9th ICCC in Korea, 2008

Secure System Integration Methodology

Satoshi HARUYAMA,
Toshiya YOSHIMURA, Naohisa ICHIHARA
NTTDATA Corporation

Contents

1. Background

- Issue of security in system integration
  - Security is not treated as primal factor in the traditional software engineering because;
    - Hard to define Security as system quality as it is sometimes subjective, obscure and relates various aspects of system
    - Complicated to resolve the interference among NFRs (Security, Performance, Efficiency, Reliability, Usability, Maintainability)

- Related works
  - Researches; UML-sec, Security Patterns, Secure Tropos
  - Vendor works; Microsoft, IBM, NTTDATA, … etc

We need effective and pragmatic methodology to assure integrate secure system

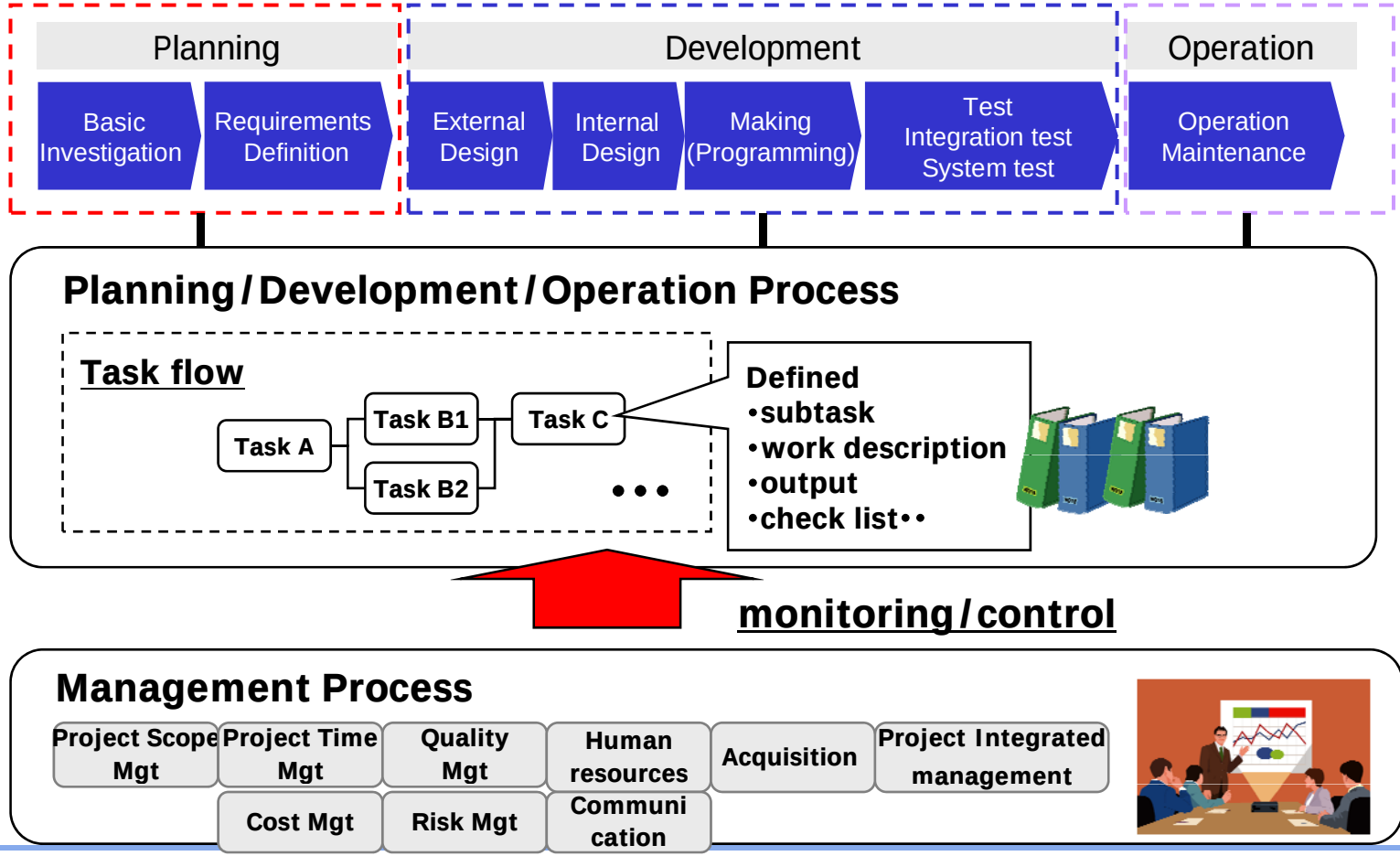
1. Background

- Standardization of system integration process
  - Defined for improving system quality as follows:
    - System life-cycle model
      - Process
      - Task
    - Standard process in system life-cycle model
      - Planning, Development, Operation process
      - Management process
      - Tailoring process

