STAMP Based Safety Analysis for Navigation Software Development Management

Xu Xiaojie Zhong Deming

Ph. D Candidate for Software Safety

Email: buaaxuxiaojie@163.com

1. Our goal

- Discuss the safety analysis method based on STAMP that:
  - Provides an organized and effective means for safety analysis at software develop management level
  - Develops appropriate countermeasures for improving software develop management structure and therefor software safety

2. Background—The Problem(1)

- After a modification for an aircraft, data for navigation system were frequently lost
- It became a serious threat to flight safety, as there were 20 such failures in 24 flights
- The mishap was reproduced on ground through simulation and then analyses were carried out by conducting SFTA combined with SFMEA

2. Background—The Problem(2)

- Finally cause for lost data was identified as "synchronization between main CPU and protocol chip", located at NS (navigation software).
- After the work, we still want to explore the underlying cause of the unsafe NS, and this could be done at develop management level for NS.

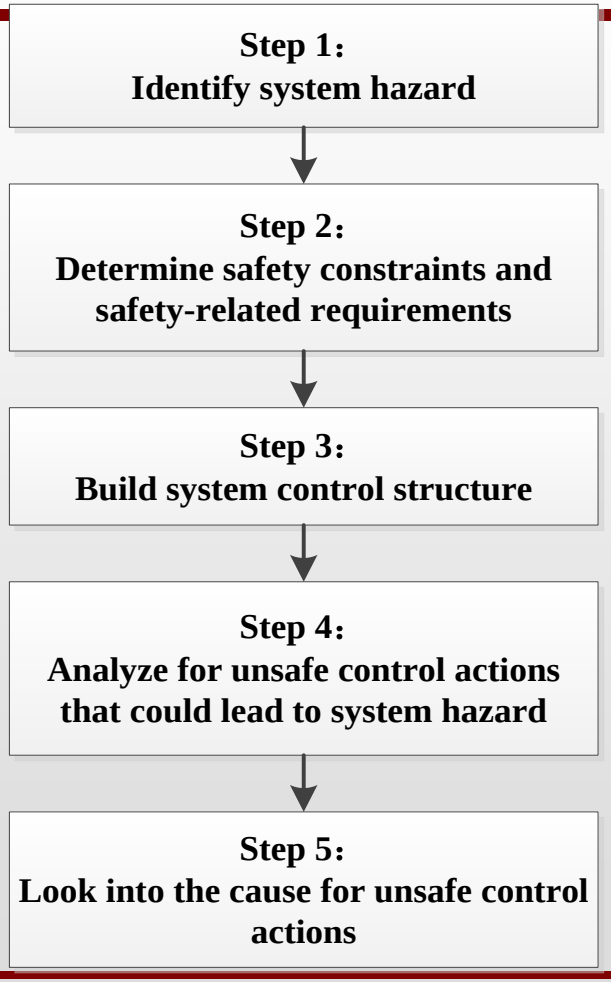
3. Why STAMP?

- **Methods used at technique level might be not effective when used to handling problems about software develop management.**
- **Finally *STAMP* based method was adopted !**
  - **STAMP is applicable to sociotechnical systems due to the theory it based.**
  - **NS develop management system could be delivered in the form of control structure, and this is the basis for STAMP based analysis.**

