Requirements-based Software System Adaptation in Practice: the Zanshin Framework

Vítor E. Silva Souza

Federal University of Espírito Santo
Vitória – ES, Brazil

vitorsouza@inf.ufes.br

http://www.inf.ufes.br/~vitorsouza

nemo
ontology & conceptual modeling research group

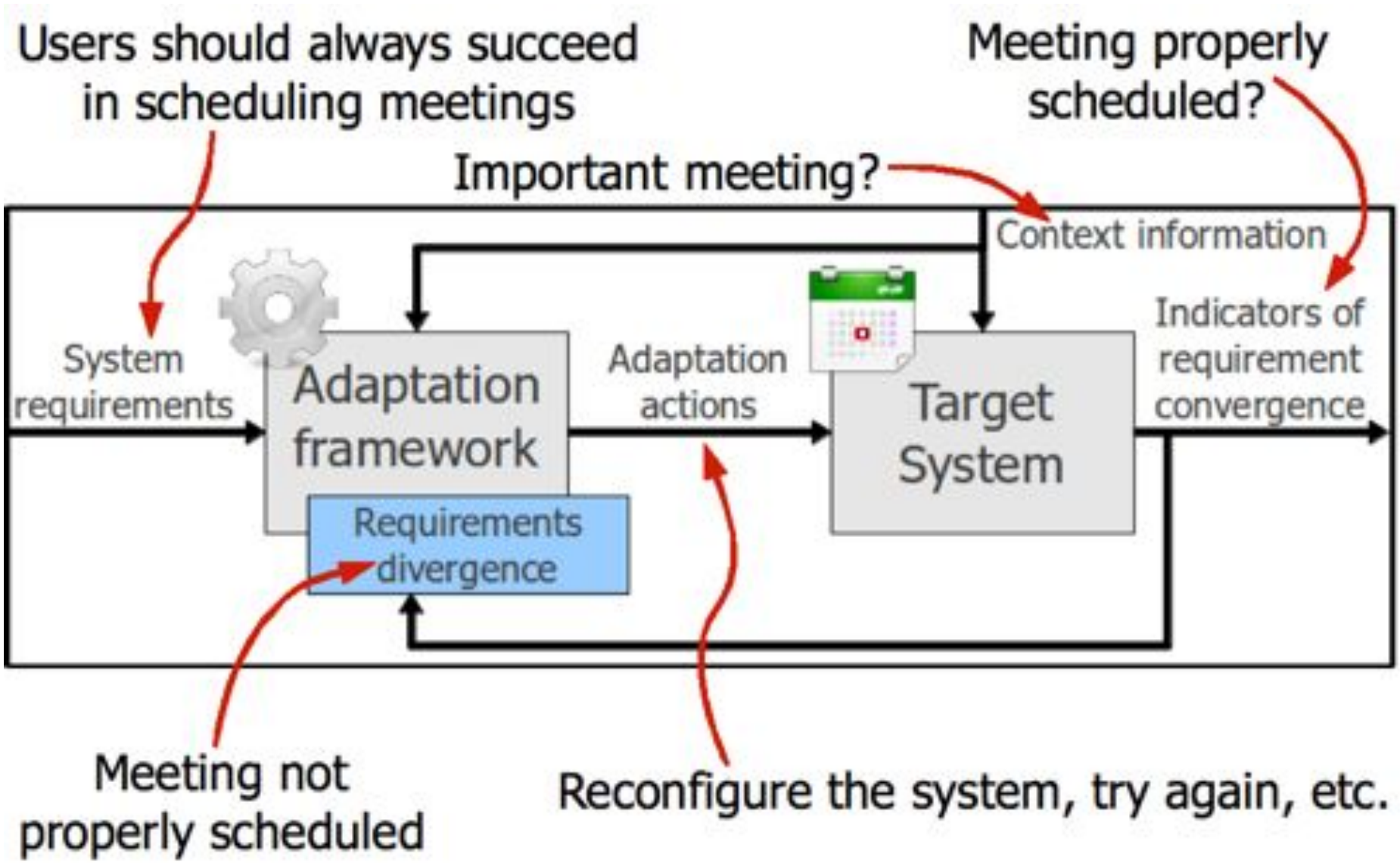
UFES

Lucretius: Foundations for Software Evolution
