

Using Metamodeling for Software Engineering: A Best-Practice with ADOxx

3 Jan 2020

Prof. Moon Kun Lee

**Chonbuk National Univ.
Republic of Korea**

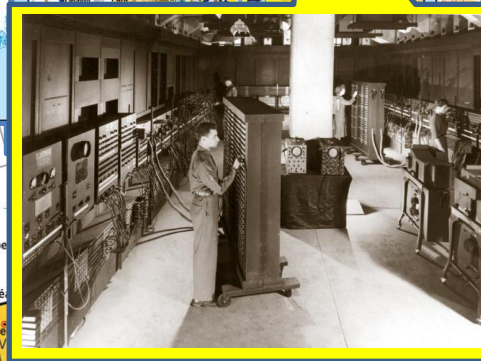
Topics

- A. Introduction: Motivation [10 m]
- B. ADOxx: A Meta-Modeling Platform [30 m]
- C. SAVE: A Formal Method Tool [30 m]
- D. BEE-UP: A Modeling Tool Suite [10 m]
- E. OMiLAB: Open Model Initiative Lab [10 m]

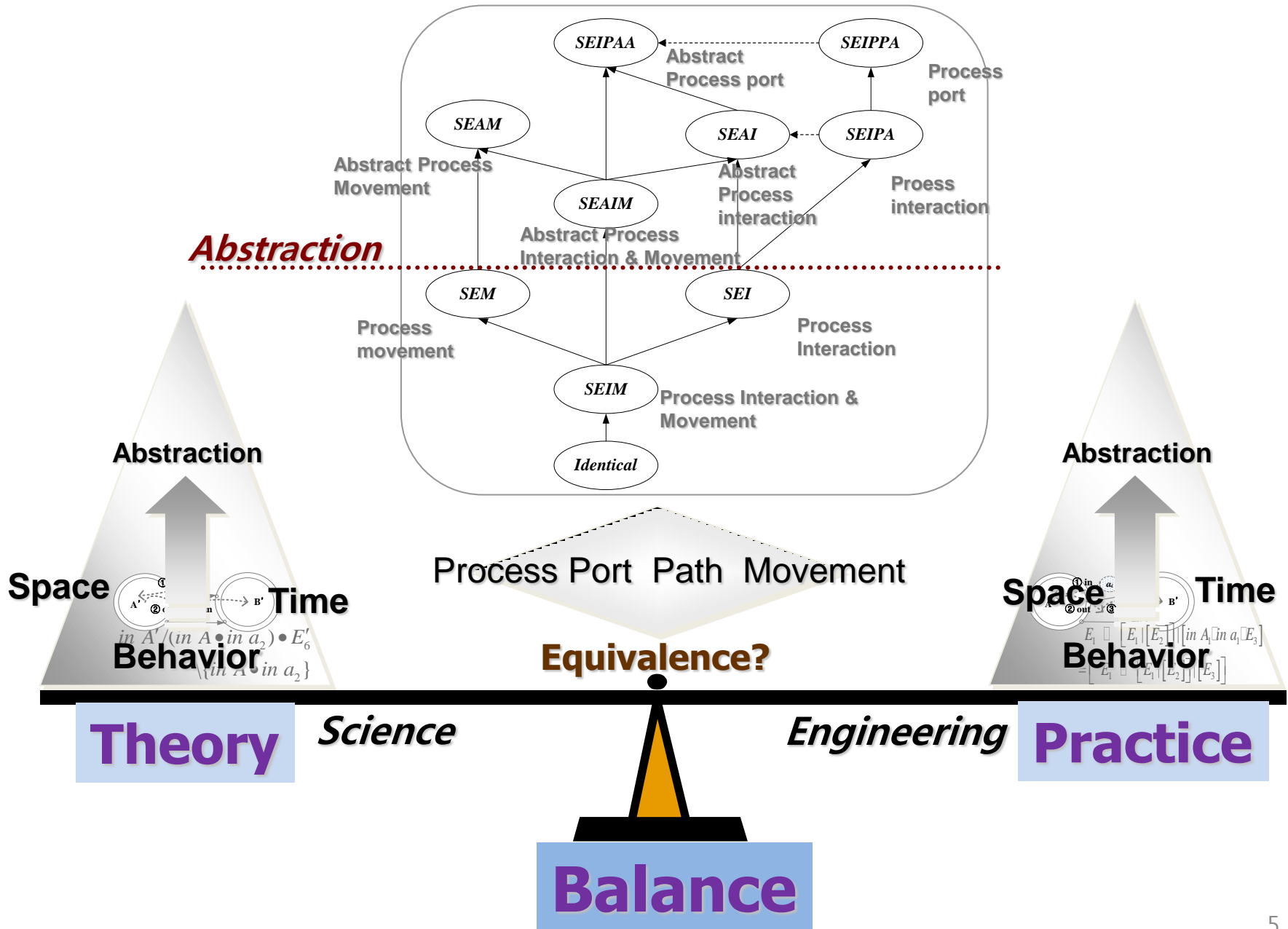
Me: Prof. Moonkun Lee

- Born and Raised in Korea.
- Academic Background
 - Bachelor in Computer Science, Pennsylvania State University, USA.
 - Master in Computer & Information Science, The University of Pennsylvania, USA.
 - *Analysis of Parallelism for MODEL Equational Language*, Advised by Prof. Noah Prywes.
 - Ph.D. in Computer & Information Science, The University of Pennsylvania, USA.
 - *An Environment for Understanding Real-time Software*, Advised by Prof. Noah Prywes and Prof. Insup Lee.
- R&D, CCCC, USA (~1996)
 - SRE(SW Re/Reverse-engineering Environment) Tool, DECDesign, 5 years (11 yrs x 5 men)
 - USA, Navy, 100,000 ~ 1,000,000 LOC (Scalability)
 - OS: ATES, SDEX-20 → Unix, VMS
 - PL: Fortran, C, Ada83 → C, C++, Ada83, Ada95
- Chonbuk National University, Korea (1996~present)
 - Computer Science & Engineering Department
- Research Background