1. Background

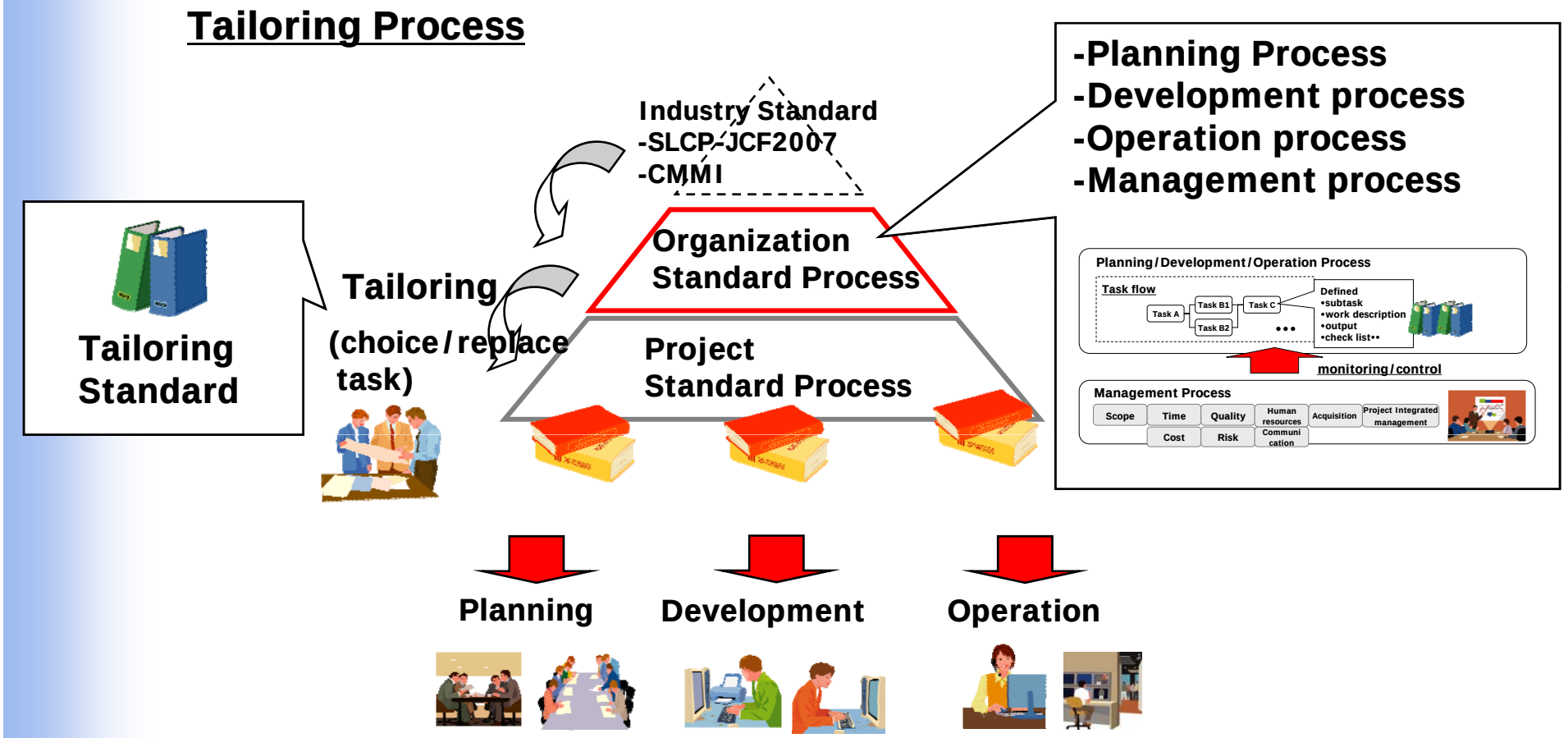
- Standardization of system integration process

System life-cycle model (image)



1. Background

- Standardization of system integration process



2. Our goal

- Goal:
  - *Establish the effective and pragmatic methodology to assure security of system.*

- Point 1: Plan the security of system **“as required”**
  - Identify the required security and its level for the system
  - Avoid spending security cost than needed
  - Agree the required security and cost with customer

- Point 2: Develop and operate securely **“as planned”**
  - Realize and maintain the security correctly as planned
  - CC concept based methodology which involves;
    - » “***Completeness***”, “***Consistency***” as well as “***Responsibility***”

