

# How we keep on improving smart card security development, following CC requirements

---

22<sup>nd</sup> of September, 2010  
Naohisa ICHIHARA  
R&D Headquarters, NTTDATA Corporation



# 1. Introduction



- NTTDATA's CC related activities;
  - 2001
    - CC study started, studied European PPs for Smartcard (PP9806,PP9911)
  - 2002-2005
    - Key member of Smart MEIJI (WG between Japan and EU), 2003
    - Join and discussed in EU smartcard security community (current ISCI)
    - The 1<sup>st</sup> success of Japanese vendor to get an EAL4+ Certificate for Composite smartcard evaluation under by French DCSSI (currently ANSSI)
  - 2006-2008
    - The 1<sup>st</sup> success of Japanese vendor to get an EAL4+ Certificate with Formal method (ADV\_SPM.3) , for e-passport IC (ICAO)
  - 2009-2010...
    - Still we are improving our smartcard security development skills...

## 2. Issues for smartcard security development



- General concerns for Smartcard security

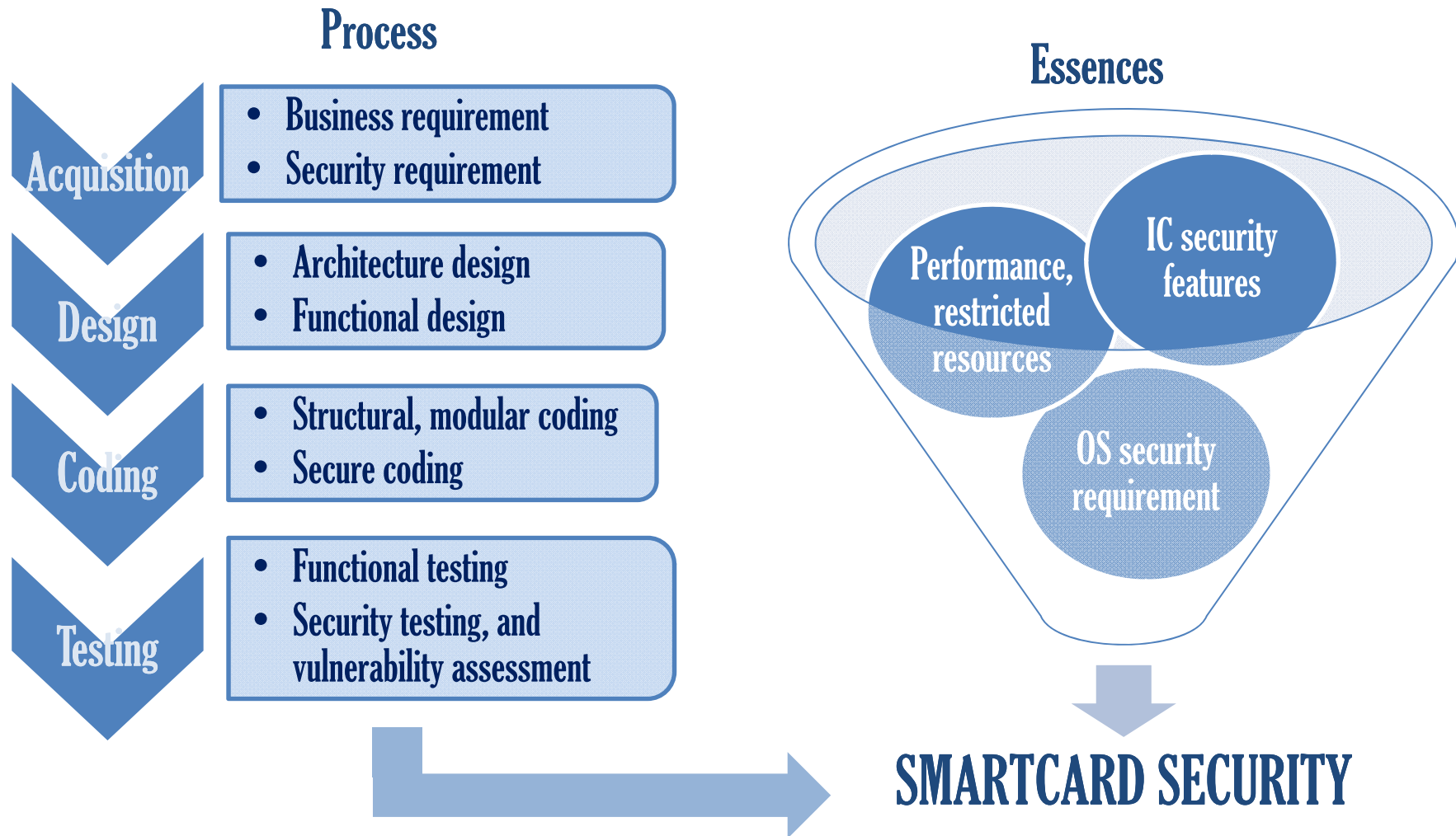
- An **inexpensive**, but required **to be secured**
- Easily exposed to an attacker's environment, which leads **new vulnerability found** day by day.
- In general, **not easy to update security** (by patch loading) after distributed to end-user



- Issues for Smartcard developers

- Develop a secure device with **restricted resources** among a restricted period
- Secure implementation may **lose its performance, occupy much memory**
- Smartcard security never be completed until Embedded Software (OS/AP) is practically coded on IC (not on emulator nor simulator).
- Utilize an accumulated **know-how, frameworks, and tools**

