Microsoft SDL: Agile Development

Nick Coblentz, CISSP
Senior Consultant, AT&T Consulting
Nick.Coblentz@gmail.com
http://nickcoblentz.blogspot.com
http://www.twitter.com/sekhmetn

Microsoft SDL For Agile Released

Source: http://blogs.msdn.com/sdl/archive/2009/11/10/announcing-sdl-for-agile-development-methodologies.aspx

Agenda

- Presenter Background
- Microsoft SDL (High-level)
- Agile Development (High Level)
- Traditional SDL Activities and Pain points for Agile development
- SDL-Agile Key Concepts
- SDL-Agile in Detail
- Tips for Making SDL-Agile Manageable

Bio

- AT&T Consulting:
  - Application Security
    - Penetration testing
    - Code review
    - Architecture and design reviews
    - Application security program development
    - Secure development methodology improvement

Projects and Publications

- ISSA Journal: Web Application Security Portfolios
- SAMM Interview Template
- Turn Application Assessment Reports into Training Classes
- Arshan Dabirsiaghi's Struts 2 Gap Analysis Whitepaper
- Observed Secure Software Development Stages
- Vulnerability Tracking, Workflow, and Metrics with Redmine
- Using Microsoft's AntiXSS Library 3.1
- Internal AppSec Portals
- Struts 2 Security Addons Code Repository
- JITSecure Code - light weight code review as you program

Microsoft Security Development Lifecycle (SDL)

Components:

- Best Practices
- Processes
- Standards
- Security Activities
- Tools

Goal:

"minimize security-related vulnerabilities in the design, code, and documentation and to detect and eliminate vulnerabilities as early as possible in the development life cycle."

Which Software?

SDL applies to software that:

- Is used in Business environments
- Stores or transmits PII
- Communicates over the Internet or other networks

Microsoft® SQL Server® 2008

Microsoft® Office Live

bing™

Windows Live™

Source: Microsoft's Product Website

SDL Principles and Process

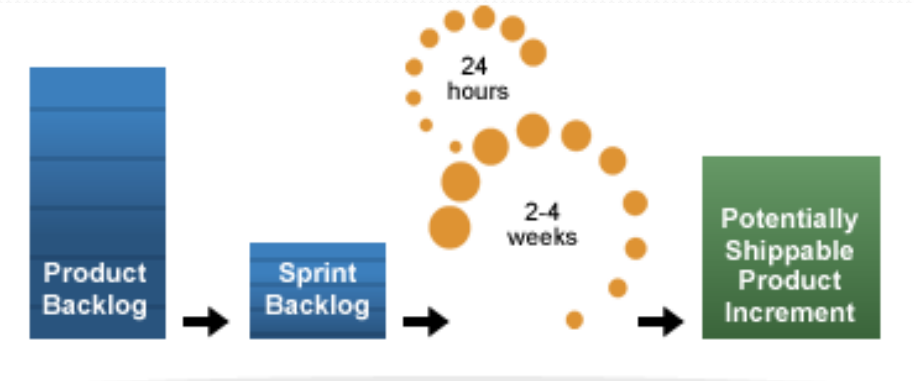
SD3+C

- Secure by Design
- Secure by Default
- Secure in Deployment
- Communications

PD3+C

- Privacy by Design
- Privacy by Default
- Privacy in Deployment
- Communications

