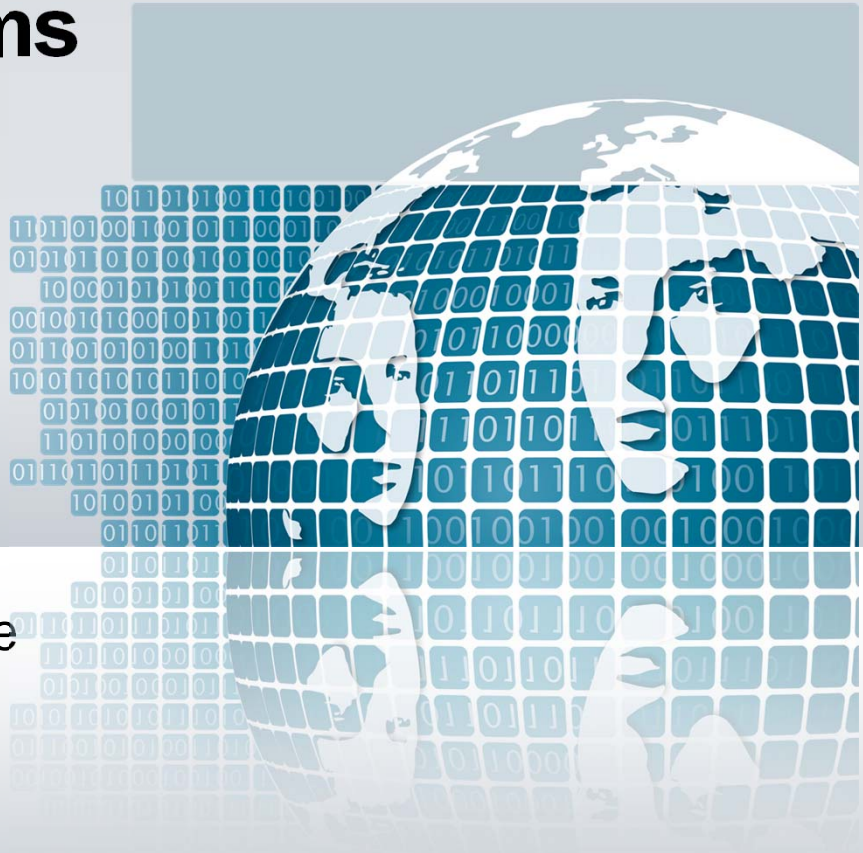
SIEMENS

Standardizing Speed and Security for Software-based Systems



Frances Paulisch, CT Software Initiative

May 9, 2012

frances.paulisch@siemens.com

SIEMENS

Characteristics of Siemens

Siemens:

- Active in many different domains (energy, healthcare, industry, infrastructure and cities)
- Over 15,000 software engineers worldwide
- ca. 60% of our sales come from products that are based on software (but typically embedded into a system)
- large, often multi-site, projects
- increasing functionality realized in software
- quality attributes (like performance, scalability, security, safety,…) are of high importance
- an “integrated technology company”
- active member of the software engineering community

i.e. software-based systems are very important
(both functionality and quality attributes)

SIEMENS

Software Initiative Curriculum

A qualification and training program, to address key critical roles in the development of software-based systems.

It recommends a holistic, architecture-driven, iterative approach. A set of “guiding principles” summarize the main recommendations.

The SWI Curriculum is a set of qualification programs is established as a Siemens Global Core Learning Programs

Qualification programs (including certification) currently available:
- senior software architect (SSWA)
- software architect (SWA)
- system architect (SyA)

Other trainings:
- **Requirements Engineering for product managers and for project managers**
- **R&D manager workshop**
- **Security, …**

Core Learning Programs
driven by the Software Initiative (SWI)