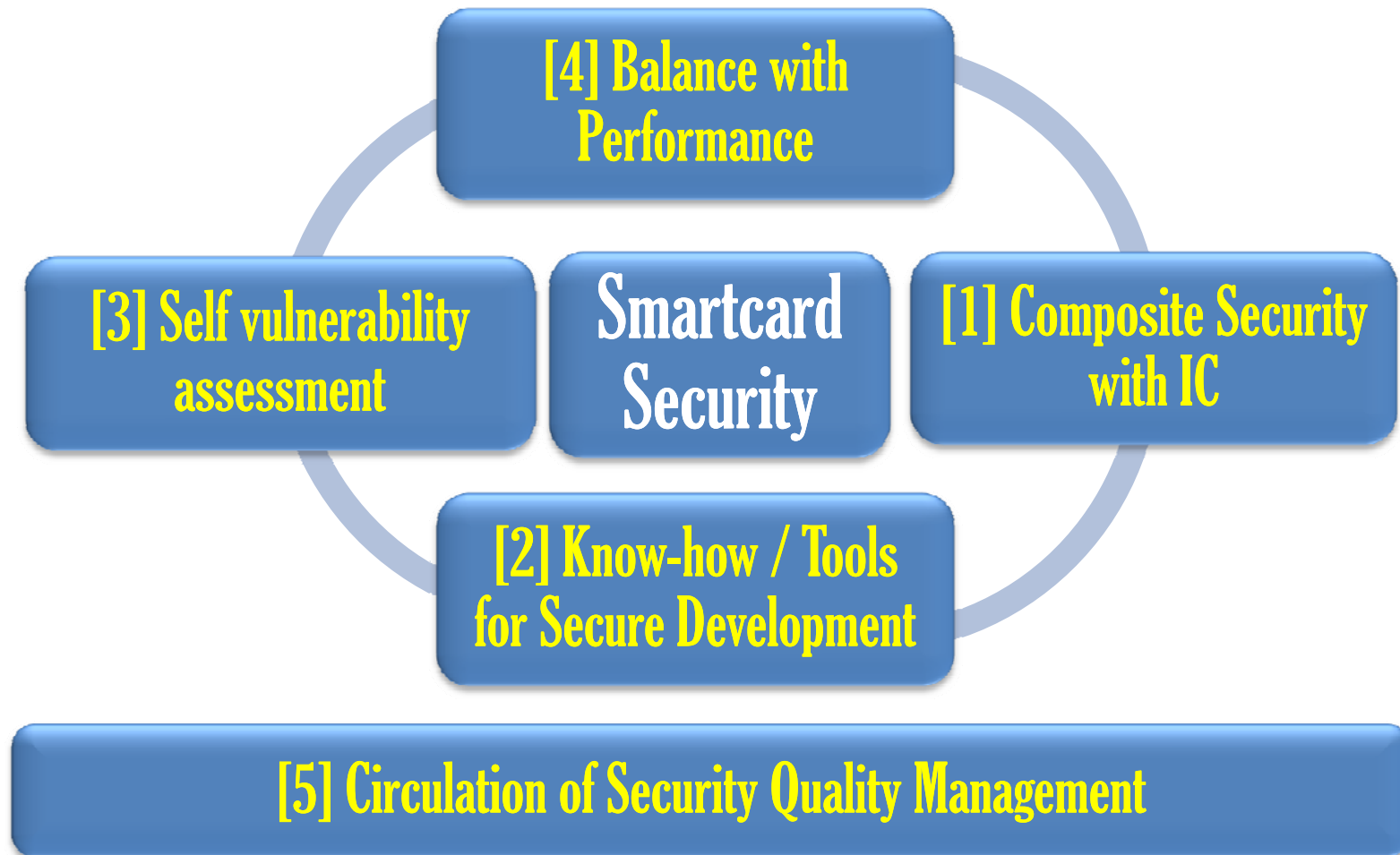
## 2. Issues for smartcard security development



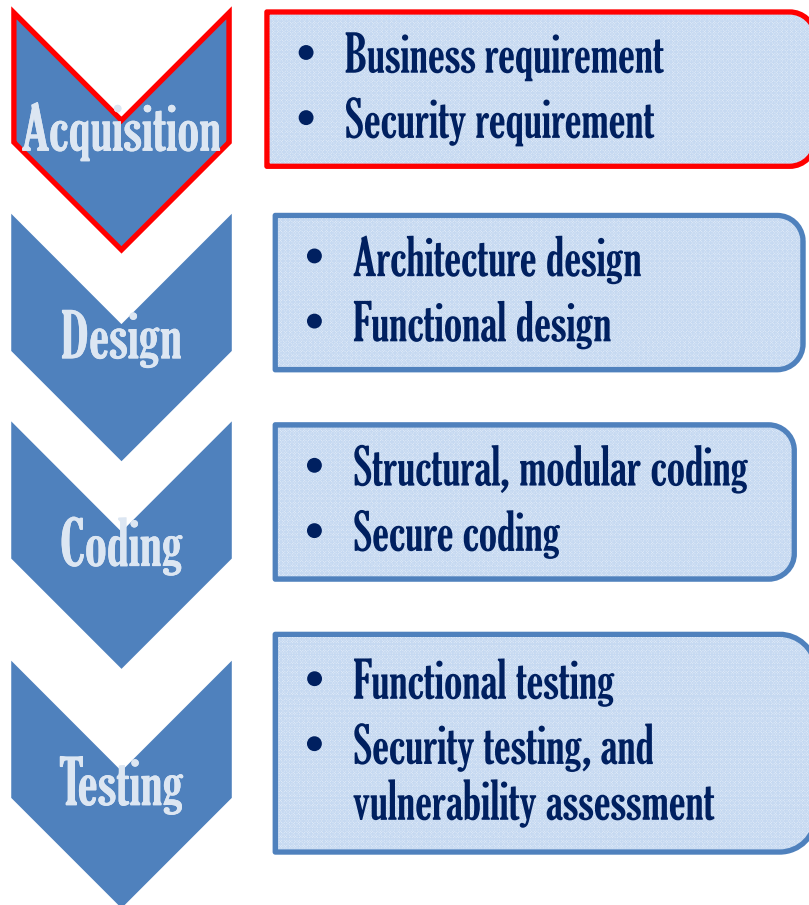
## 2. Issues for smartcard security development



- NTTDATA's approaches to keep on improving



### 3. NTTDATA's secure smartcard development



#### [1] Composite Security with IC

[1-1] Following of IC Security Target & Security Guidance

[1-2] Composite Security Design  
(to follow IC security recommendations)