Agile Development



Source: http://www.scrumalliance.org/pages/what_is_scrum

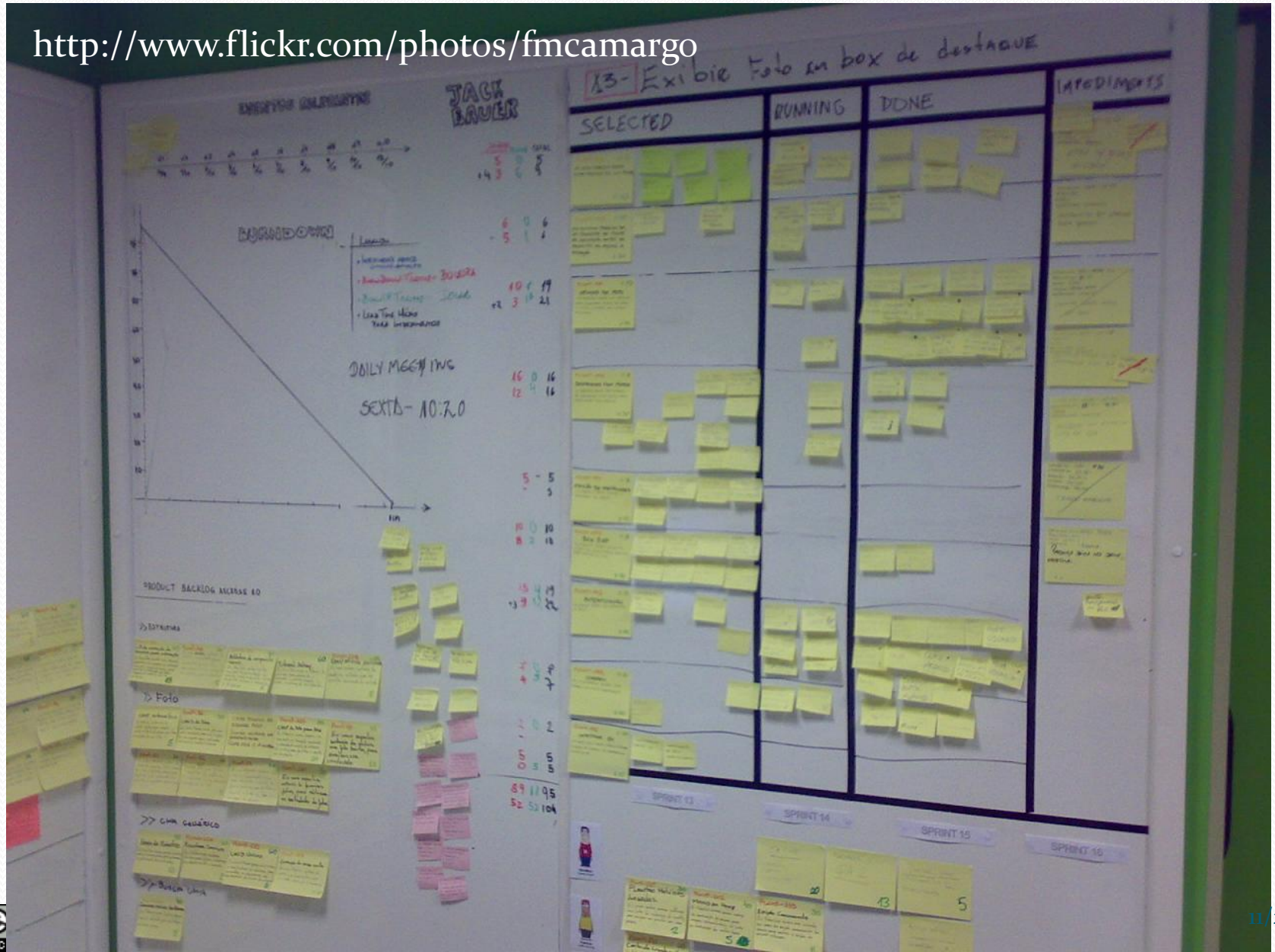
- Cross-functional, self-organizing teams
- Short, time-boxed development iterations
- Delivery of small functional stories
- No *extensive* up front design or documentation

Planning and Design

Planning and Design (cont.)

