fissea
Federal Information Systems Security Educators' Association
AWARENESS • TRAINING • EDUCATION

Developing Construction Workers for Securely Built Software

James R Lindley
CISSP, ISSAP, ISSEP, ISSMP, CISA, PMP, SSE-CMM
Team Chief, IRS Penetration Testing and Code Analysis

Tuesday, March 18, 2014,11:05 -11:40

Three Types of Security Personnel

1. Those who **advise** on security
2. Those who **audit** security
3. Those who **create** security

Types 1 and 2 are usually in cybersecurity.
Type 3 is in application development and operations.

- Most academic education and training efforts involving whole curricula are focused on types 1 and 2.
- Academic security education and training is usually grafted into non-security curricula for type 3 via non-integrated classes.

<u>We’re going to look at a suggested better way to train type 3.</u>

What is IRS PTCA

- Largest group of federal civilian code analysts and penetration testers outside DHS
- Conducts automated static source code analyses of source code placed on IRS systems
- Coordinates with penetration testers for dynamic White Box code penetration testing
- Penetration tests of all major IRS applications and other code sets as directed

An Emergent Quality

- If software security is an **emergent quality**, from what does that quality emerge?

- **Security quality will not emerge unless software project managers recognize and demand the skills and tools relevant to that quality.**

Metaphors Affect Understanding

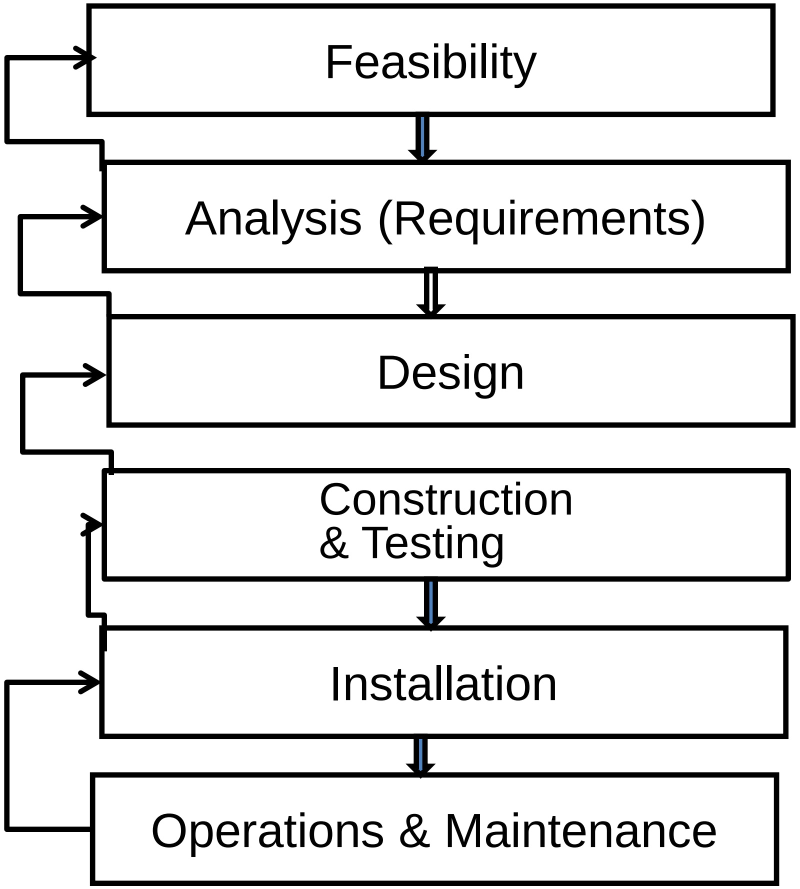
- Software development as a construction project
- Architectural perspective vs. blueprints
- Discipline specialization

Methods

- Grouping Stakeholders
- Separating Requirements from Specifications
- Data Design Documentation
- Functional Design by Atomic Function
- Supporting the Project Manager
  - Earned Value Management
  - Quality Assurance

Models - Sequential vs. Iterative

(Looped) Sequential



Feasibility

Analysis (Requirements)