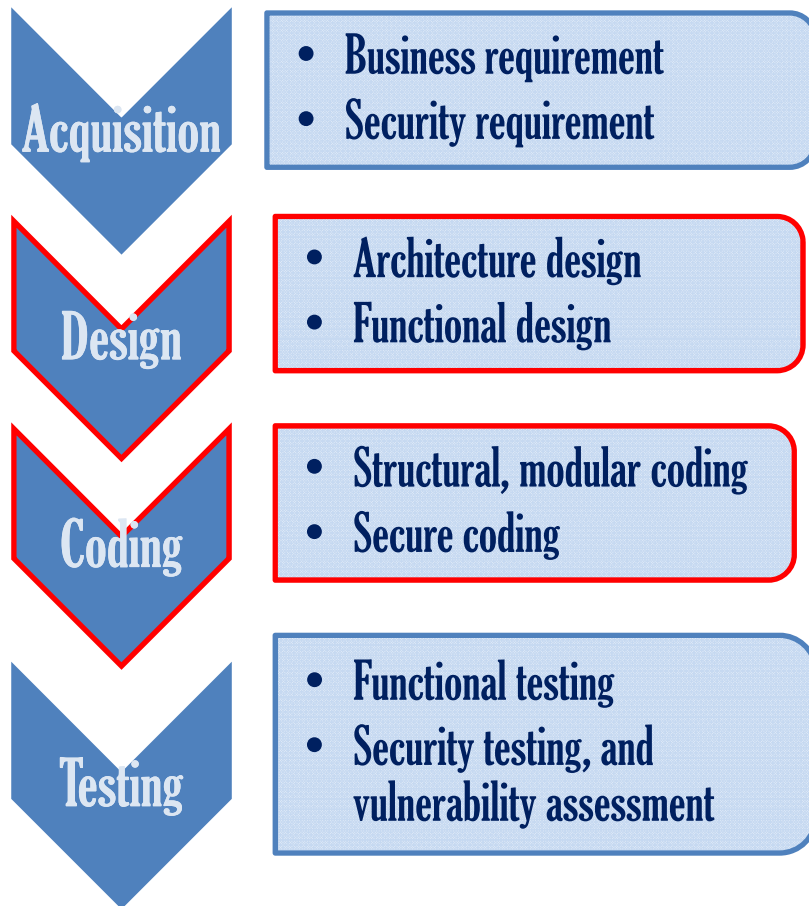
[1-3] Specific Coding  
(to follow IC security recommendations)

#### **POINT#1**

Understanding of IC security, **not only WHAT but also WHY.**

Understanding the difference between previous IC series security and new ones.

### 3. NTTDATA's secure smartcard development



#### [2] Know-how / Tools for Secure Development

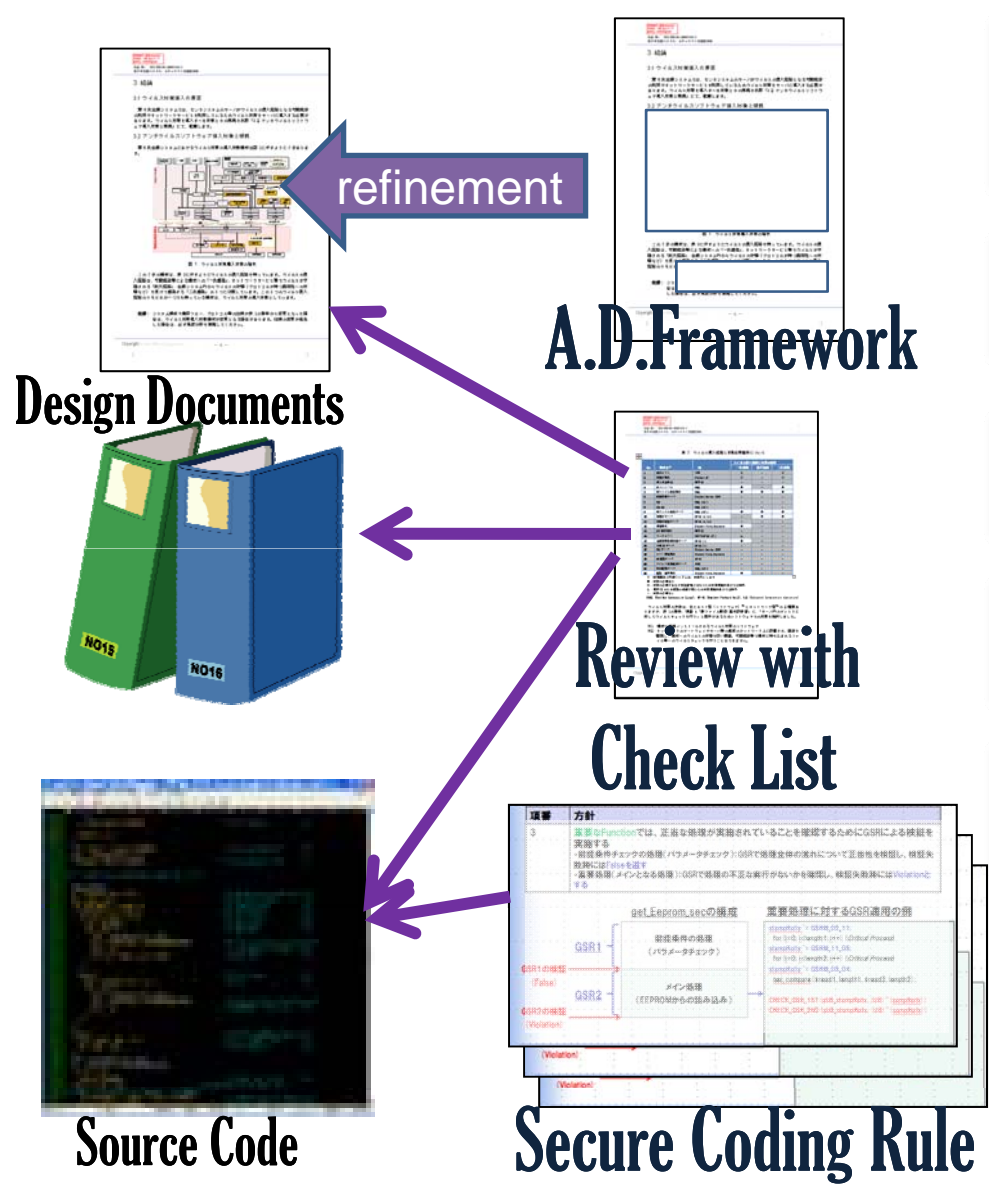
[2-1] Security Target / PP  
(for Risk Analysis, Security Requirement Acquisition)

[2-2] Architectural Design Framework  
(Reuse of previous project know-how)