 - Software Re/reverse-Engineering: Round-Trip Engineering
 - Distributed Real-Time Systems
 - Formal Methods: ATM, CARDMI, Onion
 - Behavior Modeling: n:2-Lattice
 - Software Round-Trip Engineering (SRE)
 - Specification: δ -Calculus (Process Algebra)
 - Verification: GTS Logic (1st Order)
 - SAVE/ADOxx Tool
 - ISO Certification and Domain Knowledge Engineering w/ ADONIS

State Map



Software Engineering



Introduction: Motivation

3 Jan 2020

Prof. Moon Kun Lee

**Chonbuk National Univ.
Republic of Korea**

Contents

1. Motivation

- 1) Safety Requirements
- 2) SW Engineering Requirements

2. Modeling w/ Formal Methods

- 1) Specification: dTP-Calculus
- 2) Verification: GTS Logic
- 3) Smart City: EMS Example

3. Probability Choice to Control Nondeterminism

- 1) Motivation
- 2) Definition
- 3) Smart City Example: SEES Example
- 4) Open Issues for Research

4. SAVE Tool

5. Cyber-Physical Systems Application

6. Summary

1. Motivation

- 1) Safety Requirements
- 2) SW Engineering Requirements
2. Modeling w/ Formal Methods
 - 1) Specification: dTP-Calculus
 - 2) Verification: GTS Logic
 - 3) Smart City: EMS Example
3. Probability Choice to Control Nondeterminism
 - 1) Motivation
 - 2) Definition
 - 3) Smart City Example: SEES Example
 - 4) Open Issues for Research
4. SAVE Tool
5. Cyber-Physical Systems Application
6. Summary