http://www.flickr.com/photos/acarlos1000

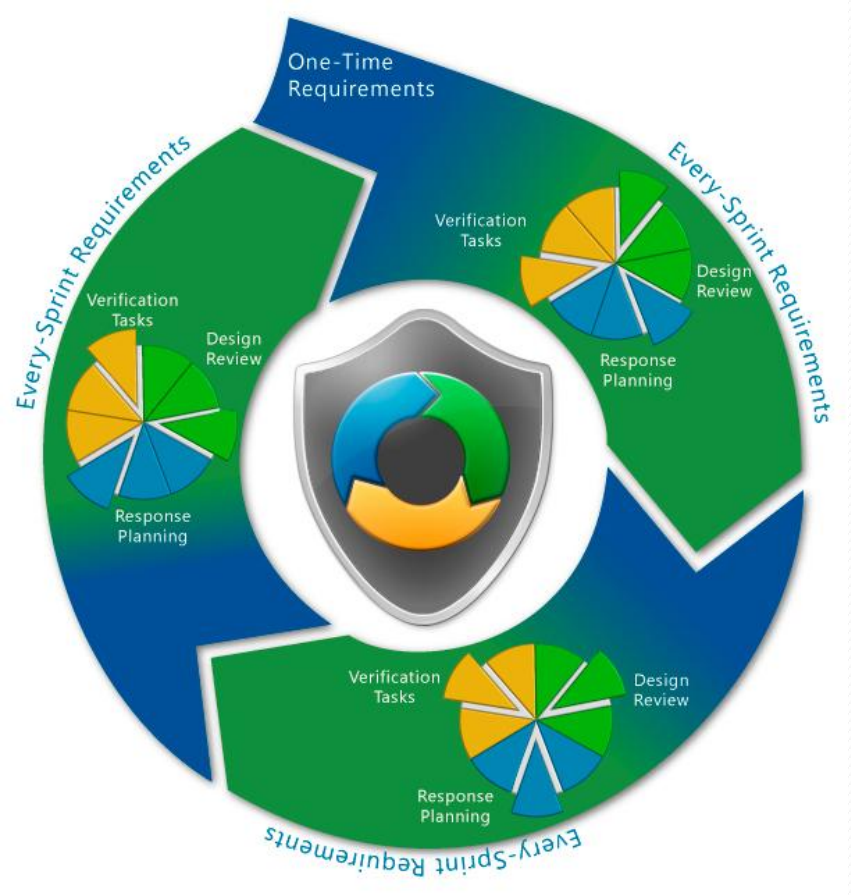
User Stories and Documentation



Traditional SDL Pain Points for Agile

- Can't complete all SDL activities for each sprint
- Requirements, architecture, and design evolves over time
- Threat model becomes dated quickly
- Data sensitivity and connection to third parties may not be immediately known

Microsoft SDL For Agile Development



Source: Microsoft SDL v4.1a

- SDL Requirement Categories:
  - Every-Sprint
  - Bucket
    - Verification Tasks
    - Design Review Tasks
    - Response Planning Tasks
  - One-Time

SDL-Agile Appendix

Appendix Q: SDL-Agile Bucket Requirements

Bucket A: Security Verification

Title	Requirement/ Recommendation	Applies to Online Services	Applies to Managed Code	Applies to Native Code
Debug the application with the Application Verifier enabled	Requirement			X
Disable tracing and debugging in ASP.NET applications	Requirement	X	X	
Investigate and service any reported /GS crashes	Requirement			X
Perform ActiveX control fuzzing	Requirement	X		X
Perform attack surface analysis	Requirement	X	X	X
Perform binary analysis (BinScope)	Requirement	X	X	X
Perform COM object testing	Requirement			X
Perform cross-domain scripting testing	Requirement	X	X	X
Perform file fuzz testing	Requirement	X		X
Perform RPC fuzz testing	Requirement	X		X
Conduct in-depth manual and automated code review for high-risk code	Recommendation	X	X	X
Perform data flow testing	Recommendation	X	X	X

SDL-Agile Appendix: Deadlines

Appendix R: SDL-Agile One-Time Requirements

Title	Requirement/ Recommendation	Completion Deadline (months)	Applies to Online Services	Applies to Managed Code
Avoid writable PE segments	Requirement	6	X	
Configure bug tracking to track the cause and effect of security vulnerabilities	Requirement	3	X	X
Create a baseline threat model	Requirement	3	X	X
Determine security response standards	Requirement	6	X	X
Establish a security response plan	Requirement	6	X	X
Identify primary	Requirement	1	X	X

Every-Sprint SDL Requirements

"…so essential to security that no software should ever be released without these requirements being met."

Examples:

- Update the threat model
- Communicate privacy-impacting design changes to the team's privacy advisor
- Fix all issues identified by code analysis tools for unmanaged code
- Follow input validation and output encoding guidelines to defend against cross-site scripting attacks

Bucket SDL Requirements

- Teams prioritize the pool of tasks over many sprints
- Each sprint, one task from each bucket completed
- Each tasks must be completed at least every 6 months

Examples:

- Security Verification Tasks
  - Run fuzzing tools
  - Manual and automated code review
- Design Review Tasks
  - Conduct privacy review
  - In-depth threat model
- Response Planning Tasks
  - Define security/privacy bug bar
  - Create support documents

One-Time Requirements

Why?