SIEMENS

Core Learning Programs

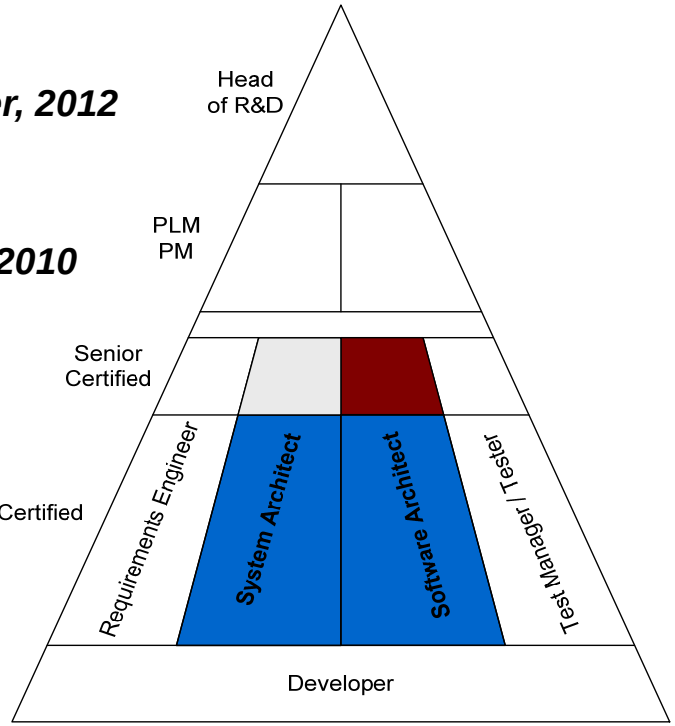
R&D Manager, 2012

RE for PM /
RE for PLM, 2010

SSWA, 2007

SyA, 2012

SWA, 2009



Runs:

- 5 runs of SSWA completed
- 12 runs of SWA completed
- 1st run of SyA starts in 2012

Networking (missionaries):

- Technical Debt
- Internal Code Quality
- Test Driven Domain Modelling
- IT Security
- Open Source
- Model Driven Development
- "collective (neutral) opinion on topic x"

Siemens-wide "guidance"

- Siemens-wide reporting of # SSWA/SWAs per Division/BU
- Recommendation to use SSWA in certain projects
- Guiding Principles (same set to most involved roles)

Software Initiative: Guiding Principles

(different focus depending on role, but apply to all)

SIEMENS

1. Architecture as well as the continuous governance of it is the key throughout the whole lifecycle as well as across releases.

2. Build on existing basis where feasible (from technical and business perspective) and be able to recognize when such reuse is not suitable.

3. Avoid unnecessary technological platform development and use technical standards and products available on the market.

4. The product (lifecycle) manager in product business and the project manager in solution business is and must act as the owner of the main requirements and quality characteristics.

5. Pay particular attention to specifying, testing , and realizing non-functional requirements (NFRs), often overlooked but are extremely important.

6. Be prepared and able to handle changing requirements, but be aware about the risk of late changes.

7. Synchronize well across the technical disciplines (software, mechanics, electronics).

8. Work together truly as a team, avoid “silo” thinking, be willing and able to speak and understand the other roles and disciplines.

9. Work iteratively and test-driven, foster defect prevention from the beginning, and strive to identify and resolve technical and business risks early. Getting real and early feedback, both from customer and from realization team, is essential.

10. Structure the system to avoid unnecessary complexity, and to actively enable and support multi-site development.

11. Strive for transparency and base decisions on clear business / technical (not political) reasons.

12. Do not underestimate the importance of soft skills, these can be particularly important for convincing and motivating.