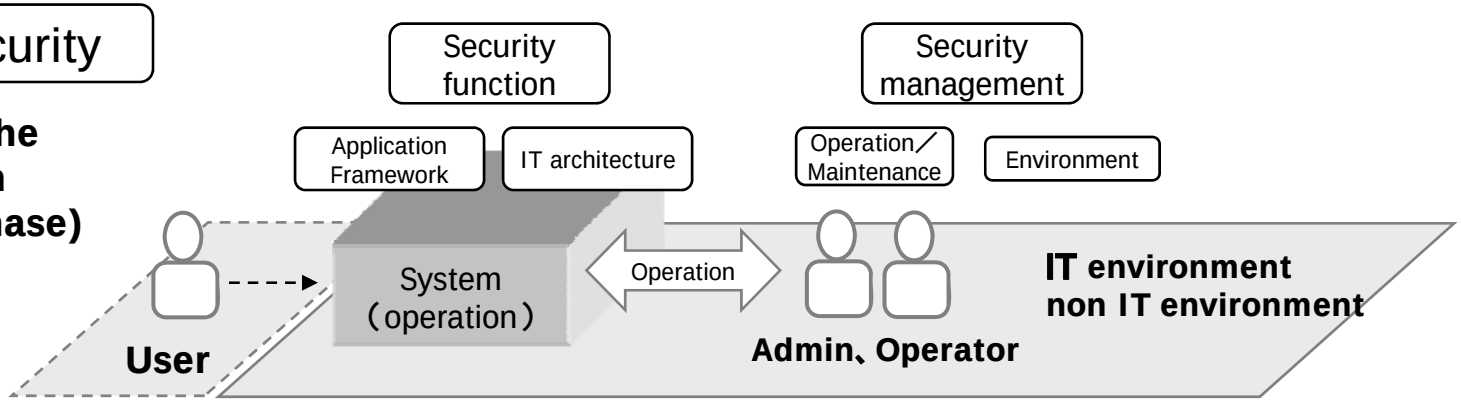
- Point 3: Aim to be more “**commonly used**”
  - Adopted to the existing development methodology
  - Define standardized tasks to develop security for all the developers

3. Our approach for secure system integration

- Scope
  - We categorize two type of security related to system integration.

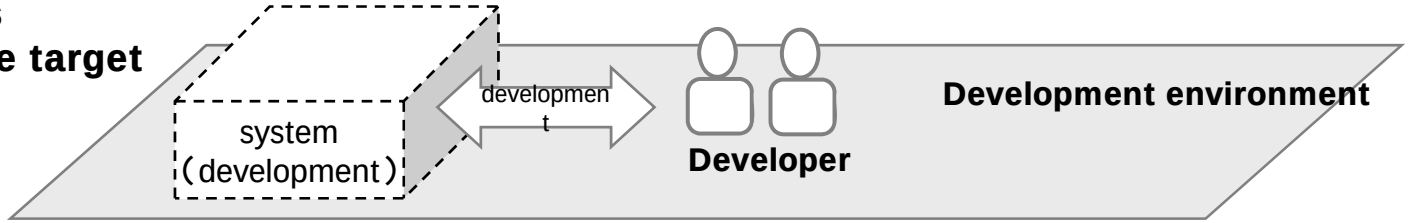
System security

Security for the target system (operation phase)



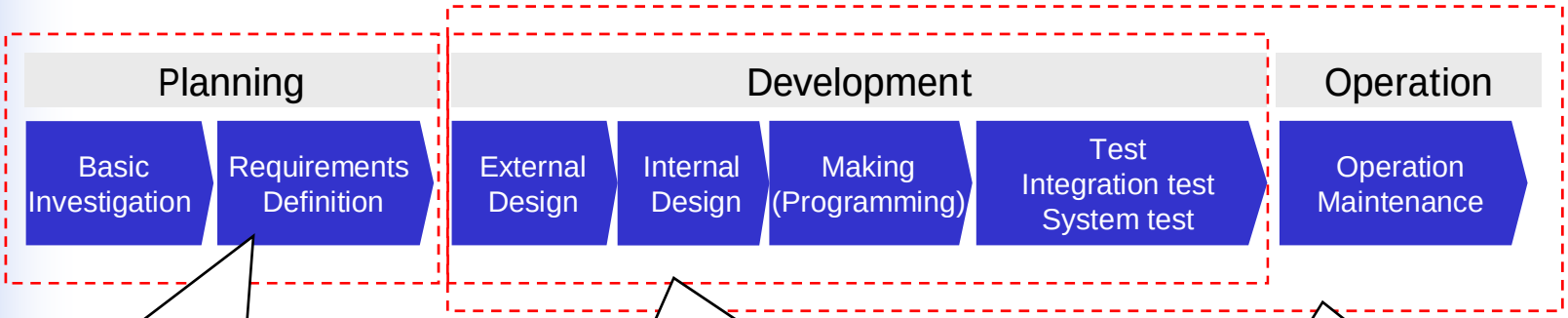
Project security

Security for the project that is developing the target system.



