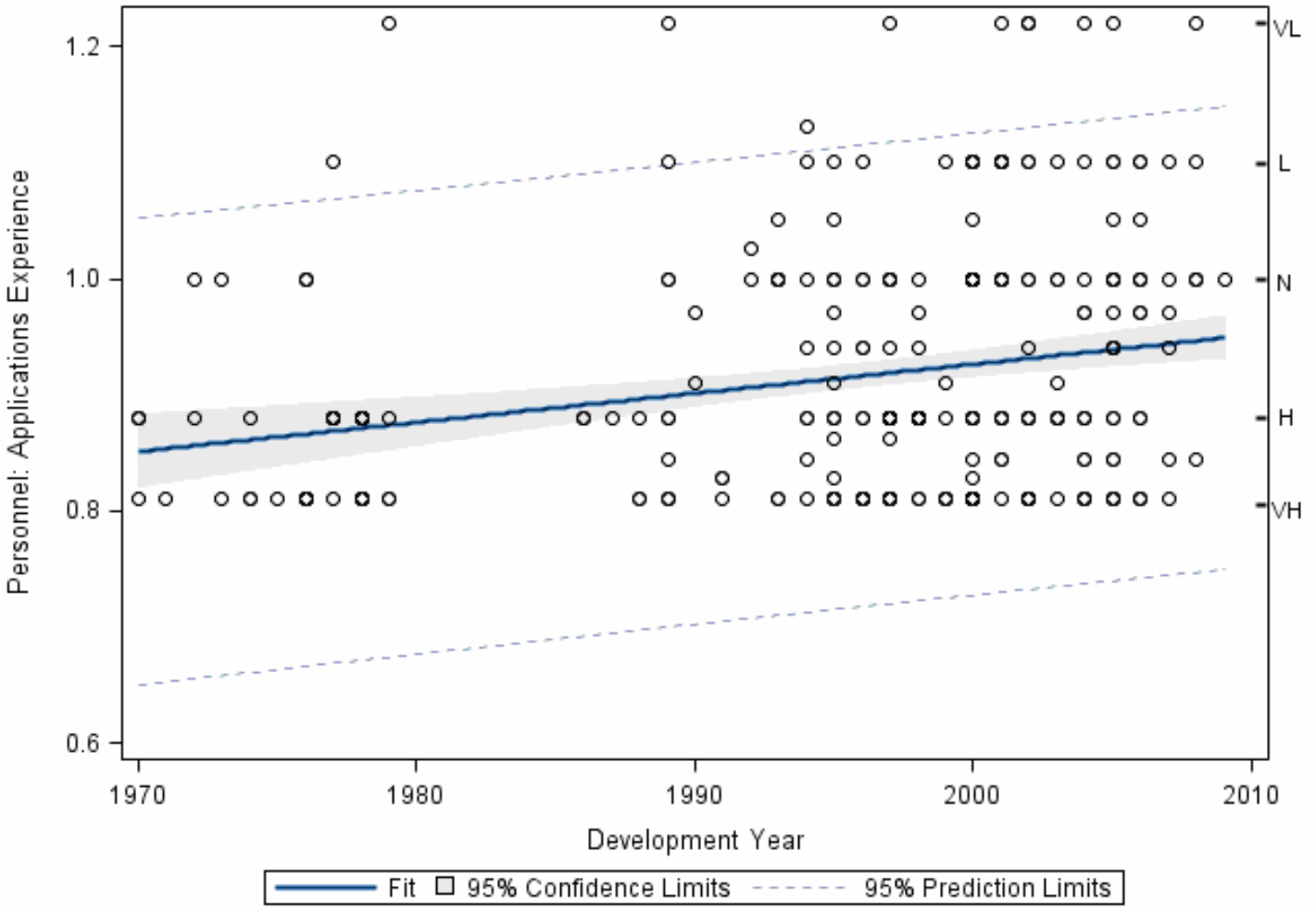
Main Storage Constraint-STOR



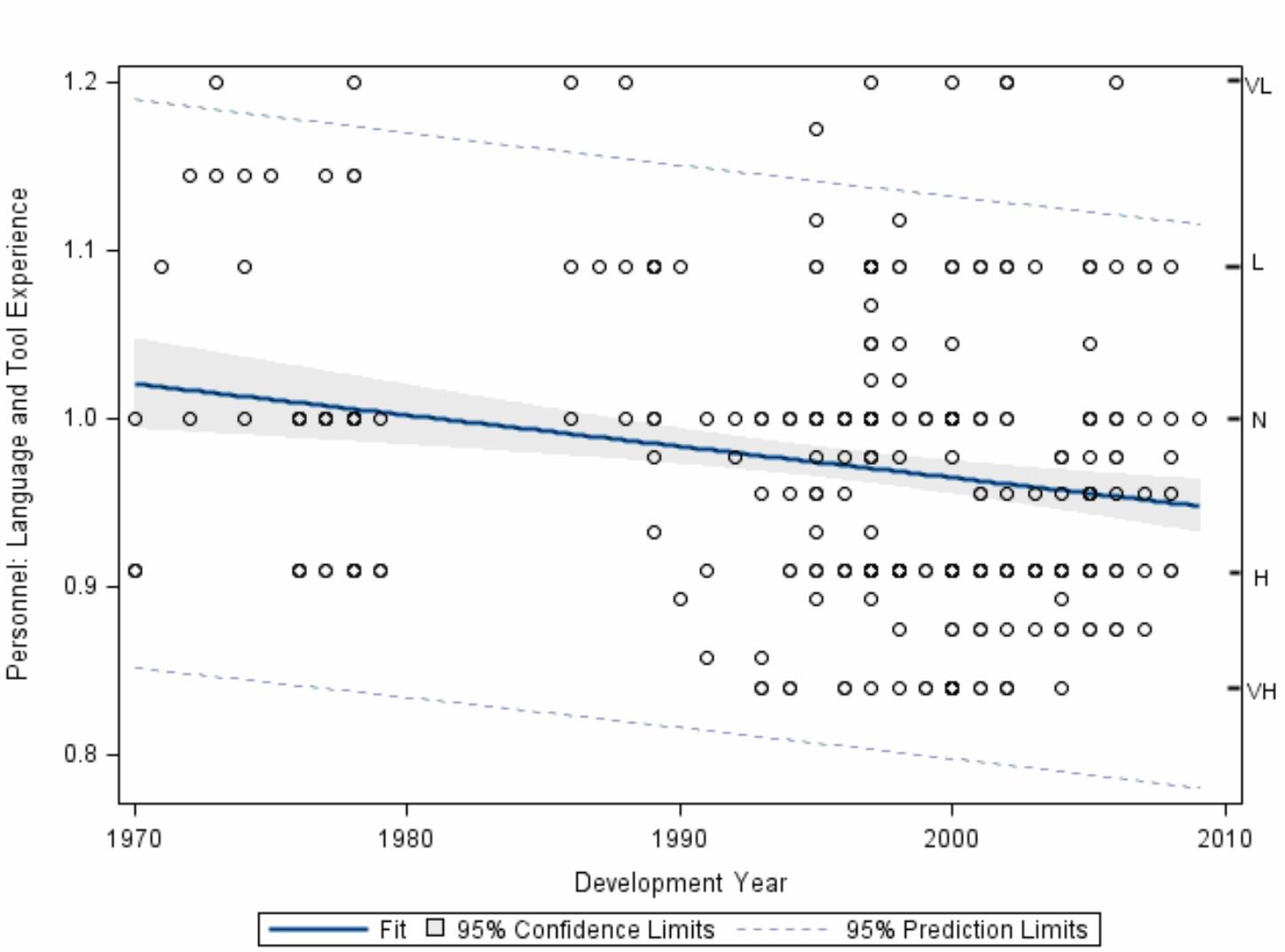
Platform Experience-PLEX



Applications Experience-APEX