1. MOTIVATIONS

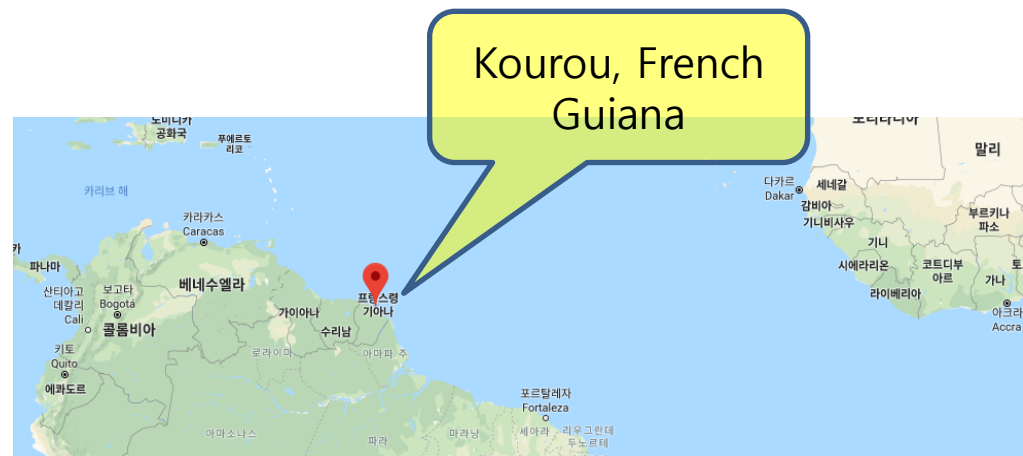
1. Motivation
 - 1) **Safety Requirements**
 - 2) SW Engineering Requirements
2. Modeling w/ Formal Methods
 - 1) Specification: dTP-Calculus
 - 2) Verification: GTS Logic
3. Probability Choice to Control Nondeterminism
 - 1) Motivation
 - 2) Definition
 - 3) Smart City Example: SEES Example
 - 4) Open Issues for Research
4. SAVE Tool
5. Cyber-Physical Systems Application
6. Summary

Motivation: Safety Requirements

MISSION-CRITICAL SYSTEMS

Year **1996**, ESA Ariane Flight 5 Failure

- Description:
 - 1996, ESA Ariane 5 Flight Exploded, at 40 seconds after launch.
- Cause of explosion:
 - A part of the flight SW, reused from Ariane 4.
 - One of the sensors in the flight 5 was designed to return a float point value.
 - But it used the code for the sensor from the Ariane 4 that was to return an integer value, which did not have an exception handler for overflow due to the inappropriate type of the return value.
 - As a result, the exception triggered a signal to the self-explosion of the flight.
- Total cost of the failure:
 - Hardware Cost: 0.37 Billion \$
 - Development Cost: 7 Billion \$.
 - Cluster II: 4 identical satellites to study the Earth's magnetosphere over the course of an entire solar cycle.



Cause: Reused Ada Source Code

Reused Ariane 4 Code for Ariane 5

```
1 procedure LIRE_DERIVE (...) is
2   ...
3   begin
4     ...
5     L_MBV_32:=TBD.T_ENTIER_32S ((1.0/C_M_LSB_BV)*
6                                   G_M_INFO_DERIVE(T_ALG.E_BV));
7     if L_M_BV_32 > 32767 then
8       P_M_DERIVE(T_ALG.E_BV) := 16#7FFF#;
9     elsif L_M_BV_32 < -32768 then
10      P_M_DERIVE(T_ALG.E_BV) := 16#8000#;
11    else
12      P_M_DERIVE(T_ALG.E_BV) := UC_16S_EN_16NS(TBD.T_ENTER_16S(L_M_BV_32));
13    end if;
14
15    P_M_DERIVE(T_ALG.E_BH) := UC_16S_EN_16NS(TBD.T_ENTER_16S
16      ((1.0/C_M_LSB_BH)*
17      G_M_INFO_DERIVE(T_ALG.E_BV)));
18  end LIRE_DERIVE;
```