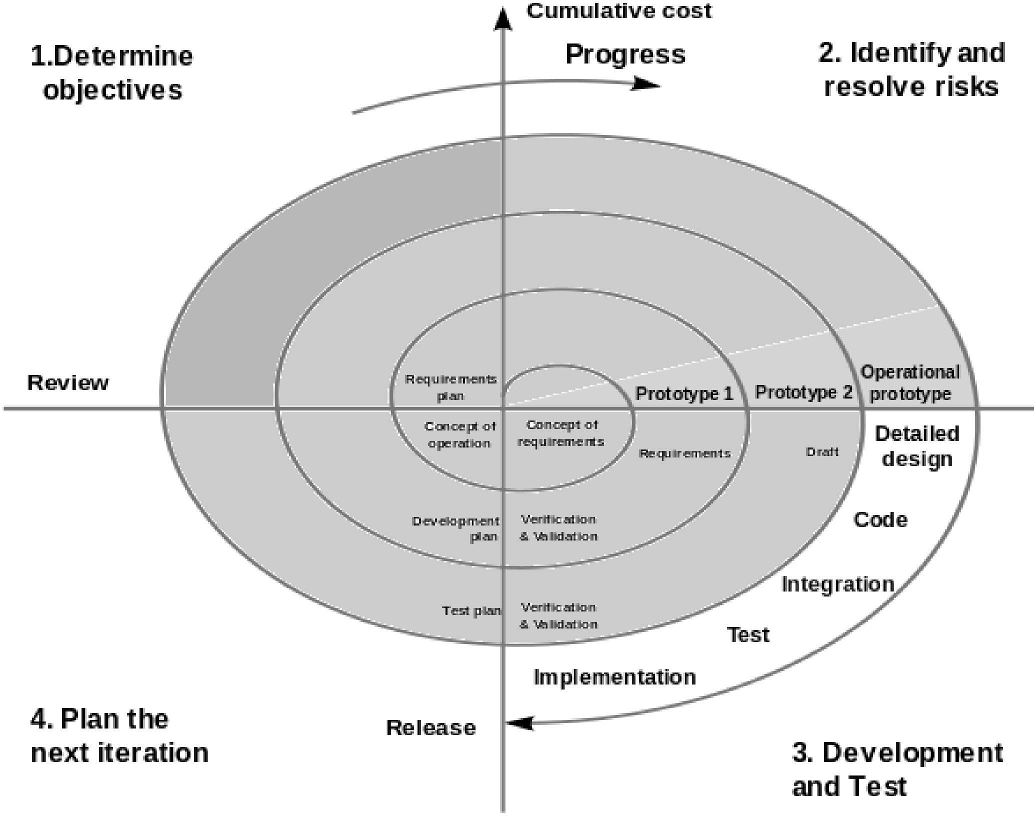
Design

Construction & Testing

Installation

Operations & Maintenance

Boehm and Agile



Agile manifesto

- We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
  - **Individuals and interactions** over processes and tools
  - **Working software** over comprehensive documentation
  - **Customer collaboration** over contract negotiation
  - **Responding to change** over following a plan
- That is, while there is value in the items on the right, we value the items on the left more.

Comparison of Models

Suitability of different development methods		
Agile home ground	Plan-driven home ground	Formal methods
Low criticality	High criticality	Extreme criticality
Senior developers	Junior developers	Senior developers
Requirements change often	Requirements do not change often	Limited requirements, limited features see Wirth's law
Small number of developers	Large number of developers	Requirements that can be modeled

Where We Are

- Strict Straight Waterfall Model
- An Implementation Phase Desert
- Waivers and Deviations
- Little Training In Security Quality

Fish And Ladders

- Have to work within the Straight Waterfall Enterprise Life Cycle model (policy)
- **Collaborative phasing** between the architectural phases (Requirements, Specification, and Design)
- Collaborative phasing is how the fish build and climb the waterfall ladder.
- **A change in practice, not a change in policy.**