4. Steps for Analysis

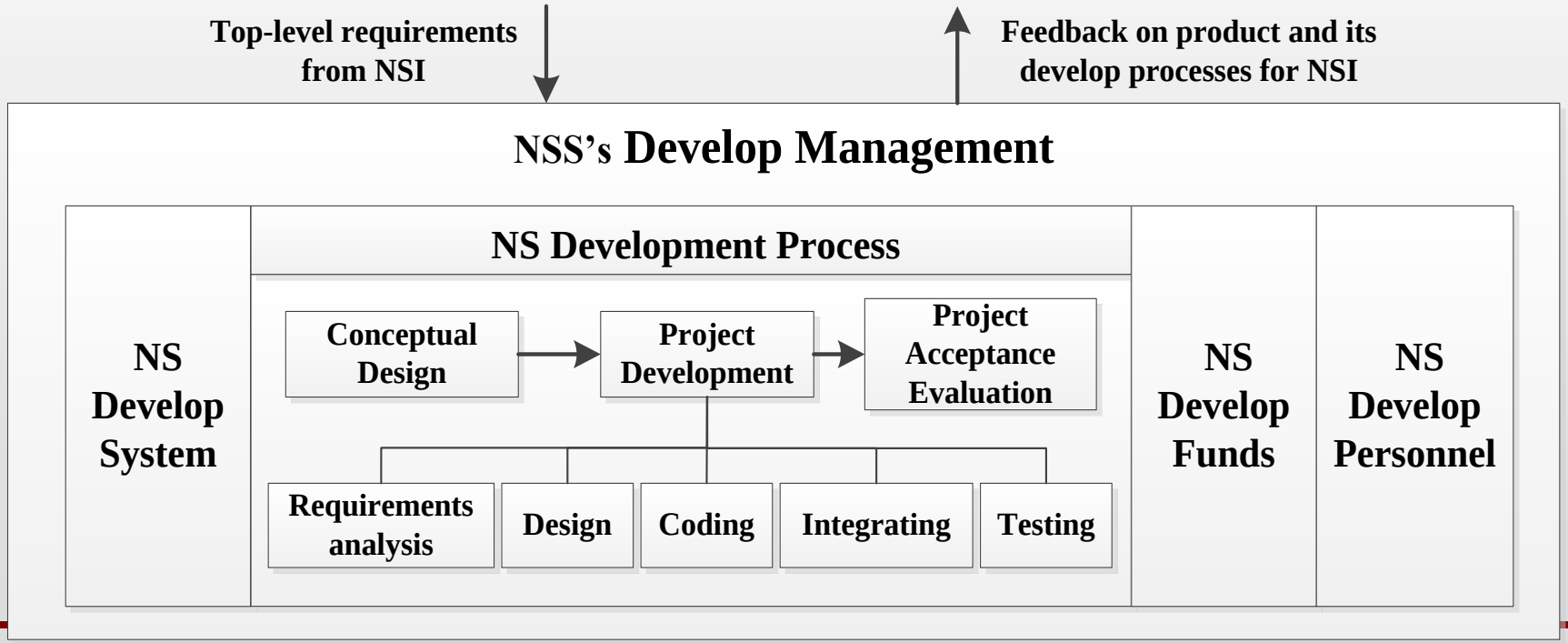
Steps for STAMP based safety analysis

- Countermeasures could be presented after step 5 based on for identified causes for the unsafe control actions .
- As a result system safety could be guaranteed.



5. Current Status of NS Develop Management

- there are two "components" in NS develop management system : NSS(navigation software supplier) and NSI(navigation system integrator).



5. Current Status of NS Develop Management

- ***1) Software develop system***: preliminary software develop system has been constructed, requirements and paradigm for NS were established**.**
- ***2) Software develop processes*:** Project develop process consists of 3 steps: determination for top-level requirements, project development and acceptance validation for the software. While project development in this case composes of requirements analysis, design, coding, integration and testing.
- ***3) Software develop funds:*** Funds for software development could be partly guaranteed.

5. Current Status of NS Develop Management

- ***4) Personnel with the ability of software engineering:*** Suppliers of NSS have professional staff with the ability of software engineering to undertaking software development and management.
- ***5) Top level requirements issued by NSI:*** Top level requirements were issued to NSS and played an important role in development process control and quality assurance for NS.
- ***6) Supervision and verification on NSS from NSI:*** Supervision and verification are realized by the navigation software development at key phase.

6. Analysis on NS develop management(1)

- **In this case, navigation system develop management structure has two roles:**
  - **Navigation system integrator (NSI)**

    NSI' responsibilities :making and issuing top-level requirements for NS.
  - **Navigation software supplier (NSS)**

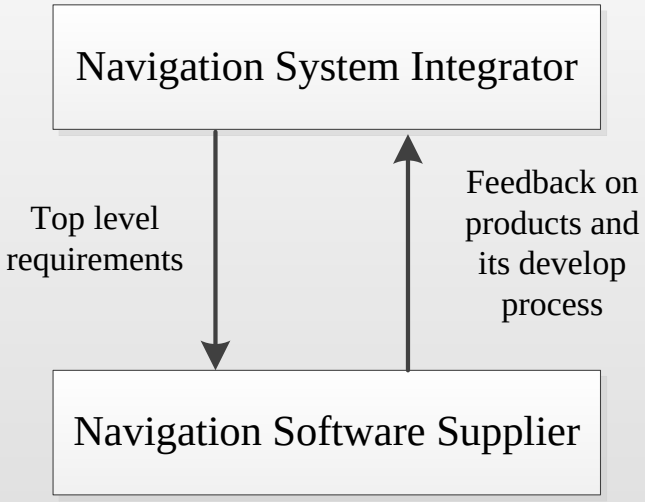
    responsibilities: develop NS according to requirements from NSI.