3. Our approach for secure system integration

- Overview of system security assurance



Agreement

•Check security policy, regulation, standard etc..
•Analysis security risk

•Agree with ;
-security scope
-required security level
-security requirement
-total cost

Realization

•Implement security specification and secure operation (rule, environment, procedure)
•Verify (test) security level agreed with customer (in planning process)
•Manage project (human resources, development environment and procedure)

Maintenance

•Monitor target system operation for keeping the security level, and response security incident throughout development and operation phase

3. Our approach for secure system integration

- Overview of project security assurance
  - Manage project for keeping security (a part of management process) as follows:

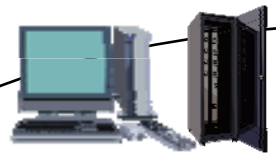
Development security
(ALC_DVS)

-Development security documentation shall describe all *the physical, procedural, personnel, and other security measures* that are necessary to protect the confidentiality and integrity of the TOE design and implementation

Refer to ISMS(ISO/IEC27002:2005)

"ISMS Aspects in Common Criteria Certificates for Development Sites", Bertolt Krüger,6th ICCC 2005

CM capabilities (ALC_CMC)
- Development documentation management



Development environment

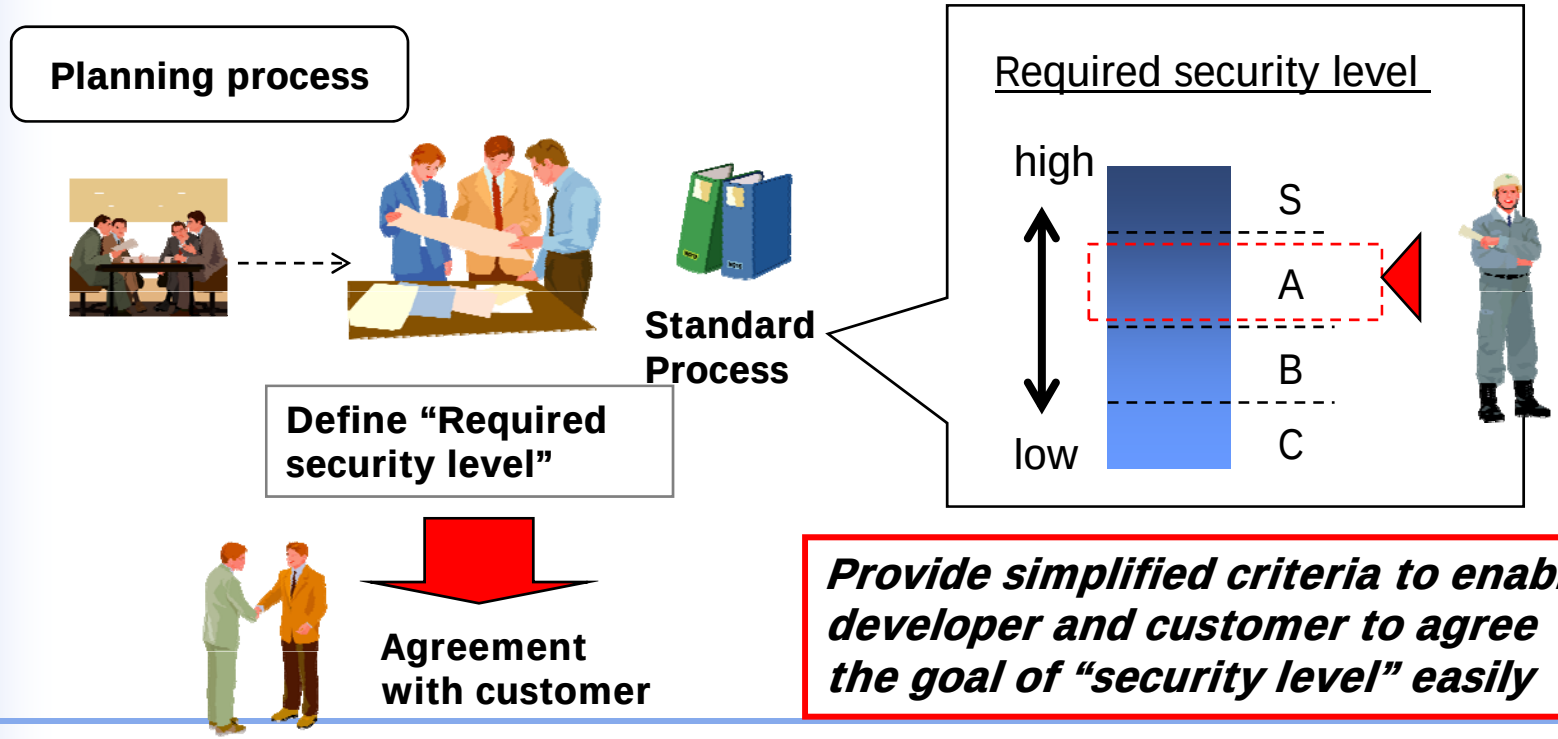
life-cycle model

Life-cycle definition (ALC_LCD)