Lessons Learned

- Agency cyber security team performs source code security scan for project exit approval
- Lesson: pushback from software project managers
  - Action: application development executives brought on board
  - Action: software project managers offered source code scanning tool for in-development software
  - Action: training on tool and security assessment for code writers
- Lesson: project managers and phase practitioners have weak software project management skills
  - Action: Develop course to teach secure software construction to project managers
  - Action: Develop courses for each of the phase practitioners

Project Manager Pushback

- Action: application development executives brought on board
- Action: software project managers offered source code scanning tool for in-development software
- Action: training on tool and security assessment for code writers

Weak Software Project Skills

- Action: Develop course to teach secure software construction to project managers
- Action: Develop courses for each of the phase practitioners
  - Requirement Elicitors
  - Specification Writers
  - Designers (Data and Software)
  - Code Writers
  - Quality Assessment

Approach to Training

- ***Craft unionism** refers to organizing … workers in a particular industry along the lines of the particular craft or trade that they work in by class or skill level. It contrasts with **industrial unionism**, in which all workers in the same industry are organized into the same union, regardless of differences in skill.*

A Human Capital Crisis in Cybersecurity
Technical Proficiency Matters

- There are continuing efforts by federal agencies to define an information technology (IT) security work force improvement program based on role definitions
- **I contend:** There is a lack of adequate detail in defining specialized IT security roles, especially as understood by managers without a security background or training.

- Center for Strategic and International Studies (CSIS) Report

Points To Ponder

- Simple evolves into Complex
- Complexity generates specialization
- Applications become APPLICATIONS
- Everybody wants to design, nobody wants to build
- Academia produces architects and engineers

- BUT…there is no degree in plumbing!

I am a dry pipe plumbing inspector

- Static source code analysis (dry pipe)
- Penetration testing (wet pipe)
- Design assessment (Architecture and Civil Engineering)
- Every stage of "plumbing" has a specialized creator and a specialized inspector
  - Requirements
  - Specification
  - Design
  - Code writing
  - Install and configure
  - Operations
  - Decommission

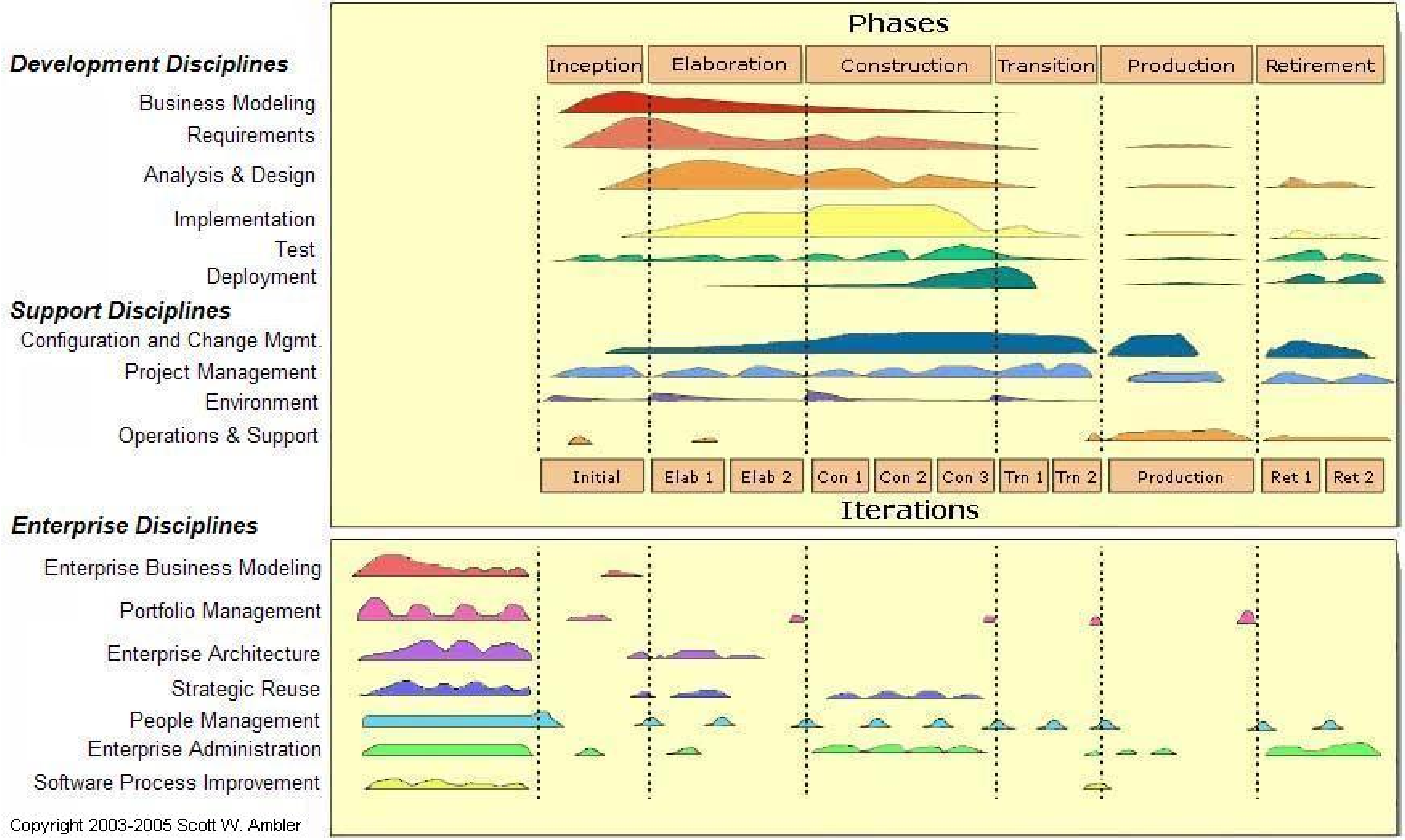
If You Build It Correctly, Security Will Come