The internal SRI software exception was caused during execution of a data conversion from 64-bit floating point to 16-bit signed integer value. The floating point number which was converted had a value greater than what could be represented by a 16-bit signed integer.*

```
1 procedure LIRE_DERIVE (...) is
2   ...
3   begin
4     ...
5     L_MBV_32:=TBD.T_ENTIER_32S ((1.0/C_M_LSB_BV)*
6                                   G_M_INFO_DERIVE(T_ALG.E_BV));
7     if L_M_BV_32 > 32767 then
8       P_M_DERIVE(T_ALG.E_BV) := 16#7FFF#;
9     elsif L_M_BV_32 < -32768 then
10      P_M_DERIVE(T_ALG.E_BV) := 16#8000#;
11    else
12      P_M_DERIVE(T_ALG.E_BV) := UC_16S_EN_16NS(TBD.T_ENTER_16S(L_M_BV_32));
13    end if;
14
15    L_M_BH_32 := TBD.T_ENTIER_32S ((1.0/C_M_LSB_BH)*
16      G_M_INFO_DERIVE(T_ALG.E_BH));
17    if L_M_BH_32 > 32767 then
18      P_M_DERIVE(T_ALG.E_BH) := 16#7FFF#;
19    elsif L_M_BH_32 < -32768 then
20      P_M_DERIVE(T_ALG.E_BH) := 16#8000#;
21    else
22      P_M_DERIVE(T_ALG.E_BH) := UC_16S_EN_16NS(TBD.T_ENTER_16S(L_M_BH_32));
23    end if;
24  end LIRE_DERIVE;
```