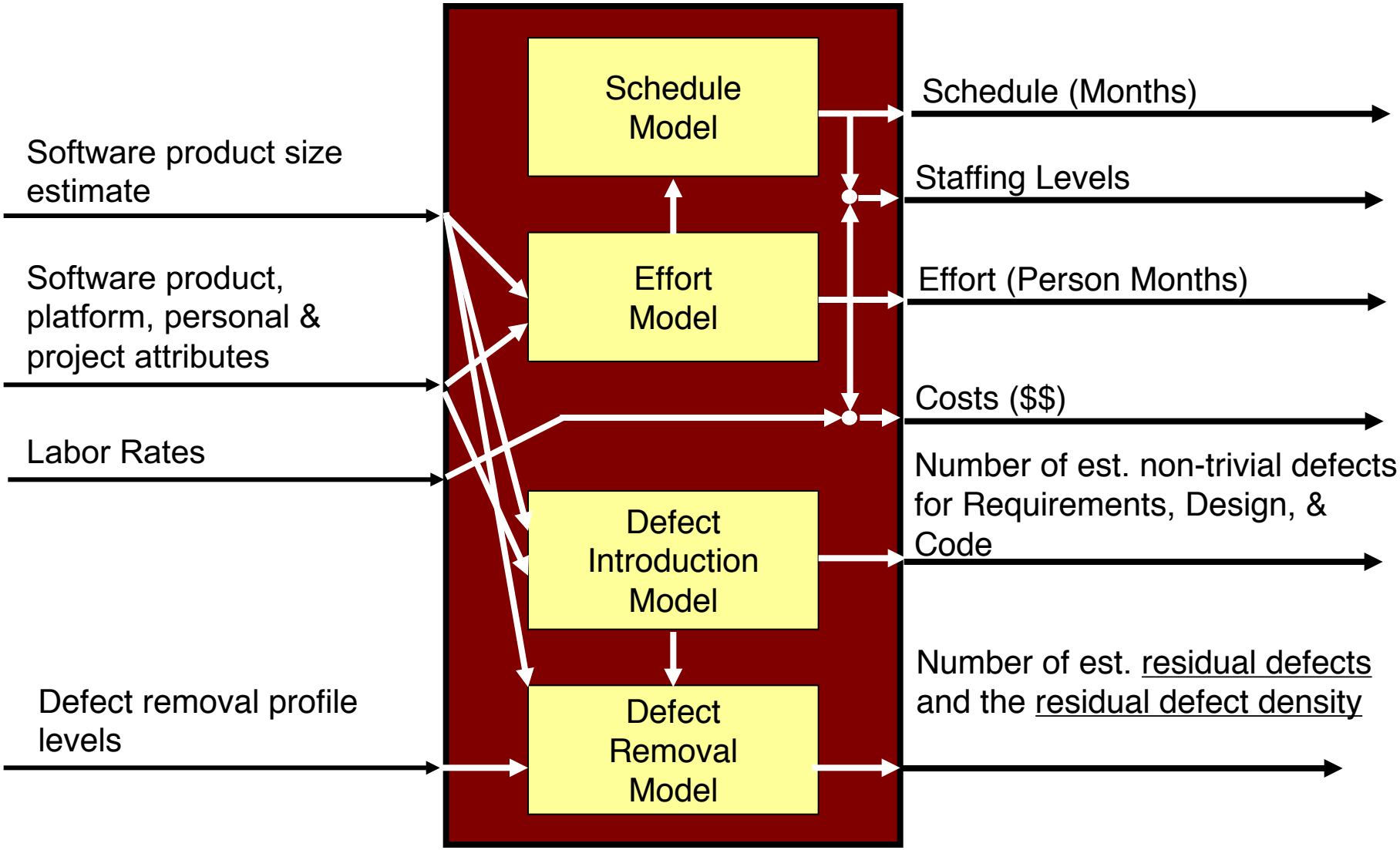
Language and Tool Experience-LTEX



Topics

- Required Security Implementation results
- New Nominal
- **Model Definition Manual - Beta**
- **Next steps**
  - **Working group review of the model**
  - **Rosetta stone COCOMO II ➔ COCOMO III**
  - **Model performance analysis**
  - **Data collection**
    - **Form**
    - **Plan**

