IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Lab-STICC
P4S

P4S meeting

SFE presentation:

Analysis and management of the co-evolution of systems and their security

Student : Pascal Douville de Franssu
Company tutor : Salvador Martinez
School tutor : Antoine Beugnard

19/09/2024

Assignment presentation

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Lab-STICC
P4S

Thesis subject "Model-Based Security Consistency Management for Complex Systems"

Objectives:

- Assessing a system's level of security
- Enabling co-evolution between system deployed artifacts and design models

Internship:

- System: Open-source java project
- Model: Security Data Flow Diagram (secDFD)

Security in the early stages of development

Context :

- Model-Driven Software Engineering (MDSE)

Benefits :

- Early flaw detection
- Production cost savings

Security modelling

Context :

- Data Flow Diagram

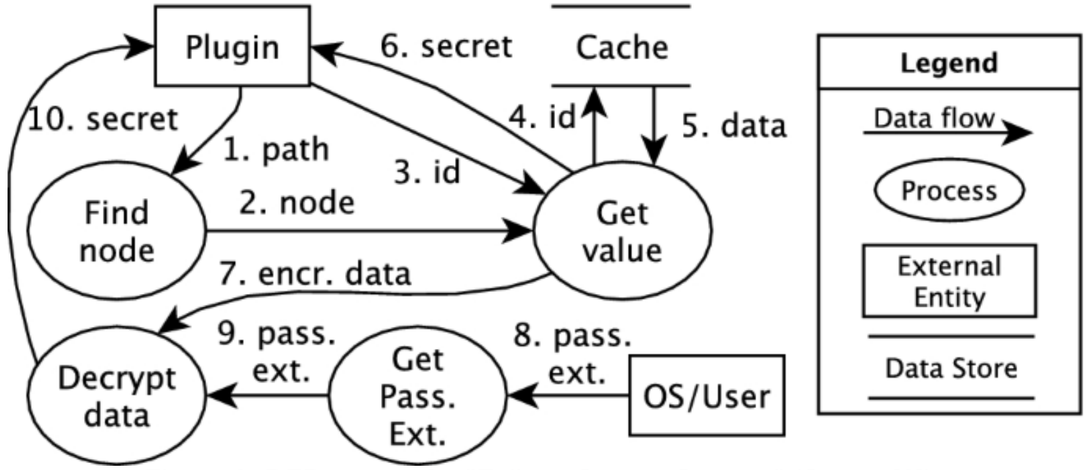

Figure 2: DFD extract for "Eclipse Secure Storage" (Source[6])

Security modelling

Adjustment:

- Security Data Flow Diagram

Benefits:

- Security reification
- Enabling traceability

Drawbacks:

- Complexity

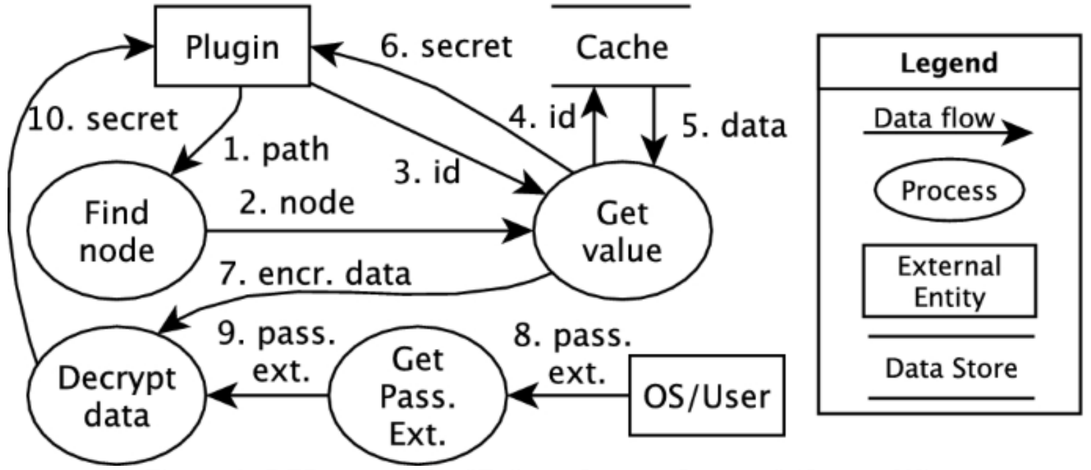

Figure 2: DFD extract for "Eclipse Secure Storage" (Source[6])