-The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE

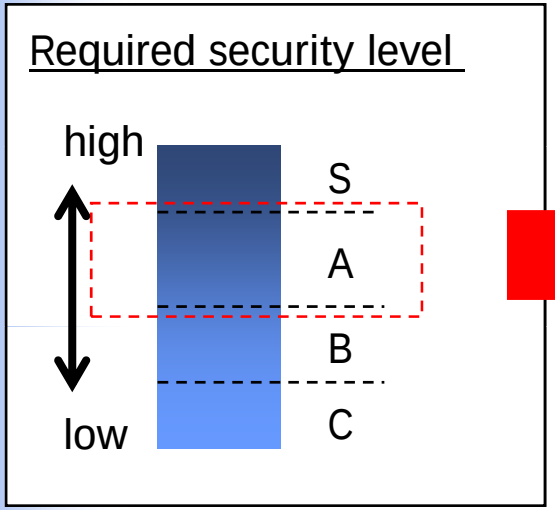
3. Our approach for secure system integration

- Required security level (1)
  - CC assurance approach is efficient to provide system security assurance
  - However, applying CC scheme to all project is not reasonable (project cost, time, human resource...)
  - Therefore, we apply the concept of "**Required security level**" based on simplified CC assurance scheme to our standard process



3. Our approach for secure system integration

- Required Security level (2)
  - Define tailoring rule according to “Required security level” in our standard process
  - Tailoring rule (choice/replace of security task) realize CC SAR scale (Scope, Depth, Rigour)
  - “Required security level” correspond to “simplified EAL”



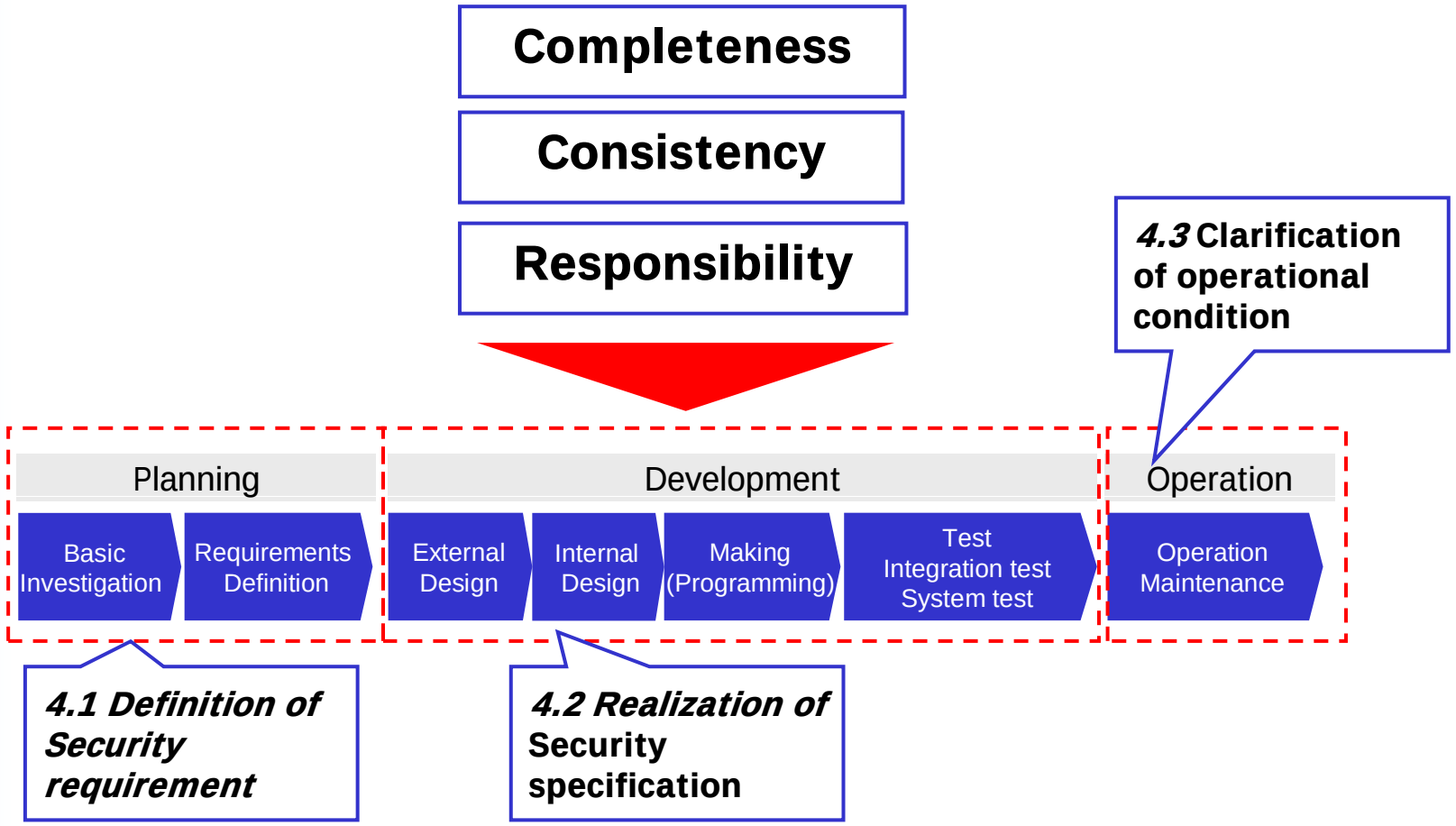
Tailoring Standard (image)