[2-3] Fault check of Security Design  
(By Review with Check list, or Formal Method Tool)

[2-4] Secure Coding Rule & Check List  
(Reuse of previous project know-how)

# 3. NTTDATA's secure smartcard development



**[2] Know-how / Tools for Secure Development**

**[2-1] Security Target / PP**  
(for Risk Analysis, Security Requirement Acquisition)

**[2-2] Architectural Design Framework**  
(Reuse of previous project know-how)

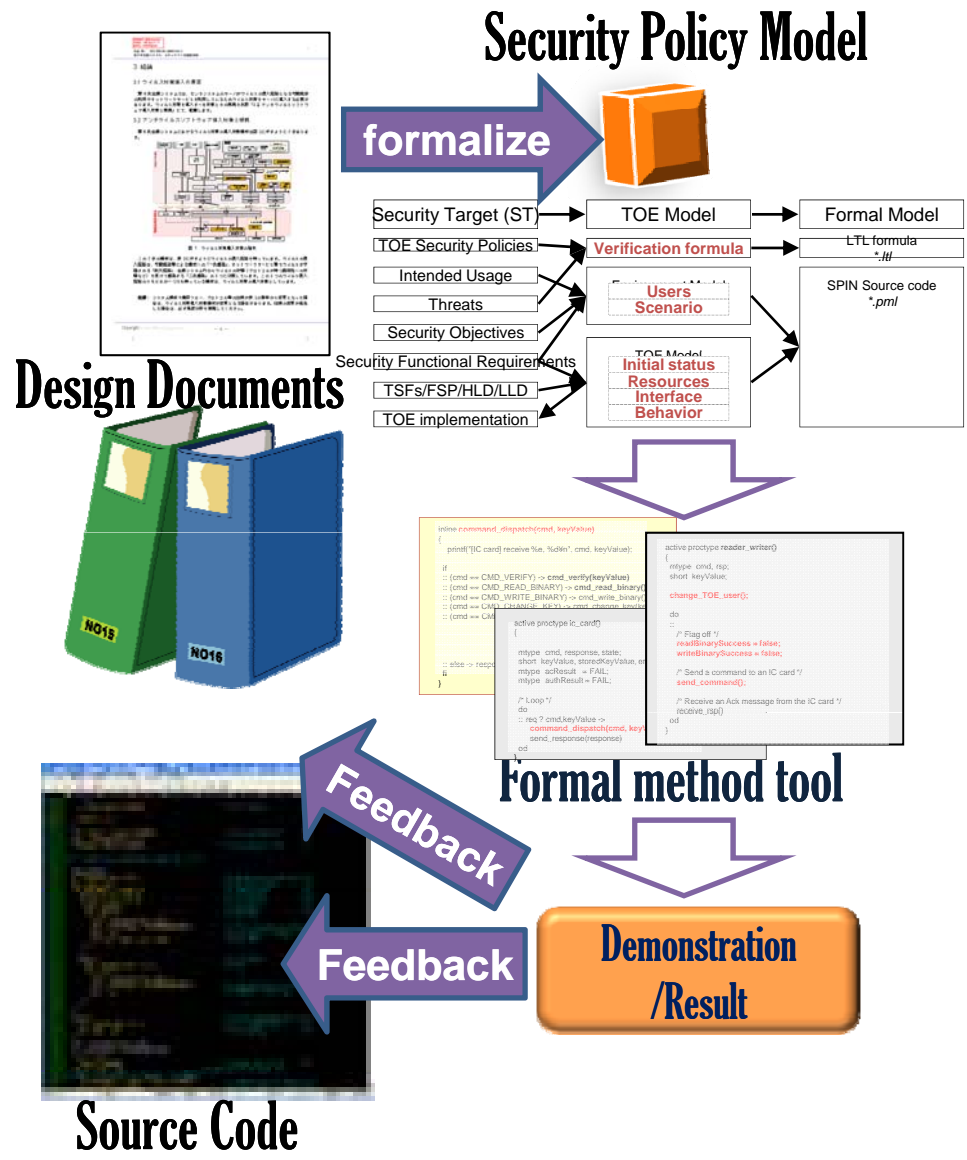
**[2-3] Fault check of Security Design**  
(By Review with Check list, or Formal Method Tool)

**[2-4] Secure Coding Rule & Check List**  
(Reuse of previous project know-how)

**POINT#2**

- Experience feeds back as FW/CL/Rules
- Not only reuse of "know-how", but also upgrading "know-how" is important

# 3. NTTDATA's secure smartcard development



**[2] Know-how / Tools for Secure Development**

**[2-1] Security Target / PP**  
(for Risk Analysis, Security Requirement Acquisition)

**[2-2] Architectural Design Framework**  
(Reuse of previous project know-how)

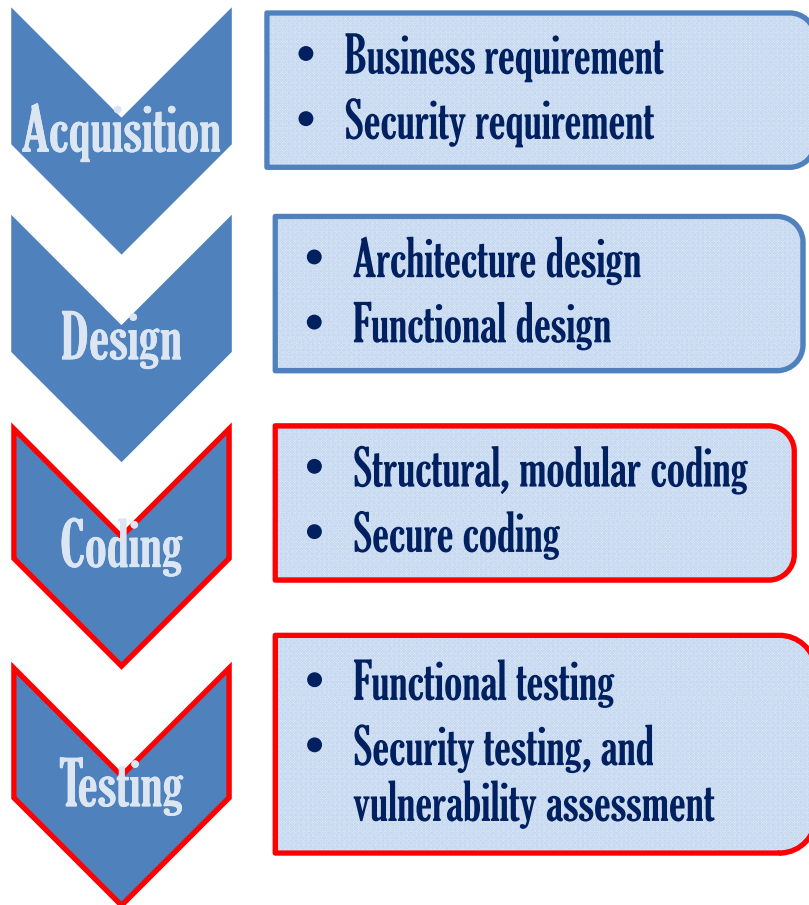
**[2-3] Fault check of Security Design**  
(By Review with Check list, or Formal Method Tool)