University of Trento, Italy

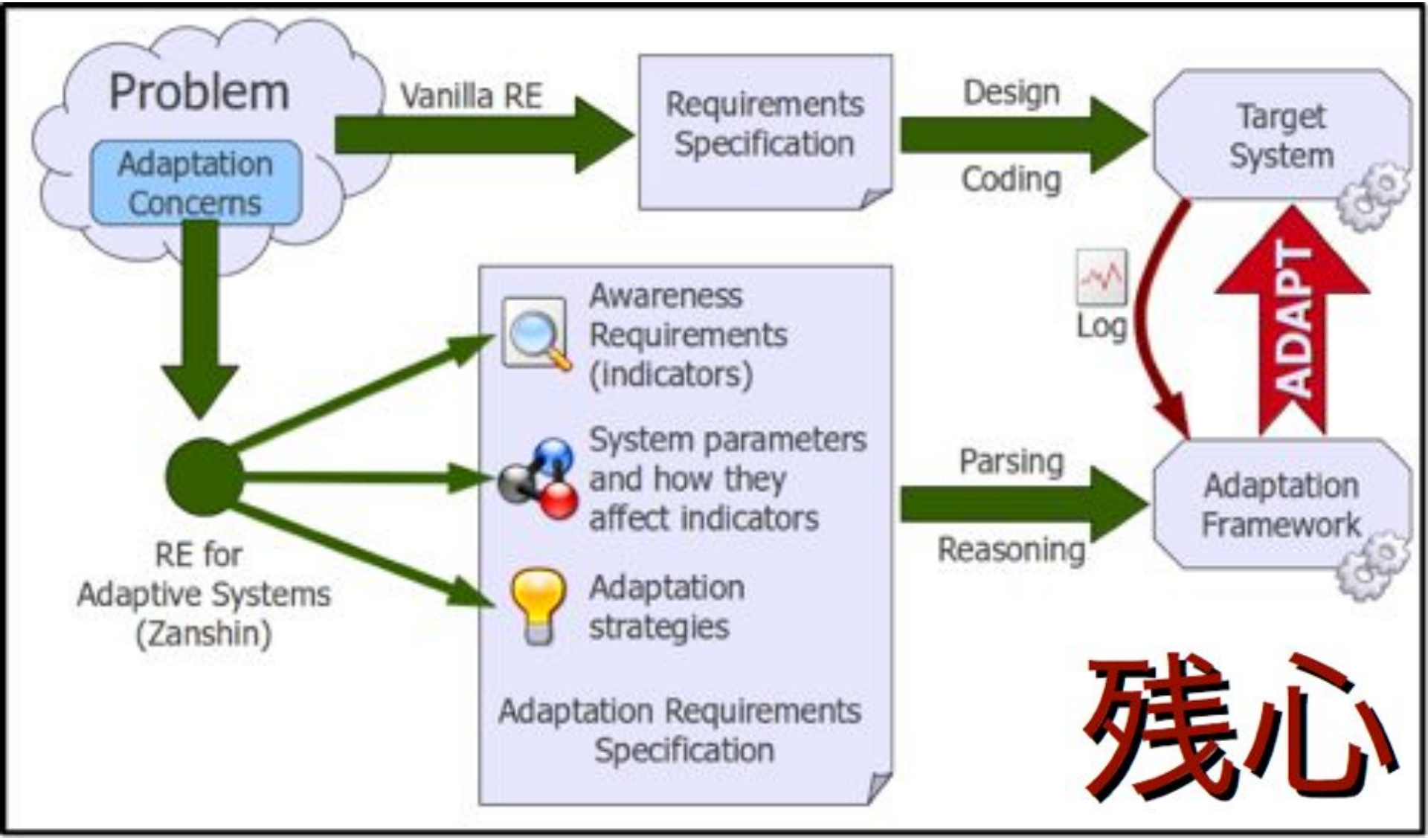
A bit of context

- Work done at the Univ. of Trento, Italy, during my PhD:
  - Advisor: John Mylopoulos;
  - Closest collaborators: Yiqiao Wang (Toronto), William N. Robinson (Georgia State), Alexei Lapouchnian (Toronto);
- Now currently a professor at the Federal University of Espírito Santo (Ufes) – Vitória, ES – Brazil.