Required Task		Required Security Level			
		S	A	B	C
Scope	Lv1 (Critical subsystem)				
	Lv2 (All)				
Depth	Lv1 (Requirement)				
	Lv2 (Design)				
	Lv3 (Implementation)				
Rigour	Lv1 (Check and review)				
	Lv2 (Automated tool)				
	Lv3 (Diagnosis by experts)				

Matrix,
to define
‘**Required task**’
from
‘**Required Security Level**’

4. Apply the concept of CC to system integration

- Concept of CC applied to our standard process

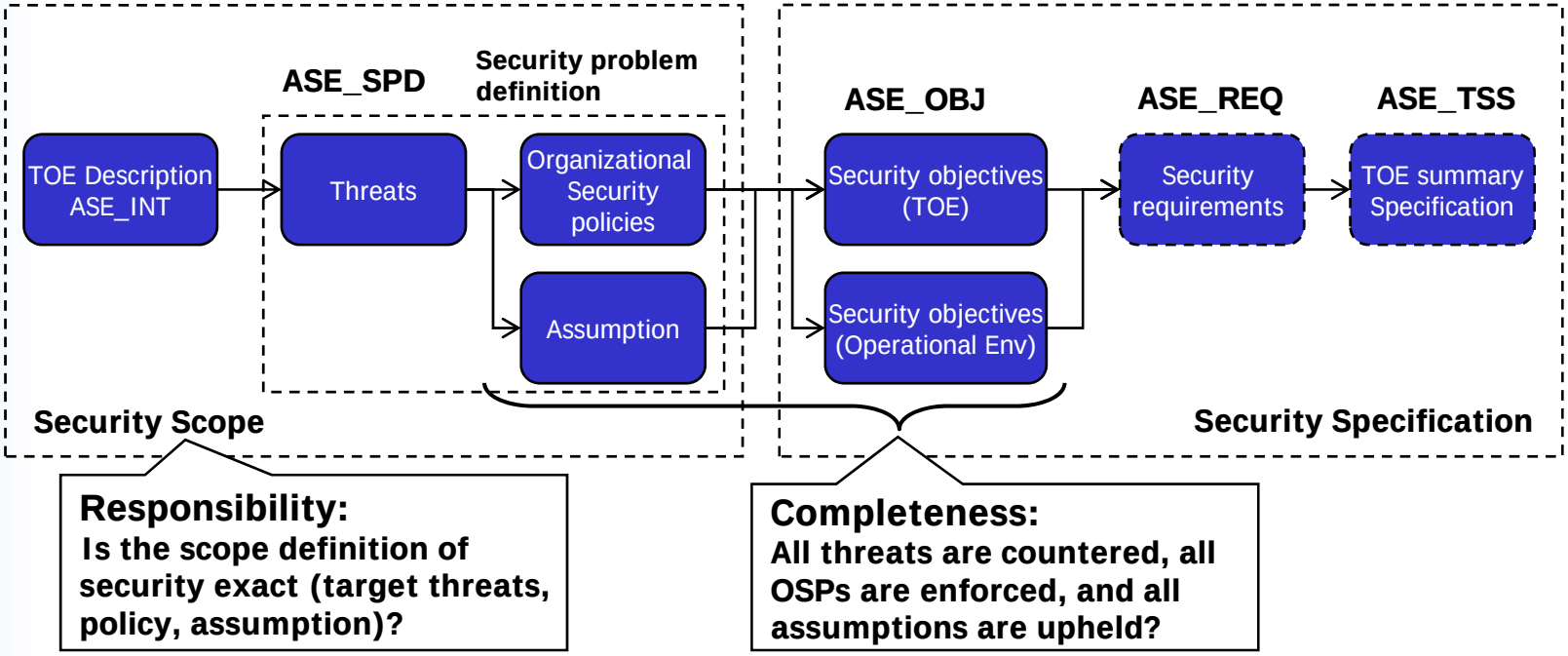


4. Apply the concept of CC to system integration

4.1 Planning Process: Definition of security requirement

– Clarify security scope (considering **Responsibility**), and solve all security concerns (considering **Completeness**) in Planning Process (BI,RD) where we applied ST concept (definition of security scope and specification) to our standard process