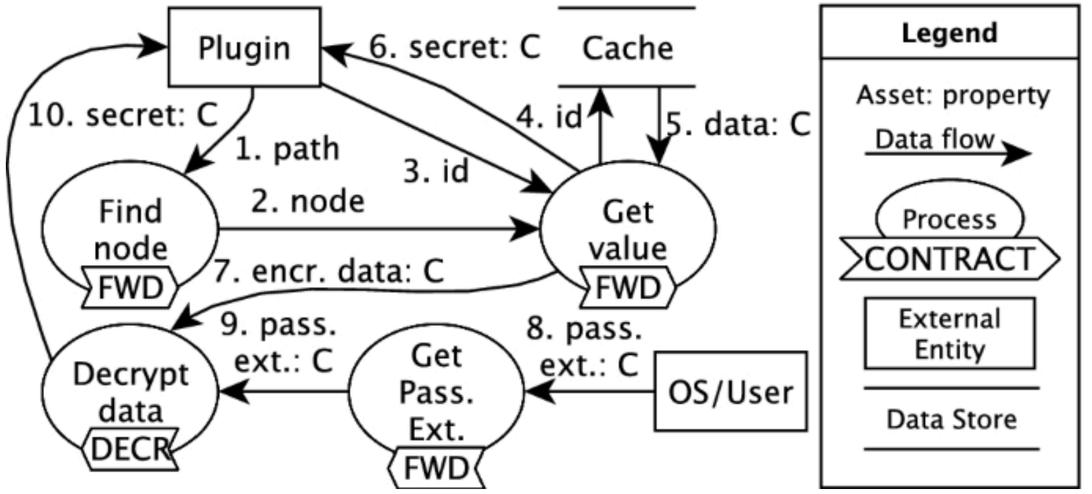

Figure 3 : Extrait de secDFD correspondant à "Eclipse Secure Storage" (Source[6])

Consistency

Problem:

- Model and code divergence

Solutions:

- Links between model and code
- Co-evolution by change detection and propagation

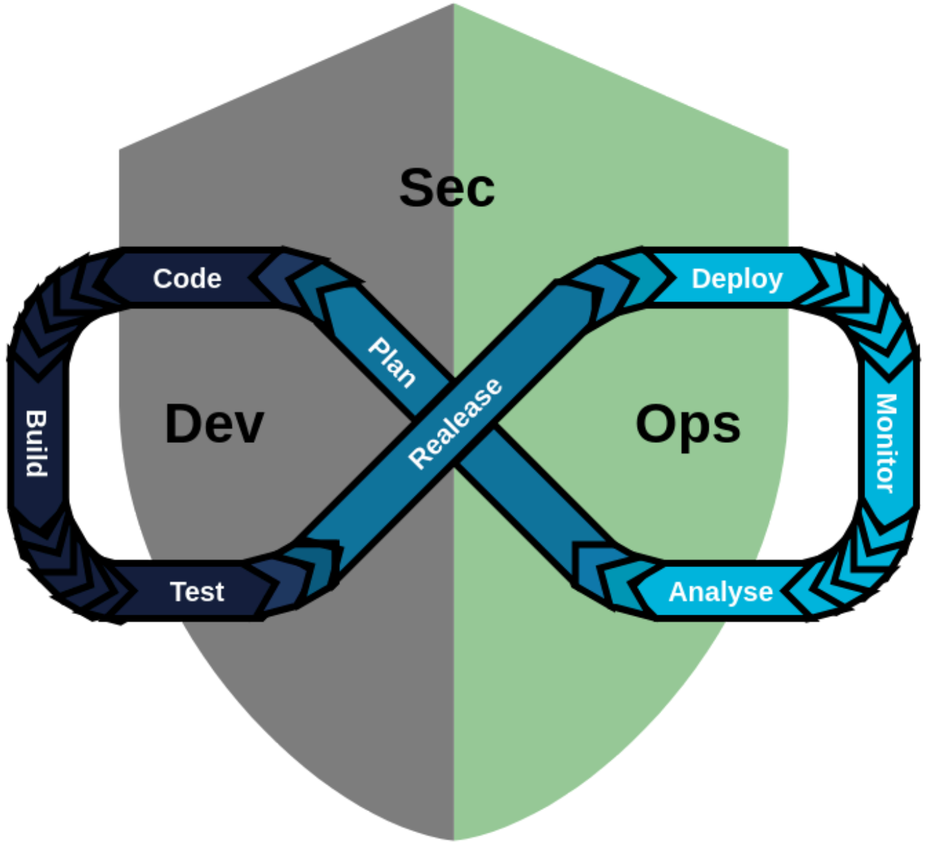

Figure 4 : DevSecOps representation

Automation

Constraints:

- Short DevSecOps cycles

Solution: automation

- Links between model and code
- Security level assessment
- Maintaining consistency

Potential requirements :

- Standardisation

Objective :

- Reproduce a semi-automatic model-code consistency checking solution
- Introduce the detection and propagation of modifications

Solution to be reproduced: K. Tuma et al. “Checking security compliance between models and code” [6]

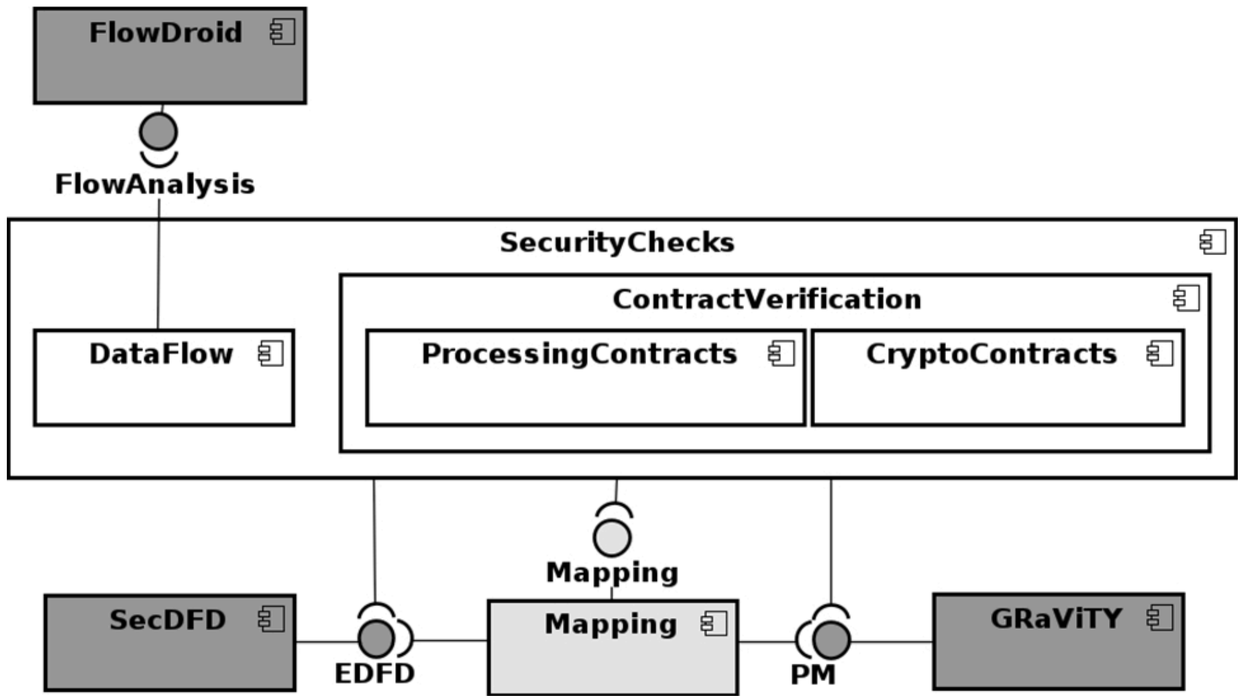

Figure 5: Solution architecture (Source[6])

Architecture and implementation

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Lab-STICC
P4S

Spoon

Usage:

- AST extraction
- Code refactoring

Benefits:

- Proximity with the code
- Performance
- Ease of use
- Model persistency

Drawbacks:

- AST complexity and size

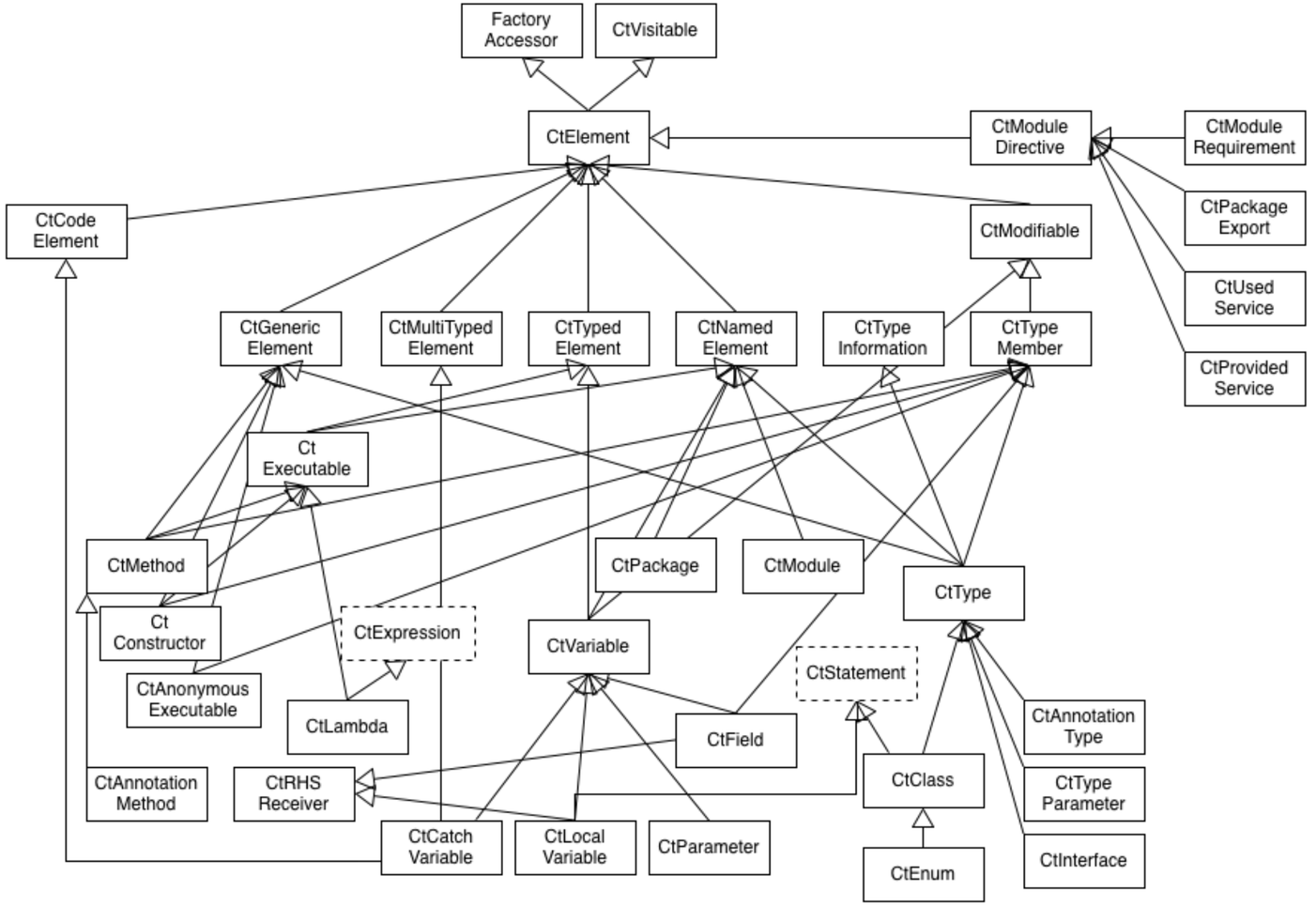

Figure 6: Spoon structural elements metamodel (Source[3])

Xtext/EMF

Usage:

- Xtext: (domain-specific)Language workbench
- EMF: Meta-modeling

Benefits:

- Read textual ‘.secdfd’ files
- Change detection
- Large compatibility

Drawbacks :