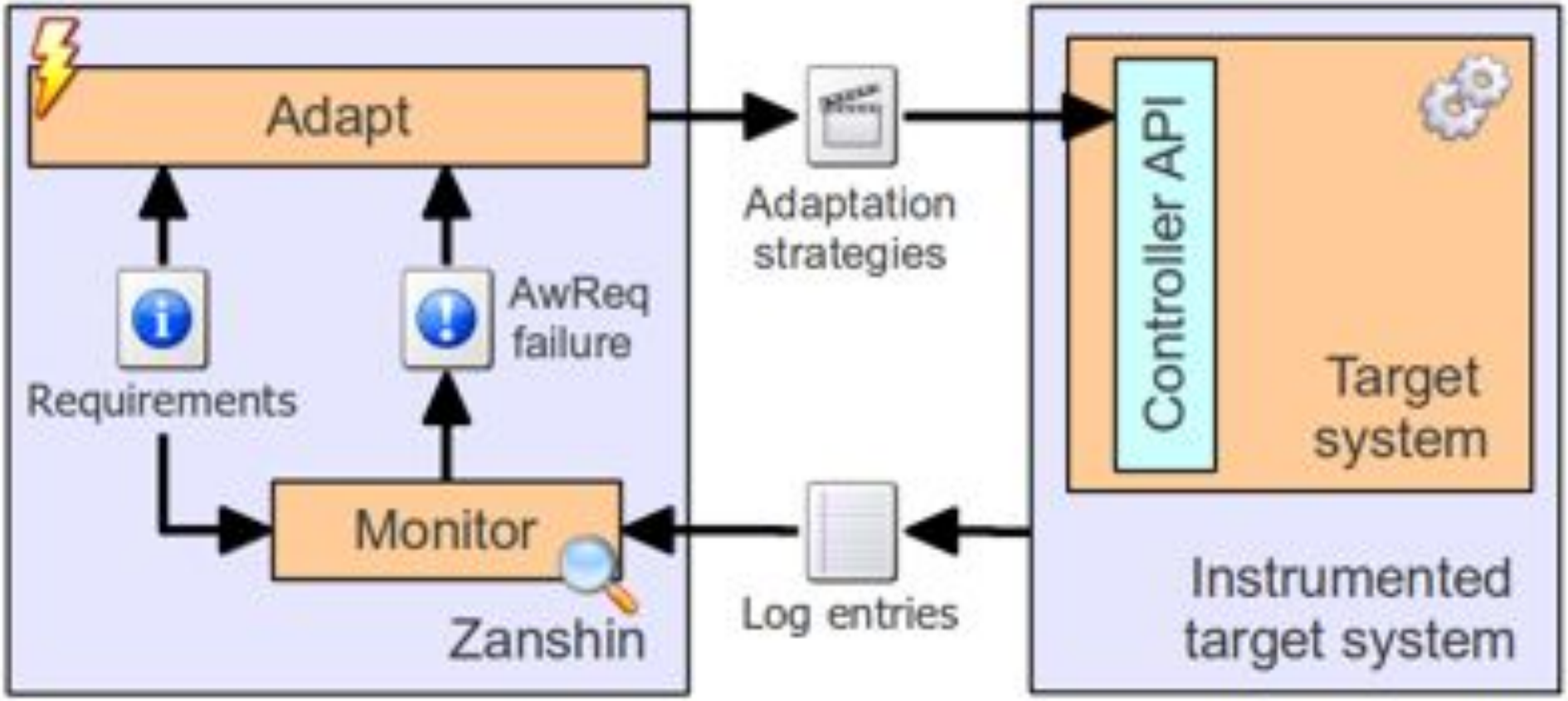
Adaptive systems and feedback control