**[2-4] Secure Coding Rule & Check List**  
(Reuse of previous project know-how)

### 3. NTTDATA's secure smartcard development



- [3] Self vulnerability assessment



#### [3] Self vulnerability assessment

[3-1] Security Testing  
(for Expected Security Functions addressing SFRs)

[3-2] Vulnerability assessment  
(for Unexpected TSF behaviors by attacks)

### 3. NTTDATA's secure smartcard development

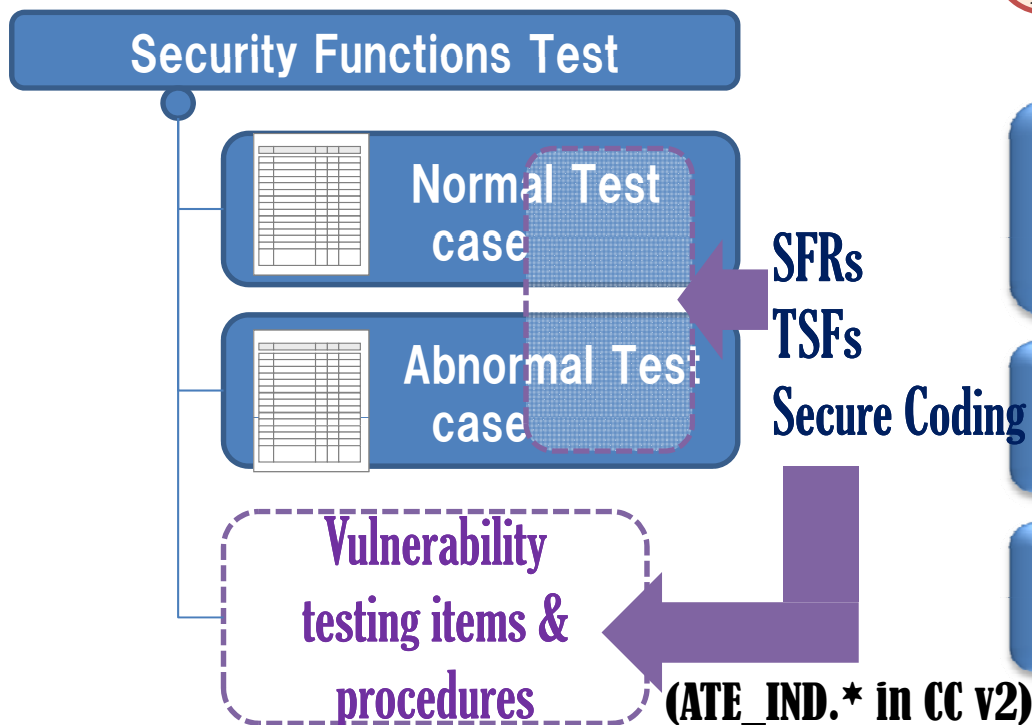


- [3] Self vulnerability assessment



#### **POINT#3**

Important to **prioritize** and pick up most critical attack paths among all your exhaustive testing patterns



#### [3] Self vulnerability assessment

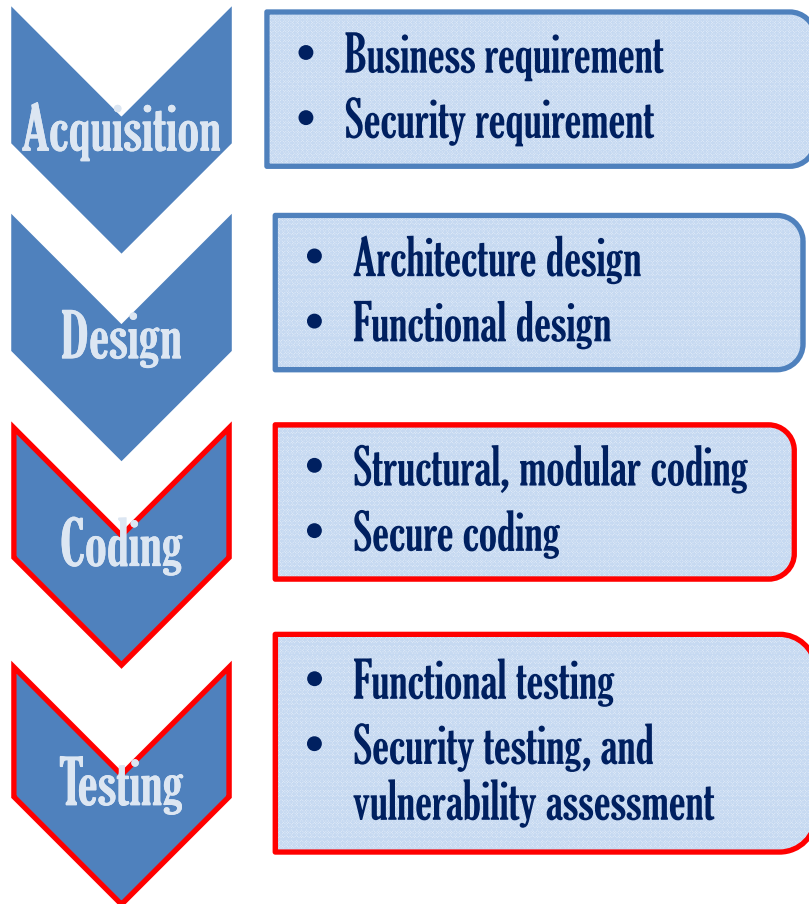
[3-1] Security Testing  
(for Expected Security Functions addressing SFRs)