Security risk analysis



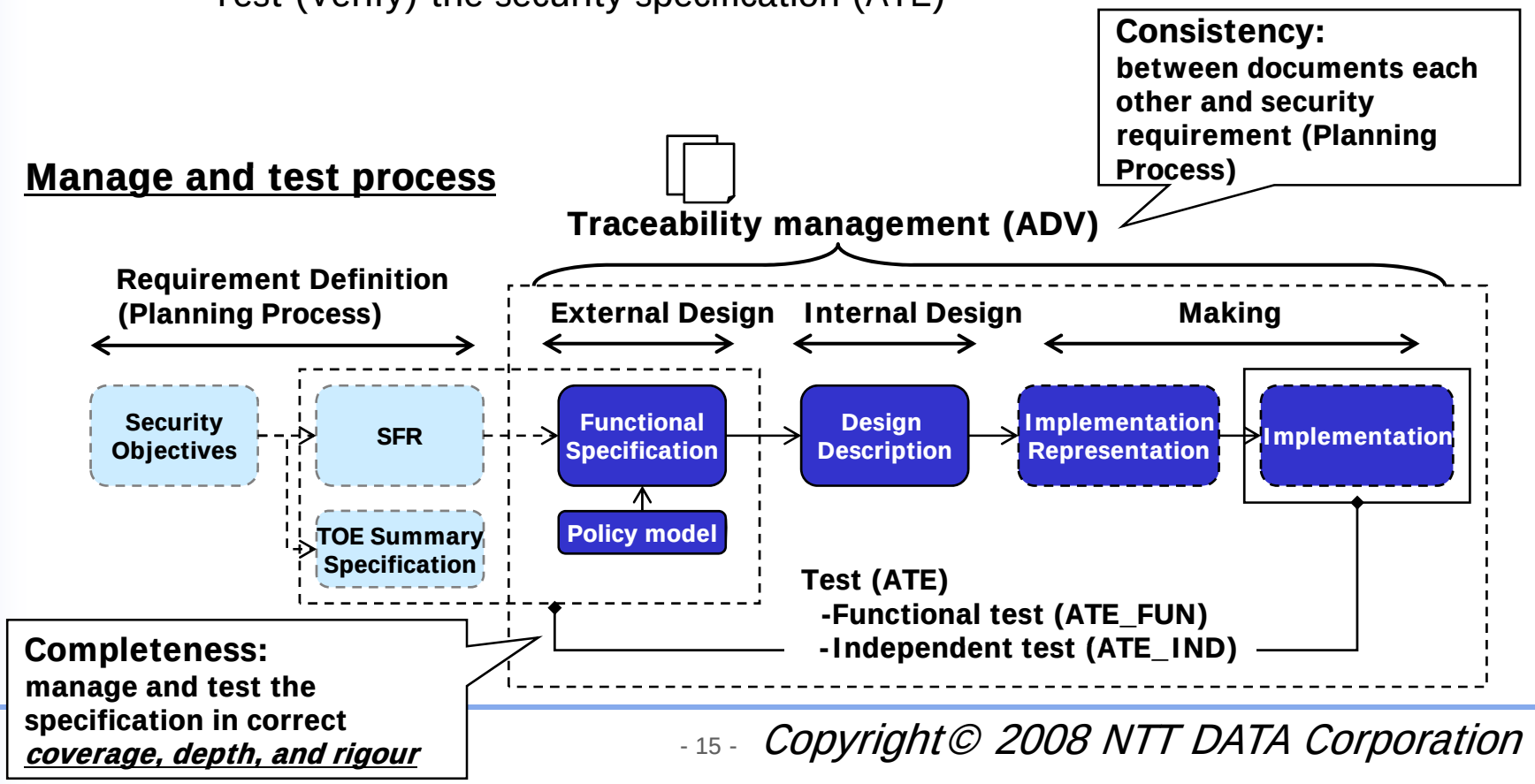
4. Apply the concept of CC to system integration

4.2 Development process: Realization of Security specification

– Manage and test (verify) to realize security specification defined in Planning process (BI,RD) (considering **Completeness, Consistency**)

where we applied CC security assurance concept as follows:

- Manage the security specification with keeping traceability (ADV)
- Test (Verify) the security specification (ATE)