Only a fully integrated secure development lifecycle ensures protection against targeted attacks

SIEMENS

Security Strategies	Example Activities	Results
Security Features	▪ Firewalls ▪ Cryptography ▪ Authentication models	▪ Insufficient security level ▪ Security defects in features, that are not security-suspect
Singular, Ad-hoc Activities	▪ Penetration testing in late development phases ▪ Use of secure coding guidelines without reviews	▪ Huge defect correction efforts ▪ Products are deployed even with severe security risks ▪ Some security risks are unknown
"Design for Security"	▪ Security is handled as another quality criteria ▪ Fully integrated in the development process ▪ Systematic engineering and management of development process	▪ Plannable security efforts ▪ Operational resiliency against attacks ▪ Reduced security risks

A model similar to CMMI is required to provide security guidance in a diverse environment

SIEMENS

Our Consid-erations

- Siemens develops a wide range of diverse products
- Development organizations have to define specific processes to fulfill business goals
- CMMI-DEV already successfully established within Siemens as process model to guide these processes
- A similar framework is needed for secure development guidance
- Existing best-practice collections (e.g. Microsoft SDL, OpenSAMM, BSIMM) are very helpful, just not on process level

Our Solution

- Plan for this to become a security extension to CMMI-DEV v1.3 for guidance for security in the development of products (currently under review by SEI/SEI Partners)

+SECURE

In four process areas, +SECURE defines requirements on the organization and the development process

SIEMENS

Process Area	Intention & Purpose
Organizational Preparedness for Secure Development (OPS)	▪ Establish capabilities to develop secure products and react to product security incidents
Security Management in Projects (SMP)	▪ Project activities to address security topics are identified, prepared, planned and managed. ▪ Evaluate and manage product security risks throughout the project.
Security Requirements and Technical Solution (SRT)	▪ Develop security requirements to meet the relevant stakeholders' security needs. ▪ Develop a secure architecture and design for the product according to security design principles ▪ Establish and maintain standards for secure product configuration. ▪ Implement the Secure product components and associated security support documentation
Security Verification and Validation (SVV)	▪ Ensure that selected work products meet their specified security requirements. ▪ Demonstrate that product or product components fulfill the security expectations when placed in its intended operational environment.

Implementation hints:
Example: SMP SG2 Manage product security risks

SIEMENS

SG 2 **Manage Product Security Risks**

The product security risks are managed throughout the project.

SP 2.1 **Establish Product Security Risk Management Plan**

Establish and maintain a product security risk management plan.