[3-2] Vulnerability assessment  
(for Unexpected TSF behaviors by attacks)

### 3. NTTDATA's secure smartcard development



- [4] Balance with Performance



#### **POINT#4**

Performance and code size estimation during secure implementation could be useful

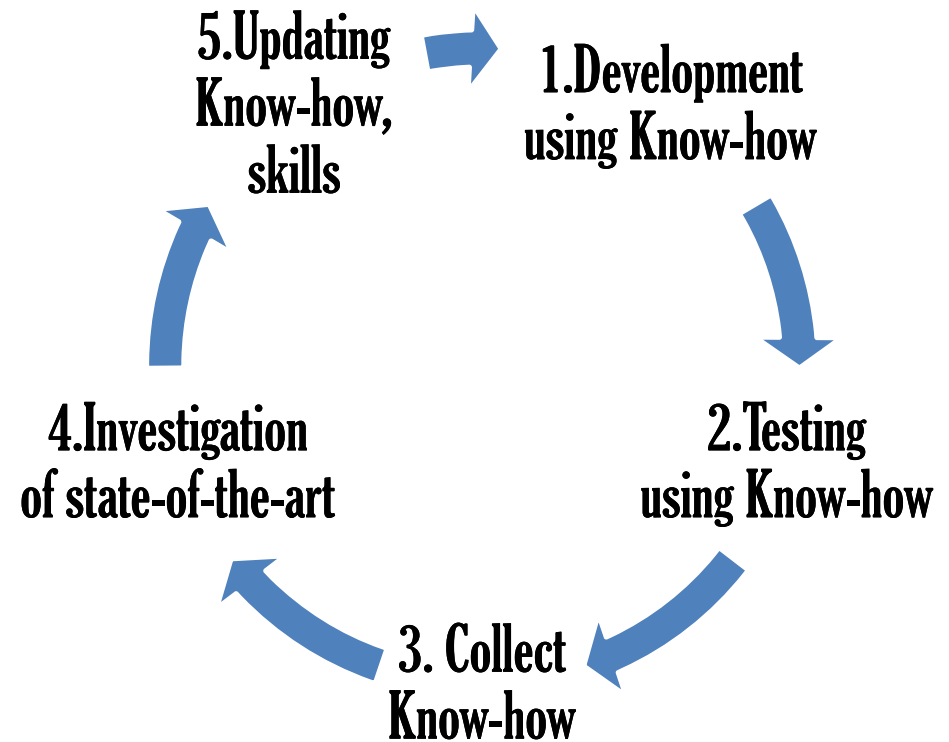
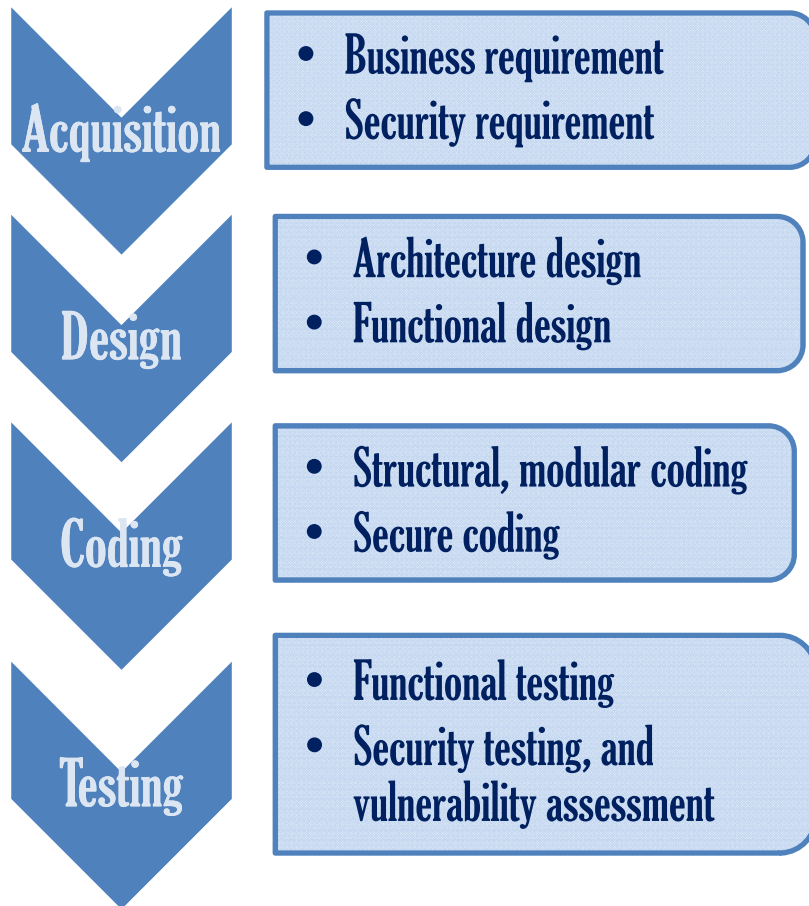
**[4] Balance with Performance**

**[4-1] Performance and Occupied Memory Estimation**

### 3. NTTDATA's secure smartcard development



- How we keep on improving ?



