# The ASCOLA team:
# Correct and secure software
# Efficient infrastructures and applications

Jean-Claude Royer

ASCOLA Group, Mines de Nantes - INRIA Rennes, Nantes France
Lead: Mario Südholt
<prenom>.<nom>@mines-nantes.fr

# The team

Context

- One of three teams of the CS department at EMN
  Two other groups: constraint programming/optimization, model-driven engineering
- Joint team with **EMN, INRIA** and **LINA** (UNantes/CNRS)

30 members

- 11 permanent staff (8 EMN, 2 Inria, 1 Polytech), 2 MA associés
- 2 post-docs/engineers (1 Inria)
- 15 PhDs (3 co-supervisions: TU Darmstadt, Lancaster U., U. Chili)

Overall objective: **Flexible and correct evolution of large-scale infrastructures and applications**

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Approach

- Integration and evolution of large-scale software systems
  - New notions of interfaces/modules/components
  - Non-modular functionalities: Energy, security, transactions, ...
  - Make explicit dependencies among software entities
- Correct and automated software development
  - Automatic assembly of software components and aspects
  - From declarative specifications to efficient implementations
  - Efficient management of infrastructures using high-level abstractions, e.g., for the virtualization of data centers
  - Formal methods for correct application assembly and evolution
- Language-based approach

# Actions related to “security”

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO “Service availability”, 2003-2006
- ANR CESSA “Secure evolution of SOAs”, 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD “Accountability for the Cloud”, 2012-2016

# Actions related to “security”

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO “Service availability”, 2003-2006
- ANR CESSA “Secure evolution of SOAs”, 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD “Accountability for the Cloud”, 2012-2016

# Actions related to "security"

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO "Service availability", 2003-2006
- ANR CESSA "Secure evolution of SOAs", 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD "Accountability for the Cloud", 2012-2016

# Actions related to "security"

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO "Service availability", 2003-2006
- ANR CESSA "Secure evolution of SOAs", 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD "Accountability for the Cloud", 2012-2016

# Actions related to "security"

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO "Service availability", 2003-2006
- ANR CESSA "Secure evolution of SOAs", 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD "Accountability for the Cloud", 2012-2016

# Actions related to "security"

- We are not security experts
- Integration of security solutions with a language based approach
- ACI security DISPO "Service availability", 2003-2006
- ANR CESSA "Secure evolution of SOAs", 2010-2013
- SecCloud COMIN Labs, 2012-2014, `http://www.cominlabs.ueb.eu/themes/project/`, Aspectizing Javascript security
- IP A4CLOUD "Accountability for the Cloud", 2012-2016

# Secure evolution of SOAs

Different challenges for security in the Cloud

- **Service composition in heterogeneous service environments** (server vs. mobile . . . )
  - ANR project CESSA (Jan. 2010-13) with SAP, Eurecom, IS2T
- **Client-side programming** (secure JavaScript . . . )

Ex.: **evolution of financial services in a service marketplace**

- Stricter regulation for financial services (e.g., Sarbanes-Oxley): modify service infrastructures and applications
- Applications
  - Evolution of a **loan negotiation application**
  - Adaptation of **confidentiality policies**
- AOP - reference monitor, type system, correct marshalling/serialization of services

# IP A4CLOUD

- http://www.a4cloud.eu/
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud

# IP A4CLOUD

- `http://www.a4cloud.eu/`
- SAP, HP, CSA, SINTEF, QMUL, Eurecom, Tilburg, U. Malaga, ...
- Multidisciplinary project: Laws, regulatory constraints, norms in the cloud and computer science technical means to enforce them
- Limit of classic security means
- To complement classic security with accountability as in real life
- Pragmatically: Store evidences and auditing (proofs)
  - Abstract language for accountability
  - Policy enforcement, service adaptation
  - Evidence and verification
- Asp4CXF: An aspect framework for CXF
- Flexible Aspect-Based Service Adaptation for Accountability Properties in the Cloud