COCOMO III Model Concept



Model Match to ICSM Common Cases -3

Case 5: Hardware with Embedded Software Component*

- Example: Multi-sensor control device
- Size, Complexity: Low
- Typical Change Rate/Month: 0.3 - 1 %
- Criticality: Medium to very high
- NDI Support: Good, in place
- Organizational Personnel Capability: Experienced, medium-high
- Activities: Concurrent hardware/software engineering. CDR-level review. IOC development, LRIP, FRP. Concurrent version N+1 engineering
- Time/Build: Software 1-5 days
- Time/Increment: Market-driven

** Means this process is suitable for COCOMO III*

Case 6: Indivisible IOC*

- Example: Complete vehicle platform
- Size, Complexity: Medium to high
- Typical Change Rate/Month: 0.3 – 1%
- Criticality: High to very high
- NDI Support: Some in place
- Organizational Personnel Capability: Experienced, medium to high
- Activities: Determine minimum-IOC likely, conservative cost. Add deferrable software features as risk reserve. Drop deferrable features to meet conservative cost. Strong award fee for features not dropped.
- Time/Build: Software: 2-6 weeks
- Time/Increment: Platform: 6-18 months

Model Match to ICSM Common Cases -5

Case 8: Hybrid Agile/Plan-Driven System*

- Example: C4ISR system
- Size, Complexity: Medium to very high
- Typical Change Rate/Month: Mixed parts; 1-10%
- Criticality: Mixed parts; Medium to very high
- NDI Support: Mixed parts
- Organizational Personnel Capability: Mixed parts
- Activities: Full ICSM, encapsulated agile in high change, low-medium criticality parts (Often HMI, external interfaces). Full ICSM, three-team incremental development, concurrent V&V, next-increment re-baselining
- Time/Build: 1-2 months
- Time/Increment: 9-18 month

** Means this process is suitable for COCOMO III*

Case 9: Multi-Owner System of Systems*

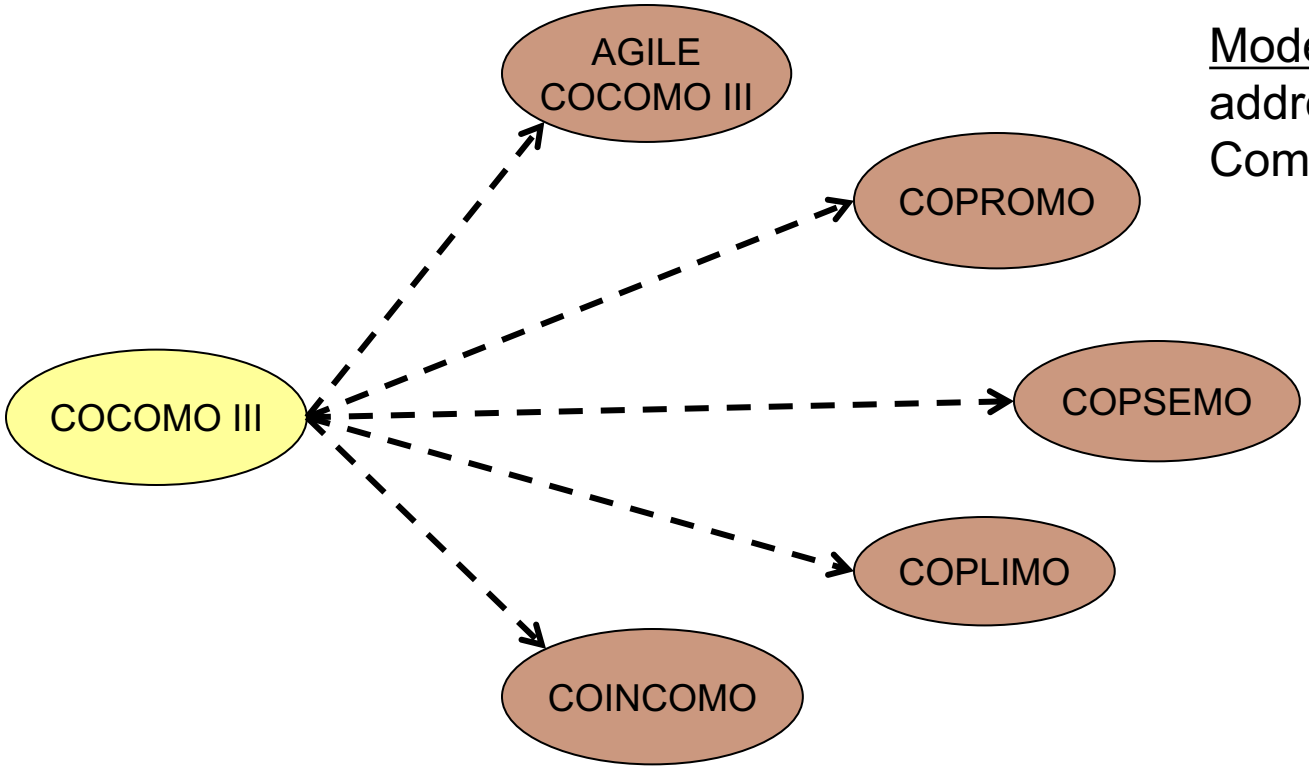
- Example: Net-centric military operations
- Size, Complexity: Very high
- Typical Change Rate/Month: Mixed parts; 1-10 %
- Criticality: Very high
- NDI Support: Many NDIs, some in place
- Organizational Personnel Capability: Related experience, medium to high
- Activities: Full ICSM; extensive multi-owner team building, negotiation. Full ICSM; large ongoing system/software engineering effort
- Time/Build: 2-4 months
- Time/Increment: 18-24 months