**[5] Circulation of Security Quality Management**

### 3. NTTDATA's secure smartcard development

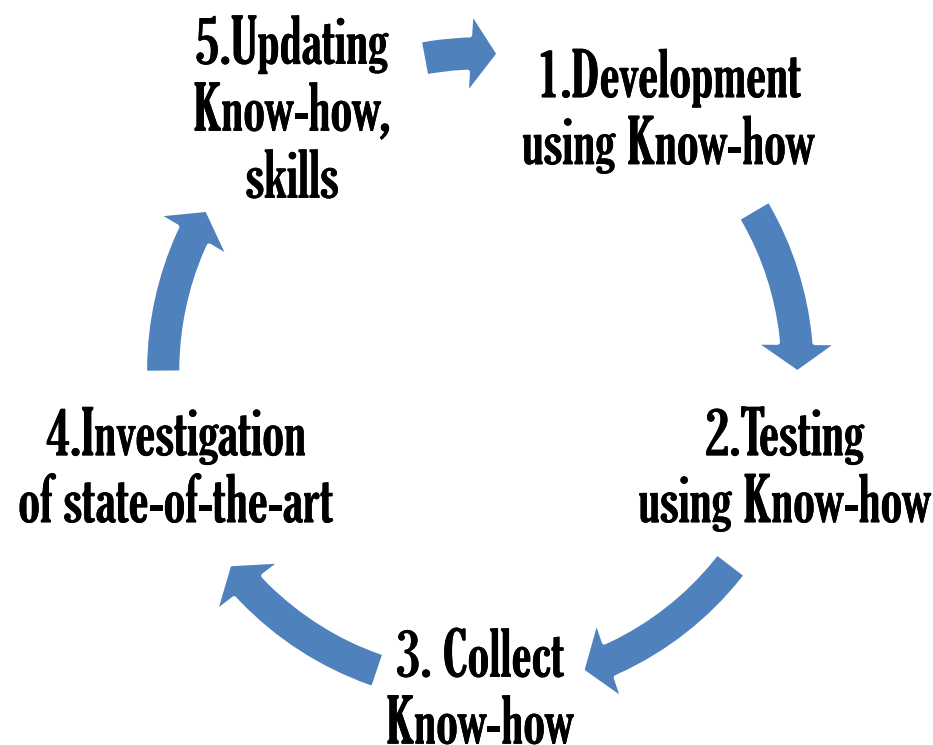


- How we keep on improving ?

#### **POINT#5**

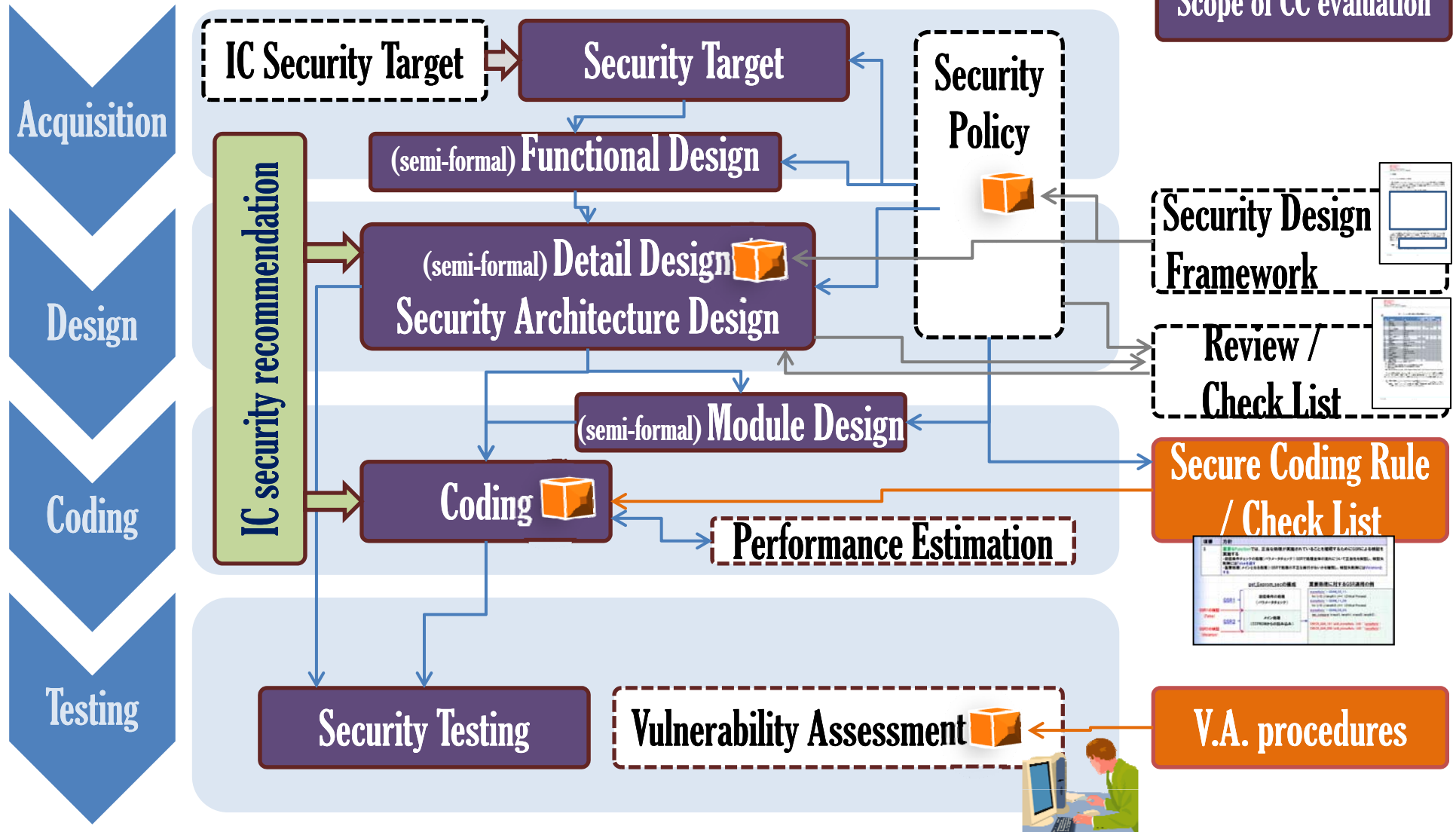
Additional tasks needed to keep security quality of product, besides the scope of CC evaluation. e.g. Security Policy Definition, Vulnerability assessment, (in some cases) Formal methods, ..

It is **MUST** to keep on improving developer's skills for secure development as well as (and based on) CC concepts.