SP 2.2 **Perform Product Security Risk Analysis**

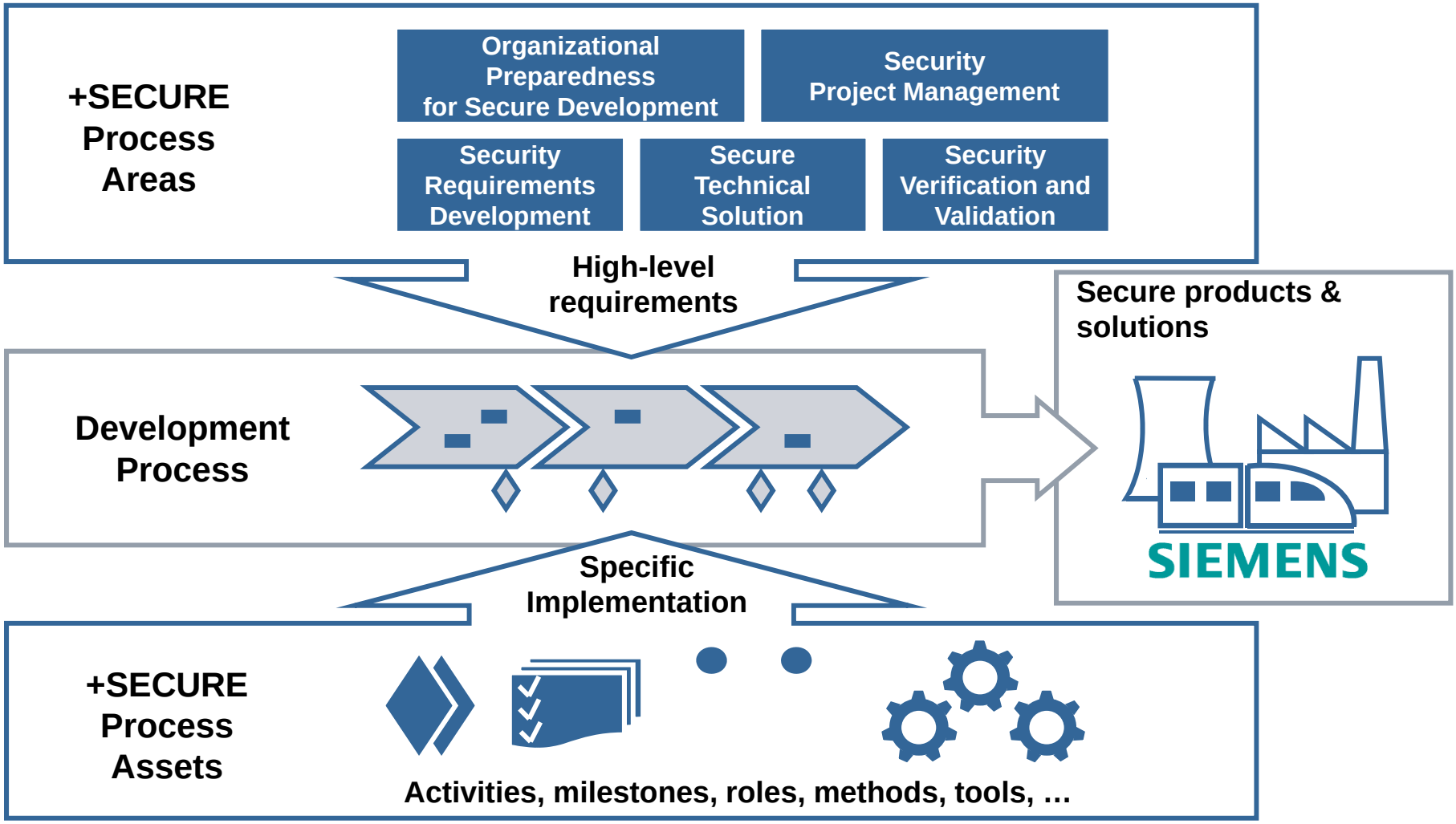
Identify and evaluate product security risks.

SP 2.3 **Establish and Implement a Product Security Risk Mitigation Plan**

Establish, maintain and implement a plan for the mitigation of product security risks.