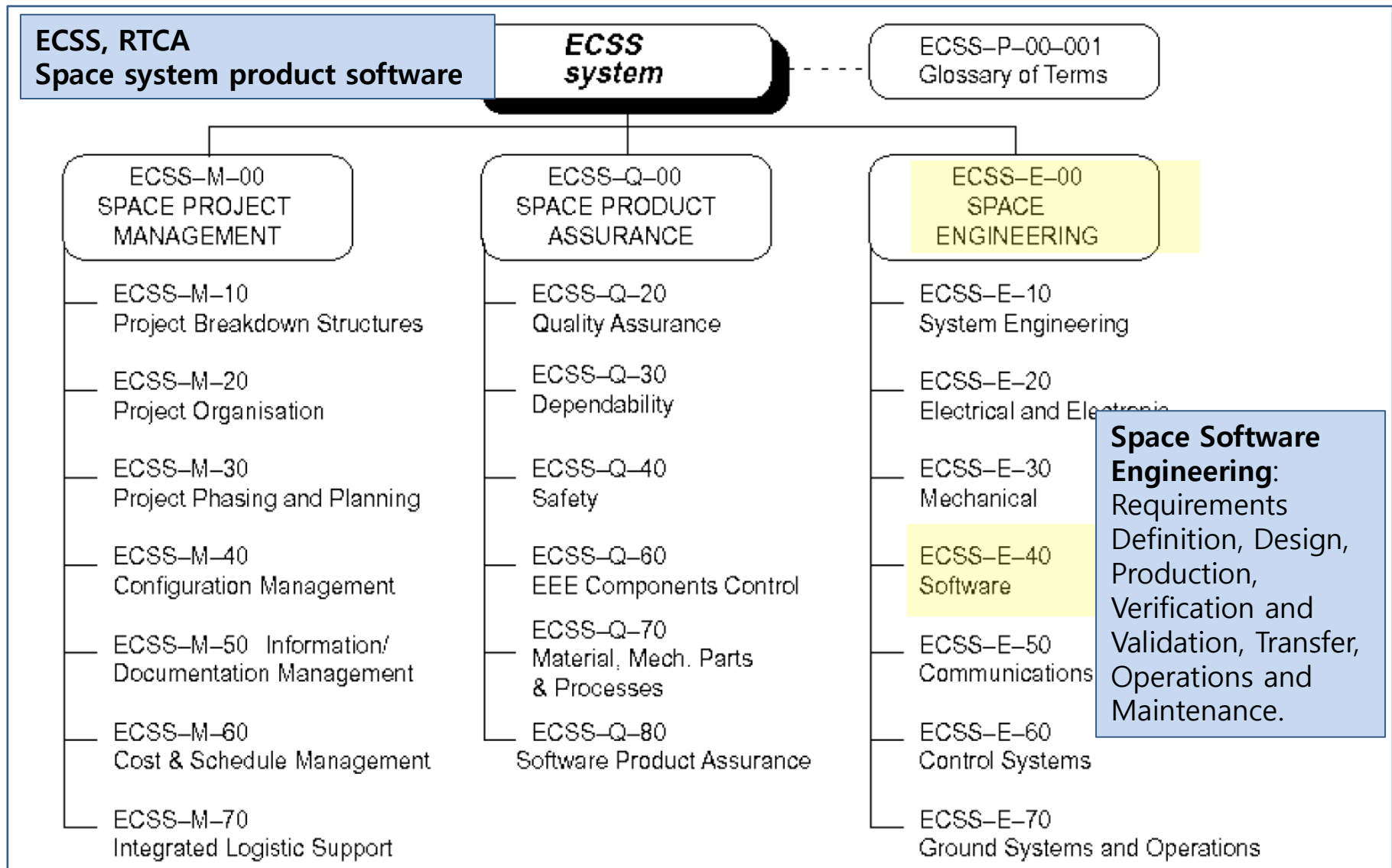
Standards

- ESA:
 - 1975
 - Paris, France
 - 22 member states
 - Budget: 5.250 Billion Euro in 2016.
- ECSS
 - Organization: 1993
- Standardization
 - Management: ECSS-M-00
 - Assurance: ECSS-Q-00
 - Engineering: ECSS-E-00
 - SW:
 - **ECSS-E-40: SW Engineering (Version A: 1999)**
 - Derived from ISO 12207 (
 - **ECSS-Q-80: SW Product Assurance (Version A: 1996)**
- **ECSS-E-40**
 - SW Engineering
 - SW Engineering Process
 - Model-Based SW Engineering
 - Formal Methods
 - Tool-Based SW Engineering
 - SW life Cycle Process
- Results:
 - ~ 2013년
 - Total 67 launches; 4 failures.



ECSS-E-40C

(~2009)



Year 2018/19 Boeing 737 Max MCAS

- Boeing 737 Max, MCAS SW
 - Boeing's newest family of single-aisle airplanes
 - The fastest-selling airplane in Boeing history
 - About 5,000 orders
 - from more than 100 customers worldwide.
- MCAS
 - The Maneuvering Characteristics Augmentation System
 - It activates when the sensed Angle of Attack (AOA) exceeds a threshold based on airspeed and altitude.
 - That tilts the 737 Max's horizontal stabilizer upward at a rate of .27 degrees per second for a total travel of 2.5 degrees in just under 10 seconds.
 - How much the stabilizer moves depends on Mach number.
 - At higher Mach the stabilizer moves less, at slower speeds it moves more.
 - The trim system under MCAS is not stopped by simply moving the control yoke.
- Accidents
 - Indonesia Lion Jet 610 [Oct 2018]: All 189 passengers dead.
 - Ethiopian Airline 302 [March 2019]: All 157 passengers dead.
- Possible cause
 - Malfunction at sensors for angle between the wings and the air current.
 - MCAS engaged to handle the situation, and it causes the nose to drop.
 - The pilots tried to hold back the flight control, but failed.
 - The planes crashed