- Repetition not necessary
- Must occur at the beginning of the project
- Not possible at the beginning of the project

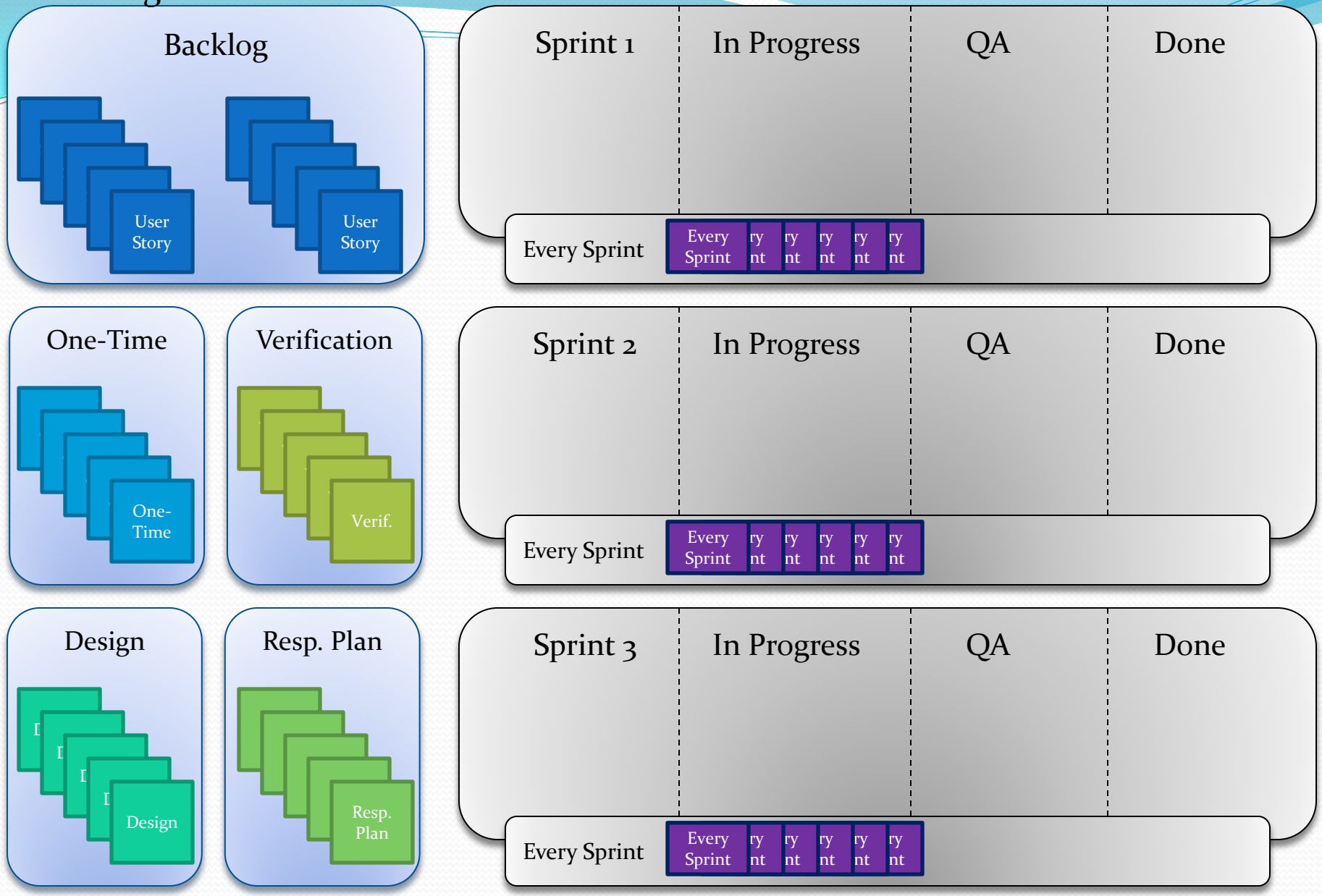
Examples:

- Configure bug tracking system (3 months)
- Identify security/privacy experts (1 month)
- Baseline threat model (3 months)
- Establish a security response plan (6 months)