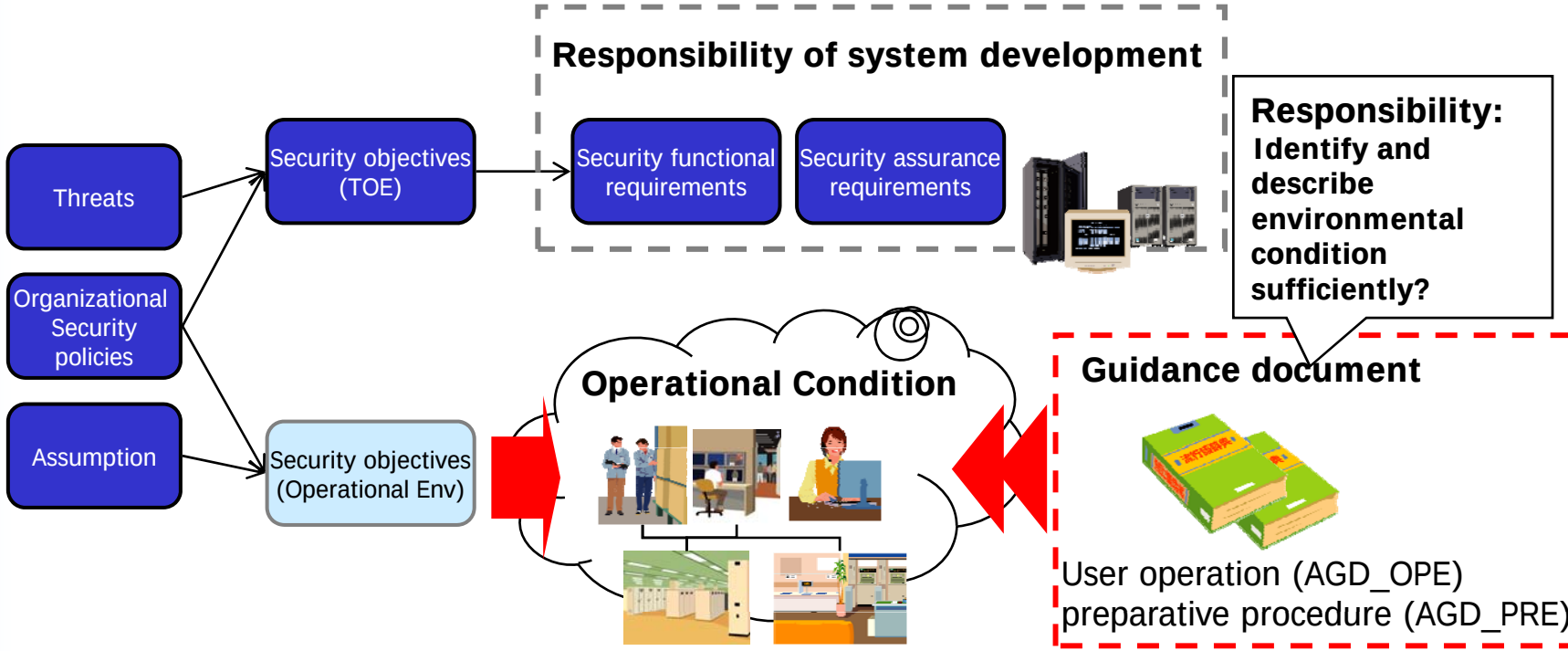


4. Apply the concept of CC to system integration

●4.3 Operation Process: Clarification of operational condition

- System security requirement is satisfied by not only TOE function but "environmental condition"
- To clarify responsibility of system development (=Responsibility), provide "guidance document" that describe environmental condition

  where we applied CC assurance concept (guidance document: AGD class)



5.Conclusion

- Our goal:

  Establish the effective and pragmatic methodology to assure integrate secure system

- Apply CC concept to our system integration standard process
  - *Project security*
  - *System security*
  - *Concept of "Required security level"*
- CC concept:
  - *Completeness*
  - *Consistency*
  - *Responsibility*

6. Further issue

- NFR interference
  - Security may interfere with Performance, Usability, as well as Maintenanceability
  - When we should take into account this problem? How we could resolve or find agreeable Quality
- Cost
  - Hard to estimate necessary cost for security quality (not only buying security product, but also development costs)
  - How we could explain the security cost to be needed in the project
  - Low cost leads less security
- Optimization
  - How to divide the responsibility of security between logical layers, different developers, different players, as well as to keep balance with security, cost and other NFRs
  - Concept of “Composite Evaluation Class” (in CC v3.*) may help us in the case of a large scale IT system development, to resolve the complexity about responsibility of security

Reference:

- System integration process:
  - ISO/IEC 12207:2008 Systems and software engineering -- Software life cycle processes
  - Software Life cycle Processes-Japan Common Frame 2007 SLCP-JCF-2007
  - CMMI for development Version1.2
  - NTTDATA TERASOLUNA® Development process ver3.0
- Framework related to security
  - Common Criteria Ver3.1 part1,2,3
  - ISO/IEC 13335-1:2004 , ISO/IEC TR 1335-5:2001 (GMITS)
  - ISO/IEC 27002:2005 Code of practice for information security management (ISMS)
  - SSE-CMM ver3.0
- Security Design
  - Secure Systems Development with UML , Jan Jurjens
  - Security Patterns, (http://www.securitypatterns.org/patterns.html#2008)
  - Trustworthy Computing Security Development Lifecycle, Microsoft
  - CLASP (Comprehensive Lightweight Application Security Process), Fortify
- Presentation, Paper
  - “ISMS Aspects in Common Criteria Certificates for Development Sites”, Bertolt Krüger,6th ICCC (2005)
  - “The Requirements for IT System Evaluation”, Haruki TABUCHI, 4th ICCC (2003)