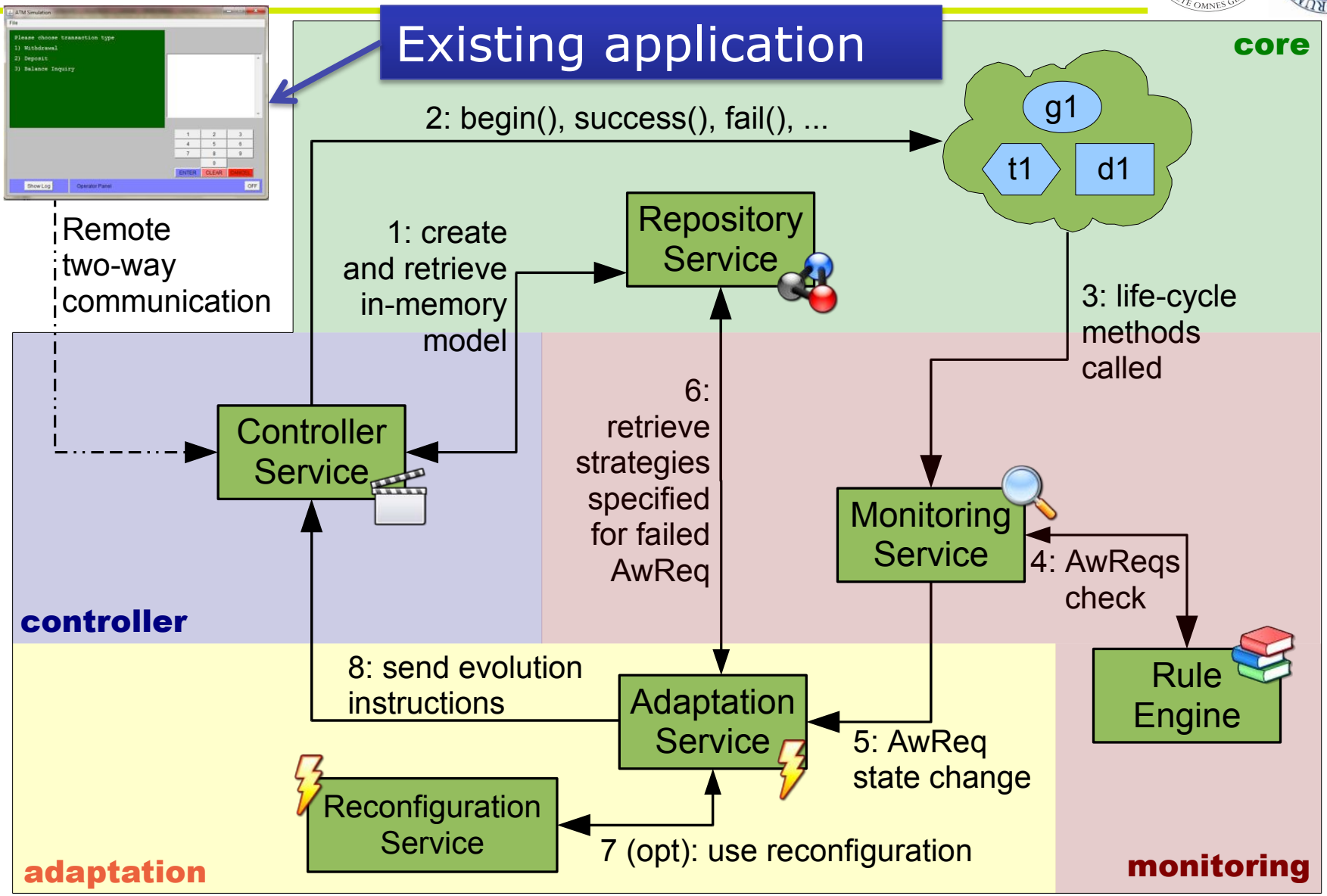
The *Zanshin* approach



The *Zanshin* framework