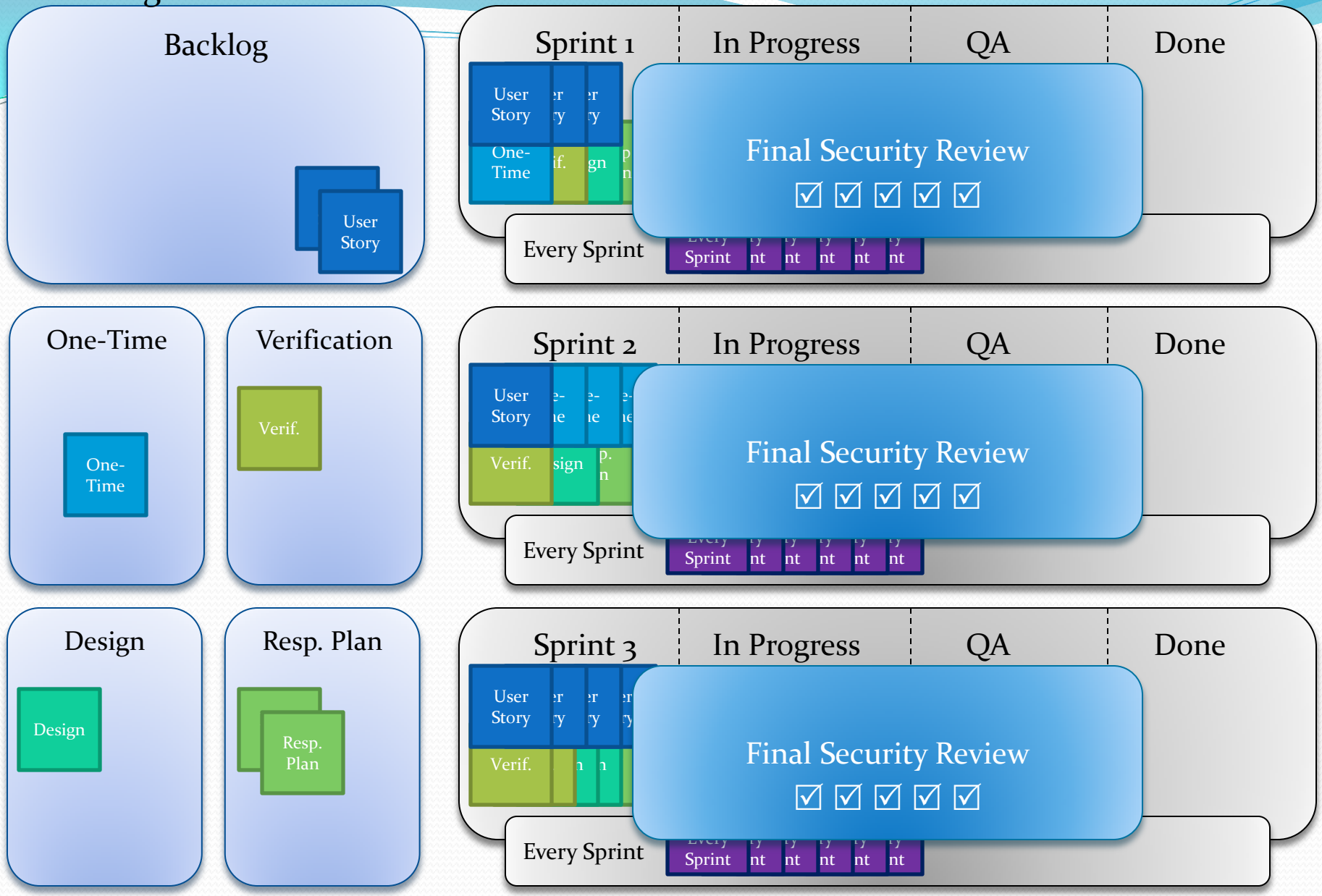
Final Security Review

- Occurs at the end of every sprint
- Checklist:
  - ☑All every-sprint requirements have been completed
  - ☑No one-time requirements have exceeded deadline
  - ☑At least one requirement from each bucket category has been completed
  - ☑No bucket requirements exceed the six month deadline
  - ☑No security or privacy bugs are open that exceed the severity threshold

SDL-Agile Process Demonstration



SDL-Agile Process Demonstration



Making SDL-Agile Manageable

- Automation
  - CI server, unit testing that include security
- Tooling
  - Automated code analysis and pen testing tools
- Continuous updates to the threat model
- Documented standards
- Security training
- Light on security artifacts

TeamCity SDL-Agile Demonstration

- TeamCity Continuous Integration Server
  - Continuous Build
    - FxCop
    - CAT.NET
    - BinScope
    - Commercial Penetration Testing or Code Review Tools
    - Security Unit Testing

Summary and Questions

More Information:
http://www.microsoft.com/sdl

Nick Coblentz, CISSP
Senior Consultant, AT&T Consulting
Nick.Coblentz@gmail.com
http://nickcoblentz.blogspot.com
http://www.twitter.com/sekhmetn

- Microsoft releases SDL-Agile Guidance in Nov. 2009
- Treats SDL Activities like team-prioritized User Stories
  - 3 Categories: One-time, Every-time, and Bucket
- Increased success with the implementation of training, automation, tools, and standards