- Learning curve

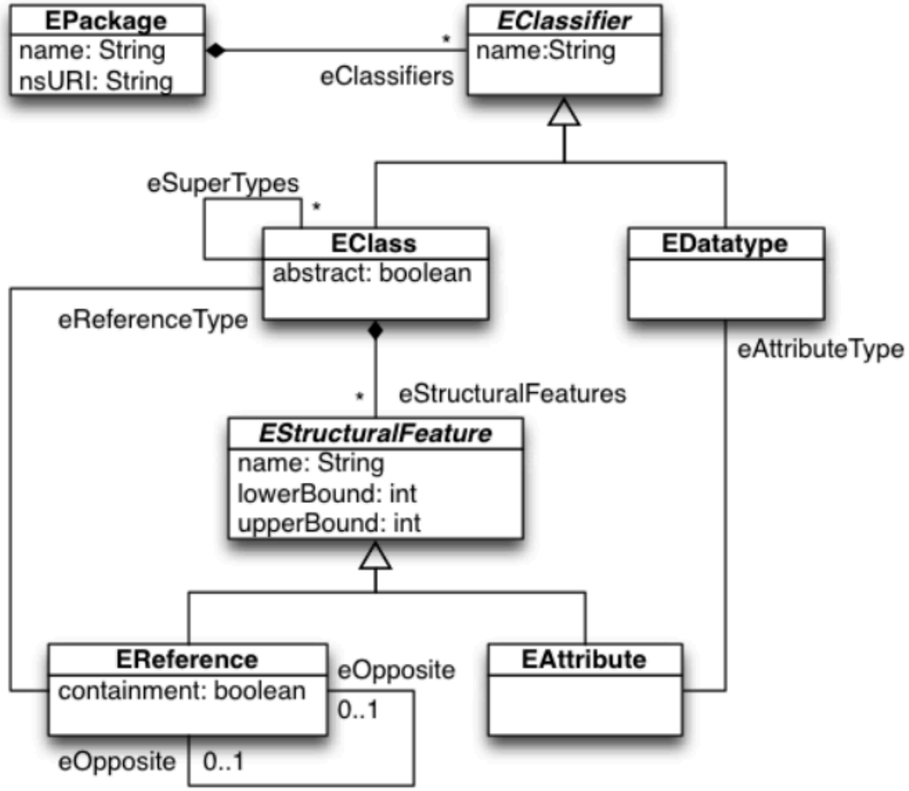

Figure 7: Main elements of the EMF meta-model (Source [2])

Epsilon

Usage:

- Model management

Benefits:

- EMF compatibility
- Reification

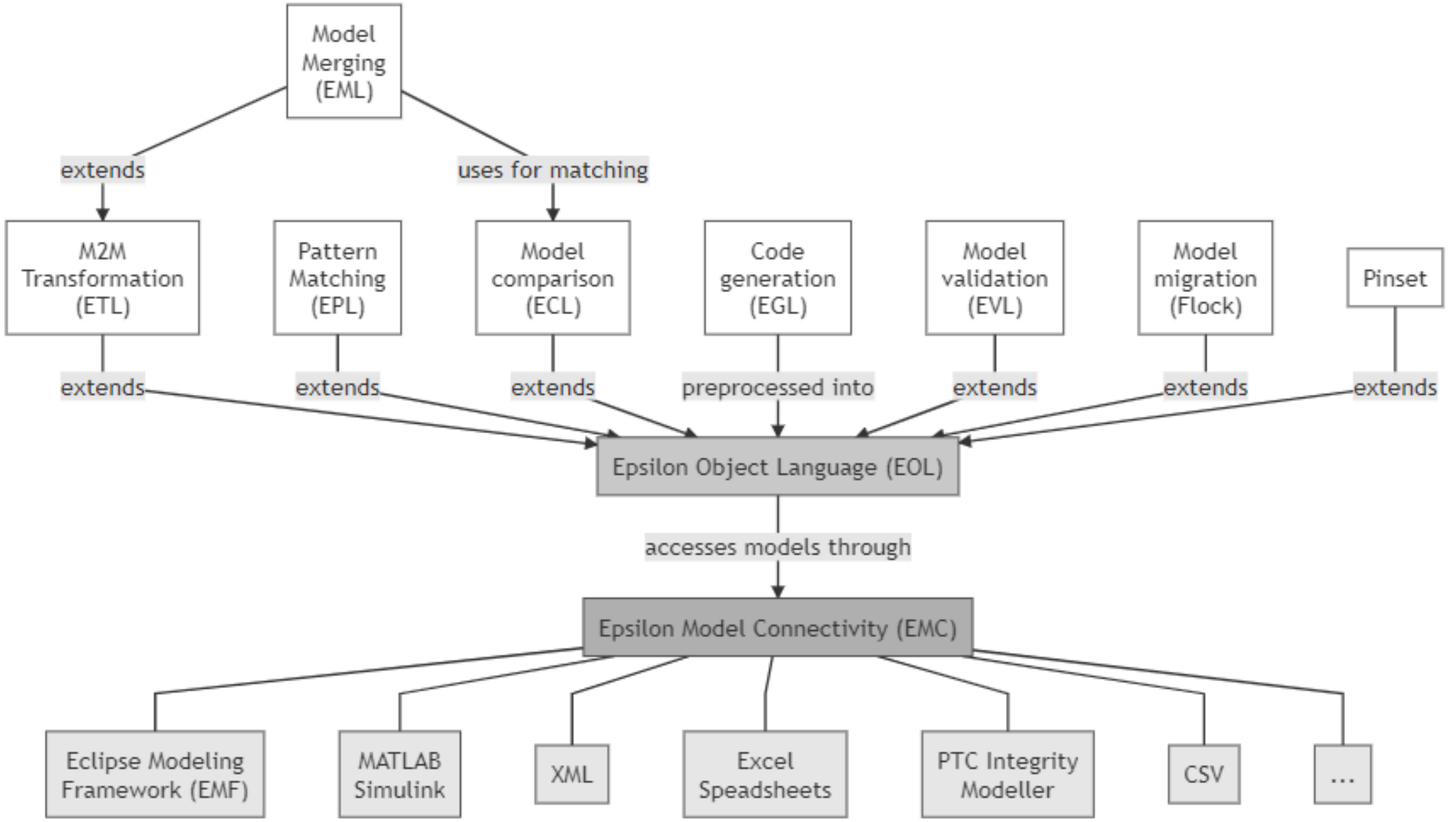

Figure 8: Epsilon's overall structure (Source[1])

b) Architecture

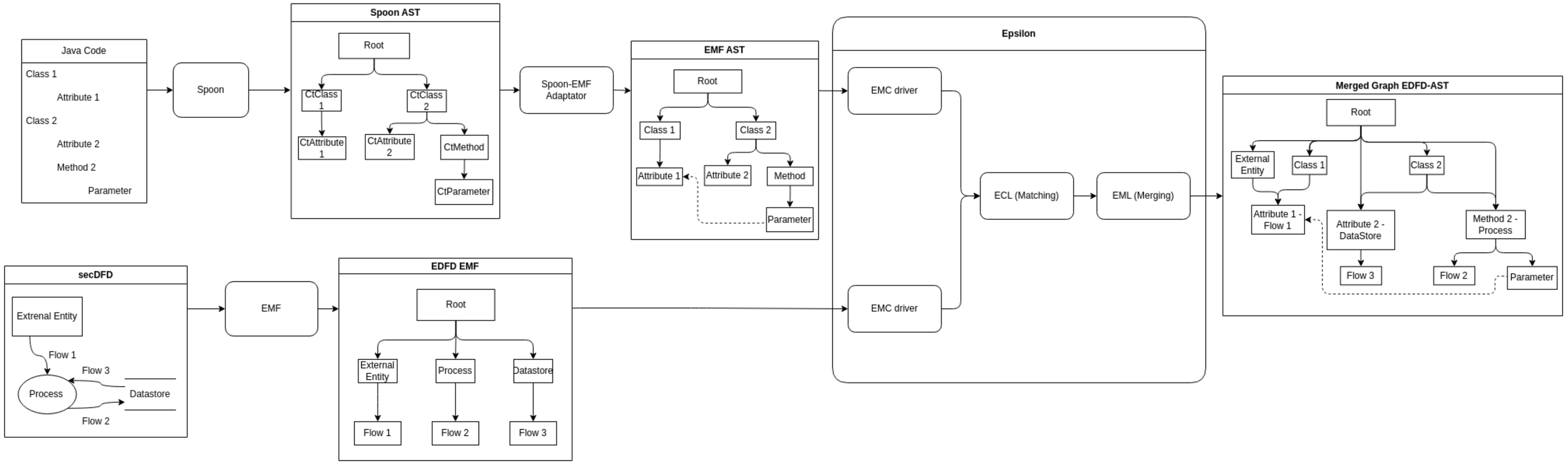

Figure 9: Solution architecture

Thanks for your attention!

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Lab-STICC
P4S

REFERENCES

[1] Eclipse. Epsilon documentation. https://eclipse.dev/epsilon/doc/, 2024. Last accessed 06 September 2024.
[2] Eclipse. Xtext documentation. https://eclipse.dev/Xtext/documentation/index.html, 2024. Last accessed 06 September 2024.
[3] Spoon. Source code analysis and transformation for java. https://spoon.gforge.inria.fr/index.html, 2024. Last accessed 06 September 2024.
[4] Lab-STICC. P4s. https://labsticc.fr/fr/equipes/p4s, 2024. Last accessed 06 September 2024.
[5] Maven. Wlecome page. https://maven.apache.org/index.html, 2024. Last accessed 06 September 2024.
[6] Katja Tuma, Sven Peldszus, Daniel Strüber, Riccardo Scandariato, and Jan Jürjens. Checking security compliance between models and code. Software and Systems Modeling, 22(1) :273–296, Feb 2023.