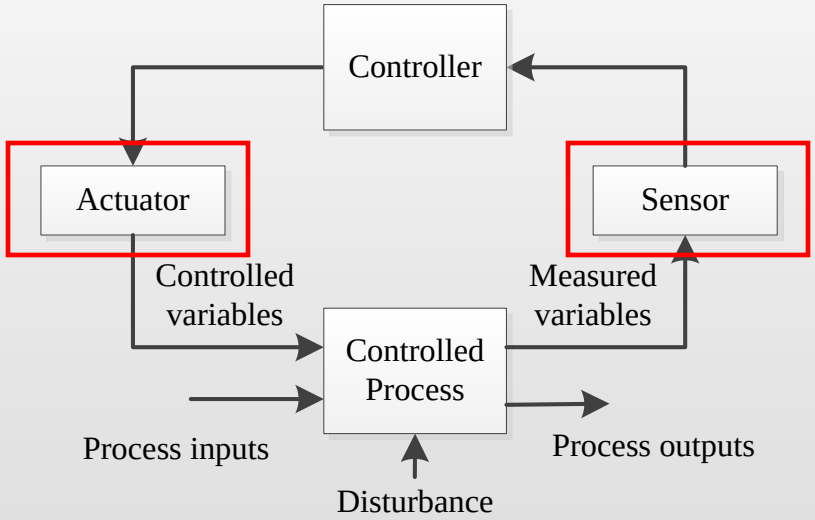
6. Analysis on NS develop management(2)

- **1st step: hazard identified for NS develop management:**
  - "Navigation software with feta defects is integrated into navigation system"
- **2nd step: Safety constraint for the hazard:**
  - "NSS should develop NS following the top-level requirements from NSI".
  - "NSI should ensure NS accept from NSS is conform to the its requirements"

6. Analysis on NS develop management(3)

✈ 3rd step: control structure





6. Analysis on NS develop management(4)

4th step: Analysis for unsafe control action

- **Three control actions are analyzed :**
  - "NSI issues top-level requirements to NSS"
  - "Supervise and inspect on NSI's capacity of development and management
  - "Provide NSI with the information of software development"
- **Four types of unsafe control actions**
  - Control action required is not provided or not followed
  - Unsafe or wrong control action is
  - Safety control is provided too late, early, or out of sequence
  - Safe control action is stopped too soon or too late.

Results of unsafe action analysis

NO.	CONTROL ACTION	NOT PROVIDING CAUSE HAZARD	PROVIDING CAUSE HAZARD	PROVIDED TOO LATE	STOPPED TOO SOON
1	NSI issues top-level requirements to NSS	1) Top-level requirements are not provided; 2) Top-level requirements are not followed by NSS.	1) Top-level requirements provided are not sufficient and clear; 2) Top-level requirements provided are Wrong or unsafe.	Top-level requirements are provided too late.	N/A
2	Supervise and inspect on NSI's capacity of development and management.	Supervision or inspection is not carried out by NSI.	Supervision or inspection is carried out but not effectively deployed.	Supervision or inspection is carried out late after the needed time	Supervision or inspection is stopped before it is effectively works.
3	Provide NSI with the information of software development	Feedback is not provided for NSI	Wrong or incomplete feedback is provided to NSI	Feedback is provided too late after the required node time.	N/A

6. Analysis on NS develop management(5)

Causes for mishaps

- *1)* **Top-level requirements issued by NSI are mainly focused on software testing, while requirements for other software development activities are not well covered.**
- *2)* **Supervision and inspection for NS develop process are not required explicitly in top-level requirements, which makes problems like low level of requirements and design capability or management problems are not detected in time.**

6. Analysis on NS develop management(6)

Causes for mishaps

- 3) Some of the top-level requirements are too vague to implement, review and trace.
- 4) Top-level requirements are implemented without any supervision or inspection
- 5) Some check or monitor from NSI is too weak even missed.
- 6) Existing check or monitor from NSI is not effective as expect. Problems found were not well improved though NSS passed the exam.

6. Analysis on NS develop management(7)

Recommended countermeasures

- 1) Top-level requirements for navigation software development should be issued to NSS timely.
- 2) Completeness for top-level requirements should be guaranteed, and there should be an explicit standard for the content of the top-level requirements.
- 3) NSS's software develop capability should be examined by NSI regularly.

6. Analysis on NS develop management(8)

✈ Recommended countermeasures

- 4) Not only the final product but also results at each phase should be supervised and inspected.
- 5) NSS should promote its own development level, including improving develop process, its personnel and other aspects.

6. Analysis on NS develop management(9)

Results

- Countermeasures we established are adopted by NSS, and later results demonstrate the effectiveness for improving NS develop management and therefor safety of navigation software.
- The experience got from NS development management was applied to another important software develop management system. Many similar management problems were found after the investigation which also demonstrate our work is practical.

7. Conclusion

- In this case, though we could not get more detailed information of NS for security's sake, we still successfully established some underlying causes and countermeasures for the unsafe NS at management level using STAMP based method.
- Results of this case and the later work show that STAMP based method works well in handling safety analysis at management level.
- More analysis could be conducted to similar management system to improve product safety in the future.

Thanks

Please contact me:

buaaxuxiaojie@163.com