- If software security is an emergent quality, ***from what*** does software security emerge?

The quality of all surrounding processes

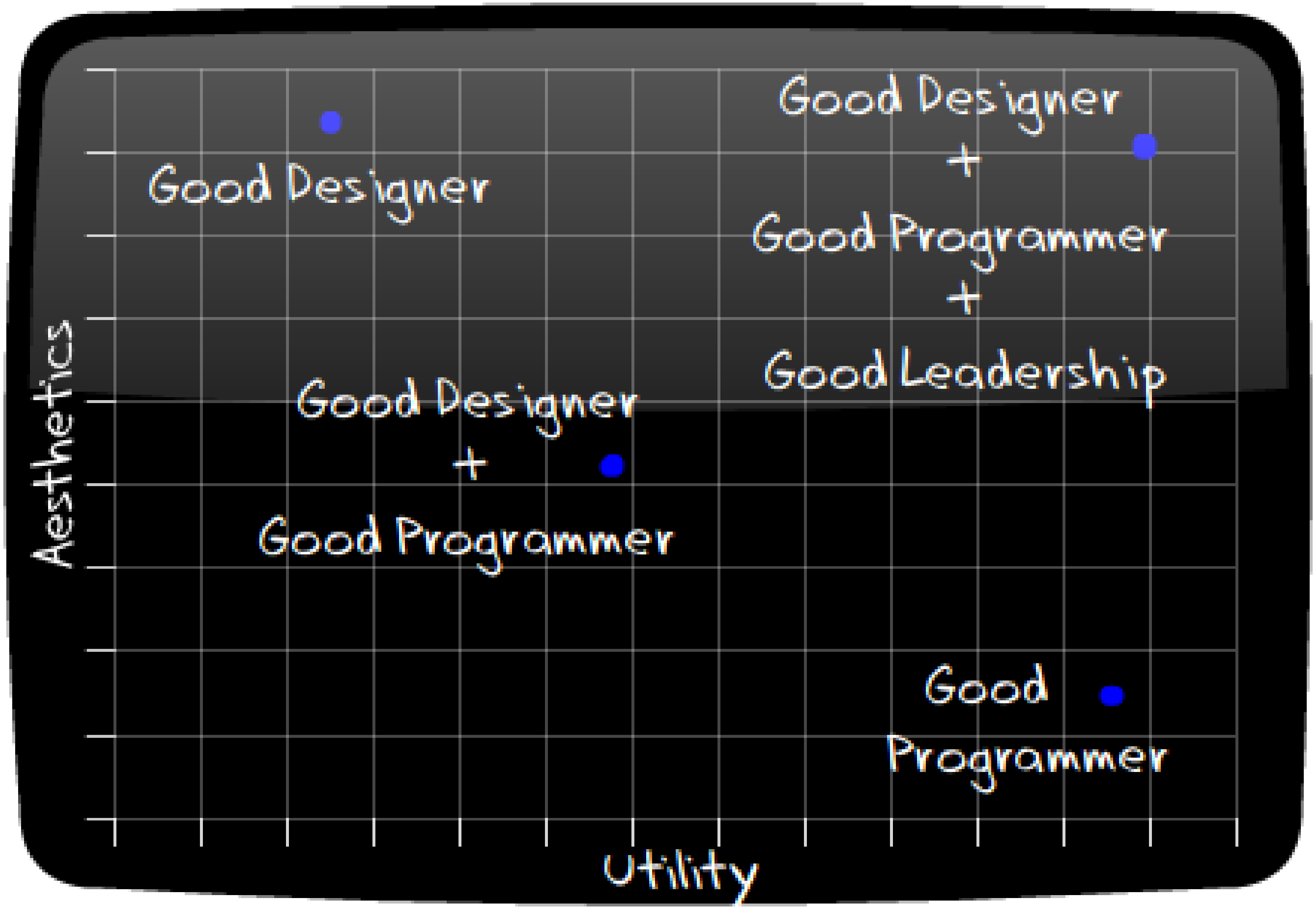
- A failure in any phase means a failed project.