**Close
Investigation**

Boeing 737 Max Maneuvering Characteristics Augmentation System

Activates automatically when:

- Angle of attack is high
- Autopilot is off
- Flaps are up
- Steeply turning

MCAS pushes the jet's nose down to reduce the risk of stalling



Deactivates when:

- Angle of attack is sufficiently lowered
- Pilots override with manual trim

 THE AIR CURRENT

<https://theaircurrent.com/aviation-safety/what-is-the-boeing-737-max-maneuvering-characteristics-augmentation-system-mcas-jt610/>



DO-178C (USA)

(~2013)

RTCA, EUROCAE

Software Considerations in Airborne Systems and Equipment Certification

DO-178B: Software Requirements and Software Design

DO-178C: + Object-oriented Programming

ISSUE LIST

Recommendation

DO-278
/ED-109

DO-178C
ED-12C

DO-178C
ED-12C
Interface Spec
for
Supplements
and Rationale
Documents

Interface
Spec.

DO-330: Software Tool Qualification Considerations

DO-331: Model-Based Development and Verification

DO-332: Object-Oriented Technology and Related Techniques

DO-333: Formal Methods

Supplement A

Supplement B

Supplement C

Supplement

Supplement N

DO-248B/ED-94B
FAQ/DP/RATIONALE

DO-248C/ED-94C
FAQ/DP/RATIONALE

International Standard Organizations and Their Standards

- ANSI: American National Standards Institute
- AIAA: American Institute of Aeronautics and Astronautics
- EIA: Electronic Industries Association
- IEC: International Electrotechnical Commission
- IEEE: Institute of Electrical and Electronics Engineers, Computer Society, SW Engineering Standards Committee
- ISO: International Organization for Standardization
- RTCA: Radio Technical Commission for Aeronautics
- EUROCAE: European Organization for Civil Aviation Equipment
- ECSS: European Cooperation for Space Standardization

Aerospace/ Avionics Systems	Railway Systems	Nuclear Power Plants	Automobile Systems	Embedded Systems	Defense Systems	Quality Business Mgmt
ECSS-E-40 DO-178C IEEE 12207 AIAA G-010- 1993 CMMI	EN-50128	IEC-60880	ISO-26262 IEC 61508	IEC-61508	MIL-STD- 882E	ISO9001

1. Motivation
 - 1) Safety Requirements
 - 2) SW Engineering Requirements**
2. Modeling w/ Formal Methods
 - 1) Specification: dTP-Calculus
 - 2) Verification: GTS Logic
 - 3) Smart City: EMS Example
3. Probability Choice to Control Nondeterminism
 - 1) Motivation
 - 2) Definition
 - 3) Smart City Example: SEES Example
 - 4) Open Issues for Research
4. SAVE Tool
5. Cyber-Physical Systems Application
6. Summary

Motivation: SW Engineering Requirements from ECSS Standard & Others

SW Engineering Process Models

R1

Prescriptive Process Model

Waterfall Model

Incremental Model

RAD Model

Evolutionary Model

UP Model

Agile Process Model

Extreme Programming

ASD Model

DSDM

SCRUM

Agile Modeling

R2

Generic Engineering Process

Communication

Planning

Modeling

Construction

Deployment

Modeling/Formal Methods

R3

Logic

Z

Temporal logic

I/O Automata

CASL
(Common Algebraic
Specification Language)

State Machine

LTL
(Linear temporal logic)

CTL
(Computational tree
logic)

ASM
(Abstract state Machine)