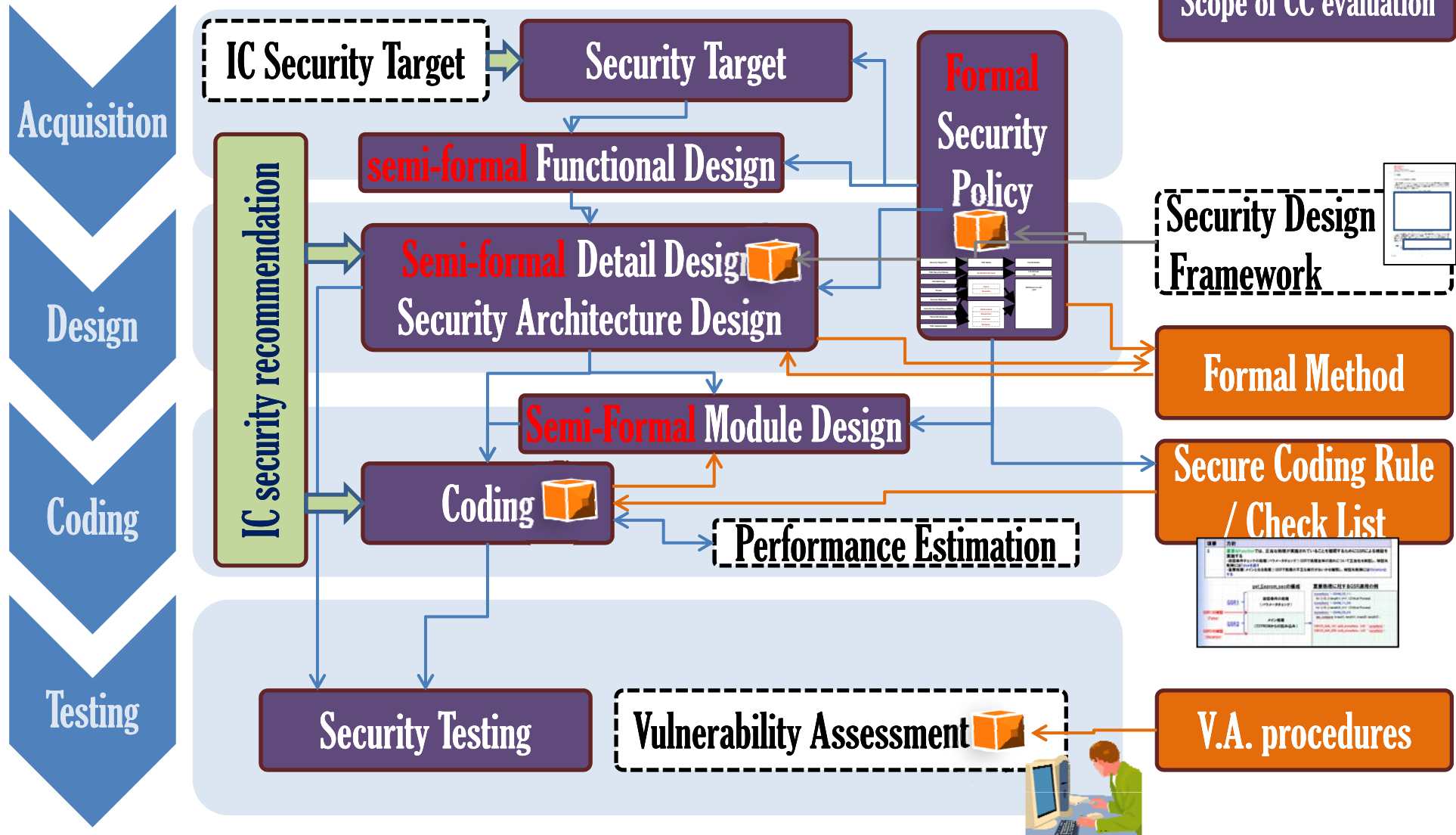


**[5] Circulation of Security Quality Management**

# 4. Case study (EAL4+ ~ EAL5)



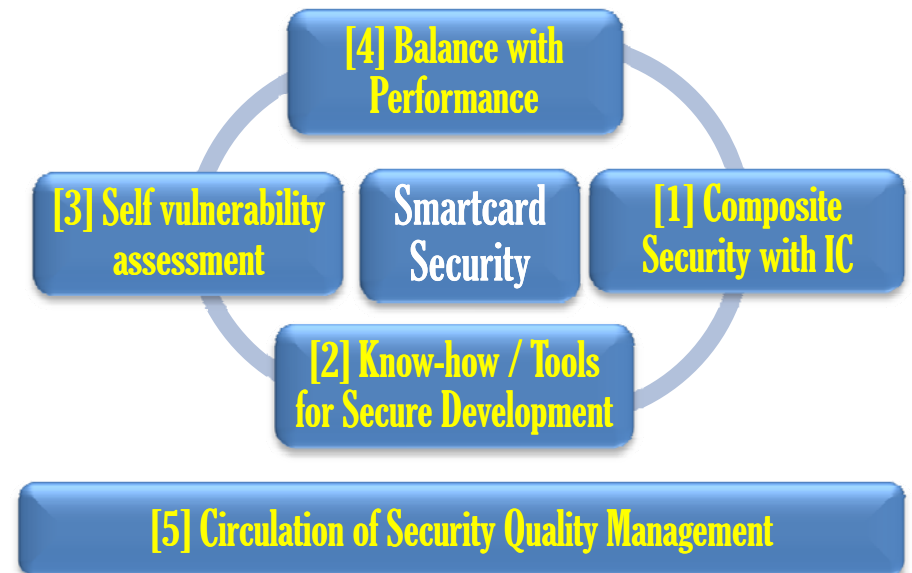
# 4. Case study (EAL6)



## 5. Summary



- [1] Composite Security evaluation with IC
  - Understanding of IC security, **not only WHAT but also WHY**.
  - Understanding the **difference** between previous IC series security and new ones.
- [2] Know-how / Tools for Secure Development
  - Experience feeds back as FW/CL/Rules
  - Not only reuse of “know-how”, but also **upgrading “know-how”** is important
- [3] Self vulnerability assessment
  - **Prioritizing** invisible attack path for testing
- [4] Balance with Performance
  - **Performance and code size estimation** during secure implementation could be useful
- [5] Circulation of Security Quality Management
  - **Additional tasks** needed to keep security quality of product, **besides the scope of CC** evaluation. e.g. Security Policy Definition, Vulnerability assessment, (in some cases) Formal methods, ..
  - It is **MUST** to keep on improving developer’s **skills for secure development** as well as (and based on) **CC concepts**.



# Questions?

e-mail: [ichiharan@nttdata.co.jp](mailto:ichiharan@nttdata.co.jp)

変える力を、ともに生み出す。

NTT DATAグループ