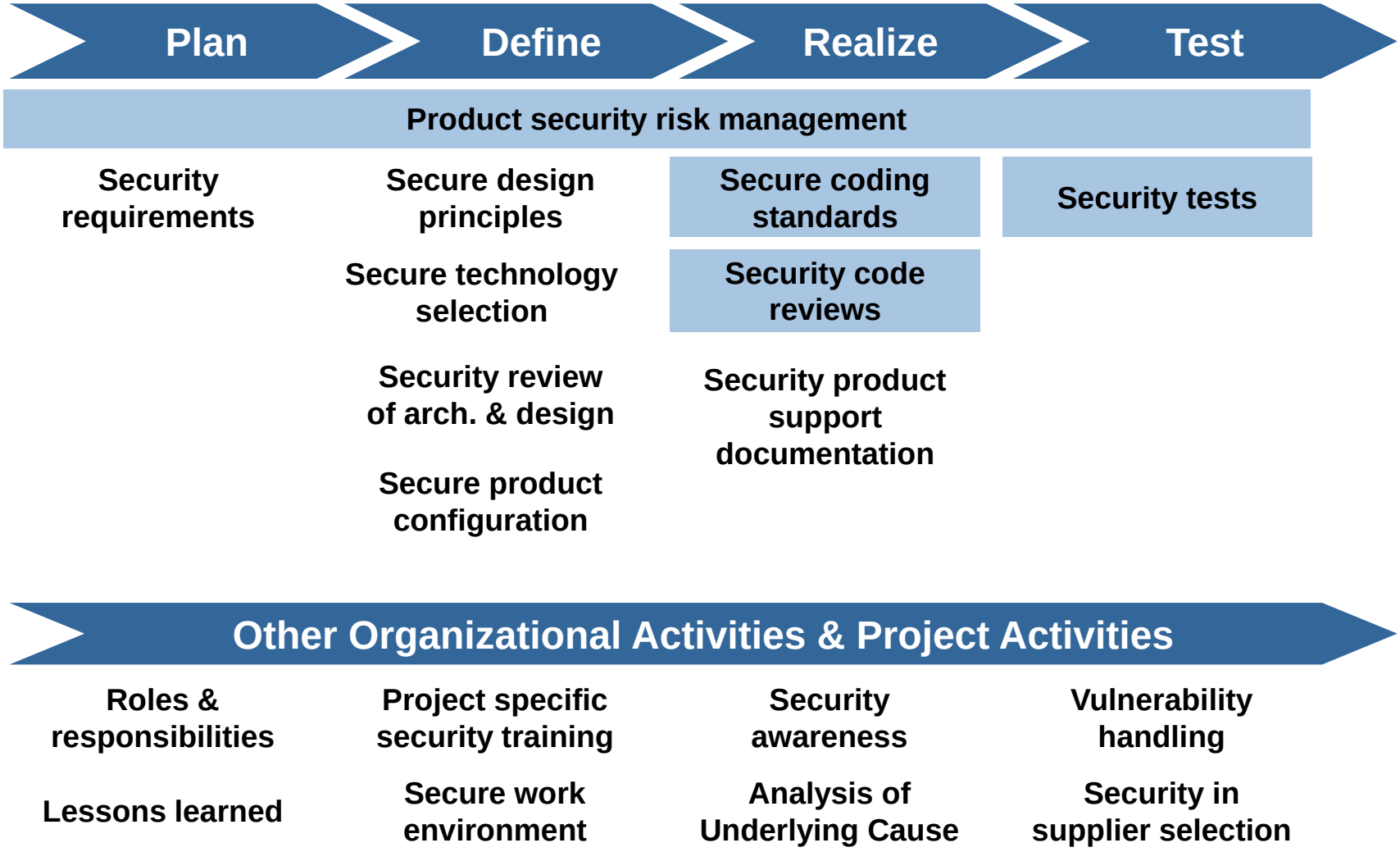
+SECURE defines security requirements for the development process

SIEMENS



Secure development activities – from security requirements to security testing

SIEMENS



SIEMENS

Be active in the relevant communities

Be active member of international security communities e.g. SAFECode, IEEE Software, IEEE Security&Privacy, conferences, workshops, also standards play an important role here.

Two central sources for guidance in this area are

- SAFECode (www.safecode.org) and
- BSIMM (www.bsimm.com).

Many organizations are active and follow activities in both.

The two approaches complement each other well. SAFECode offering a more prescriptive approach and BSIMM offering more of a (data-based) overview of which activities are well-established in the broad set of companies that are involved in BSIMM.

SIEMENS

Selected list of IEEE Software resources

Security & Privace: Promising Advances, Charles P. Pfleeger and Deborah M. Cooper, IEEE Software, Oct. 1997

Building Software Securely from the Ground Up, Anup K. Ghosh, Chuck Howell, James A. Whittaker, IEEE Software, Jan./Feb. 2002

Reducing Internet-Based Intrusions: Effective Security Patch Management, Bill Brykczynski and Robert Small, IEEE Software, Jan/Feb. 2003

Organizing Security Patterns, Munawar Hafiz, Paul Adamczyk, Ralph E. Johnson, IEEE Software, July/August 2007

Cybersecurity Econonomic Issues: Clearing the Path to Good Practice, Shari Lawrence Pfleeger, Rachel Rue, IEEE Software, Jan./Feb. 2008

Technology Transfer: A Software Security Marketplace Case Study, Gary McGraw, IEEE Software (Insights column), Sept./Oct. 2011

SIEMENS

Thank you – Any Questions?

Many thanks to:

Barbara Fichtinger, Peter Panholzer, Winfried Russwurm of Siemens Corporate Technlogy