Actor model

Process Algebra

Pi-Calculus

CCS

CSP

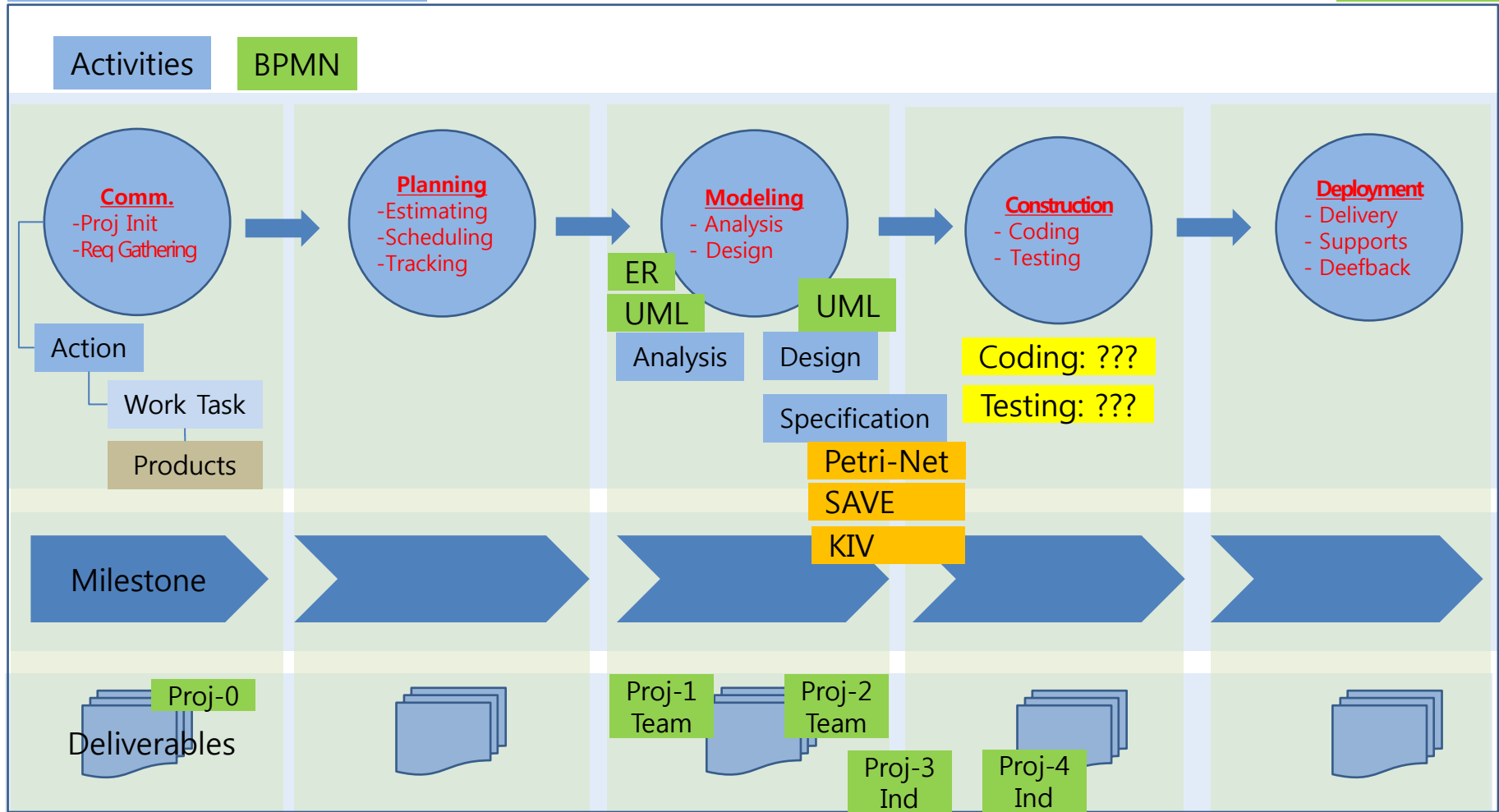
δ -Calculus

R4

TOOLS

R5

Object-Oriented Paradigm



Umbrella Activities

- SW Project Management
- SW Quality Assurance
- SW Configuration Management
- Risk Management Management