A Team of Craftsman



Copyright 2003-2005 Scott W. Ambler

Software is a Synergistic Effort



FORGET DEVELOPER!!!!!
Think Code-Writer

- Requirements Elicitor (Security policy)
- Specification Writer (Security Engineer)
- Application and Data Designers (Security Architects)
- Code Writers (Code Analysts, Pen Testers)
- Installation and Configuration (Pen Testers)
- Quality Assurance Testers (functional and non-functional)
- Operations and Operational Security (Security Monitors)
- Decommission (Data and application destruction specialists)

The Blue Collar Office Worker

Requirements Elicitors

- Focused on **problem space**
- Group Stakeholders
  - Regulatory
  - Environmental
  - Customer
  - Users
  - Project Team
- Elicitation skills
  - Strategic Debrief
  - Interrogation
  - Document Research and Analysis
- Reading and writing Skills
- Gregarious personality

Specification Writer

- Focused on **solution space**
- Mathematically resolvable solution descriptions for problem requirements

- Active writing skills
- Formal methods
- Detail-oriented perfectionist

Designers

- Data designers
  - Schema
  - Data Item Dictionary
  - Detail-oriented
- Code Designers
  - UML skills
  - Function Point Design = Earned Value Management
  - Vision of the whole

Code Writers

- Adopt a standard of secure coding practices
- Teach the coding standard
- Teach code evaluation tools and skills
- Demand the standard by evaluating employees using the standard
- Detail-focused

Secure Coding Quality Assessment

- Study and learn from the Building Security In Maturity Model (BSIMM)
- Train and use a security evaluation team as a part of the application development (AD) team and processes.
- Specifications = development of test scripts and scenarios
- Teach the evaluation tools and standards
- Collaboration between the AD team and the agency cyber-security penetration testing and code analysis team

How?

- Identify currently available skill sets in application development and cybersecurity personnel.
- Establish mentoring programs using what you already have available
- Find **or develop** SPECIALIZED training (look to community colleges and in-house programs)
- Require contractual trainers to craft training IAW the organization policies
- GRANULATE your role definitions

A Suggested Activity Timeline

- Start and train a software security quality team
- Establish a software security gate in EACH project process phases
- Offer tools and training to code writing teams
- Train project managers in software process quality with a focus on securely constructed software
- Train architectural and quality assessment practitioners in designing and testing for secure software construction
- Use documented standards for secure software

Document or Secure?

- You must do both, but don't mistake one for the other
- Government example is FISMA, a mandate to document the security that we are not mandated to produce.

Discussion