ascens

is about ...

ensembles

- achieve an overall system's goal
  - have a massive number of nodes
    - operate in open and non-deterministic environments
  - are built from self-aware components
- adapt dynamically to new conditions

engineering ensembles

- language for autonomic behavior
  - knowledge representation of self-aware components
    - mechanisms for adaptation
  - verification using formal methods
- set of tools and tool integration platform

ascens

visit us!

www.ascens-ist.eu

blog.ascens-ist.eu

Coordination

Prof. Dr. Martin Wirsing
ascens@pst.ifi.lmu.de

Partners

Ludwig-Maximilians-Universität München
Università di Pisa
Università di Firenze
Fraunhofer Gesellschaft
VERIMAG Laboratory
Università di Modena e Reggio Emilia
Université Libre de Bruxelles
Ecole Polythechnique Fédérale de Lausanne
Volkswagen AG
Zimory GmbH
University of Limerick – Lero
IMT Lucca
Mobsya
Charles University in Prague
CNR – ISTI

autonomic **s**ervice **c**omponent **ens**embles

a software engineering approach

based on formal methods

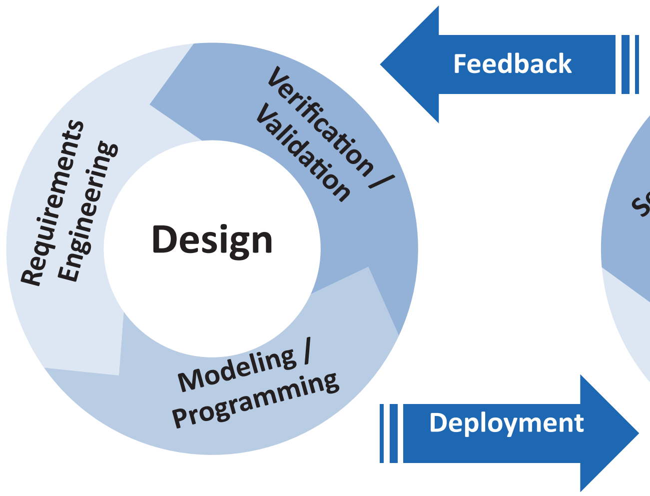
IST/FET Integrated Project
2010 – 2014
GA 257414

www.ascens-ist.eu

SEVENTH FRAMEWORK PROGRAMME

Ensemble Development Life Cycle

The EDLC is an iterative process that proposes a doubly connected design-runtime life cycle for the development of service component ensembles (SCE) characterized by self-* properties like self-awareness and self-adaptation.



Requirements Engineering

In this phase a conceptual and operational framework is provided to elicit and rationally represent adaptation and awareness requirements of ensembles.

➠ *ARE, SOTA*

Modeling/Programming

For the specification and coding of self-* properties of ensembles a set of languages were designed. They address how different components interact to form ensembles, their behavior, and knowledge manipulation according to specific policies.

➠ *ARGoS, BIP, DEECo/JDEECo, FACPL, Helena, IRM, JRESP, KnowLang, SCEL/MISSCEL*

Verification/Validation

Formal proofs of ensembles' models and code are proposed for planning and controlling execution.

➠ *BIP D-Finder, GMC, Iliad, jSAM, MESSI, MISSCEL*

Deployment

Static and hot deployment is supported.

➠ *ARGoS, BIP, DEECo/JDEECo, jRESP*

Monitoring

Both individual components of an ensemble and their environment are monitored using mechanisms at runtime to collect data for the purpose of awareness.

➠ *ARGoS, SPL (performance monitoring)*

Awareness

This phase comprises the knowledge of the system and its environment as well as the reasoning mechanisms that an enesemble can employ at runtime.

➠ *ARGoS, KnowLang, MATSim, POEM/Iliad*

Self-Adaptation

In case of awareness of malfunctions, contingencies or performance issues, the system evaluates possibilities of adaptation in form of re-configuration or self-expression.

➠ *SOTA patterns*

Feedback

The feedback transition takes data collected during monitoring back to the design phases.

➠ *IRM*

Case Studies

Science Cloud

The cloud computing scenario is designed as a Platform as a Service (PaaS) solution composed by autonomous nodes that are (self-)aware of

- changes in load;
- the network structure (i.e. nodes coming and going);
- the need of self-healing properties (network resilience).

The Science Cloud provides fail-over solution, i.e. self-adaptation or what we may call application execution resilience.

Swarm Robots

In the disaster recovery scenario a robot swarm is used to perform dangerous activities.

Part of the building has collapsed, trapping a number of victims inside. The autonomous robots must explore, search for victims, and collaborate for the rescue.

The robots must build a wall to screen themselves from a harmful radiation source.

Cooperative Vehicles

The e-Mobility scenario focuses on avoiding contingency situations in an open-ended and highly dynamic system.

The main components are the user, the electric vehicle, the parking lot and charging stations. Through-out runtime, contingency situations may occur. Components and ensembles require self-adaptive actions to resolve these situations.

ascens
autonomic service-component ensembles