ICSM Common Cases Not In Scope

- Case 1: Use Non-Developmental Item (NDI)
- Case 2: Agile
- Case 3: Architected Agile
- Case 4: Formal Methods
- Case 7: NDI-Intensive
- Case 10: Family of Systems
- Case 11: Brownfield

- Modeling opportunities for these cases

COCOMO® III Suite of Models Concept



Model Extensions to address other ICSM Common Cases

Legend:
Model has been calibrated with historical project data and expert (Delphi) data
Model is derived from COCOMO III

COCOMO III Effort & Schedule Estimation Model

$$\text{Effort (PM)} = A * \text{Size}^{E} * \text{Product(14 Cost Drivers)}$$

$$E = B + \text{Sum(5 Cost Drivers)}$$

$$\text{Schedule (M)} = C * \text{PM}^{F} * \text{SCED\%/100}$$

$$F = D + 0.2(E-B)$$

Where:
A, B, C, D are constants determined by calibration
E represents (dis)economies of scale and project-wide scale factors

COCOMO III Defect Introduction and Removal Model

Defect Introduction (DI) = A * Size^{E} * Product(DI Drivers)

E = Initially set to 1.0

Residual Defects = C * DI * Product(1 – DRF)

DRF: Defect Removal Fraction from 3 profiles:

1. Automated Analysis
2. People Reviews
3. Execution Testing

Workload Sizing

- **The amount of development work to be done is expressed as either a functional or product size**
  - Software Requirements
  - Function / SNAP Points
  - Fast Function Points
  - Automated Function Points
  - COSMIC Points
  - Use Case Points
  - Source Lines of Code
- **The desire is for COCOMO III to use different size types organically as a size input**
  - **Want to move away from converting one size type into Source Lines of Code, e.g. Function Points to Source Lines of Code**

Model Manual Overview

- Tour of the draft COCOMO III Model Definition Manual

Topics

- Required Security Implementation results
- New Nominal
- Model Definition Manual - Beta
- **Next steps**
  - **Working group review of the model**
  - **Rosetta stone COCOMO II ➔ COCOMO III**
  - **Model performance analysis**
  - **Data collection**
    - **Form**
    - **Plan**