Current implementation: OSGi / Eclipse

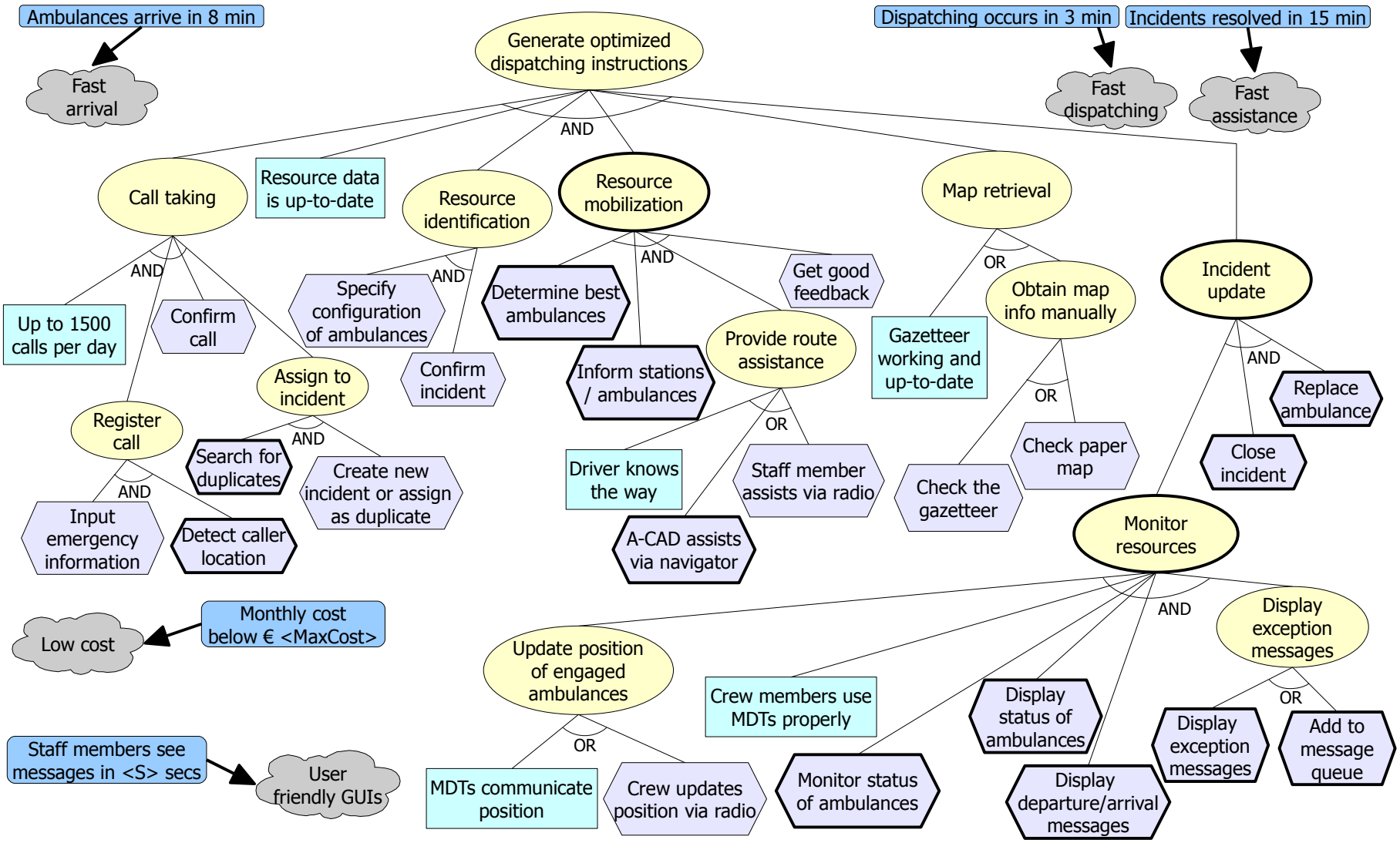


Experiments

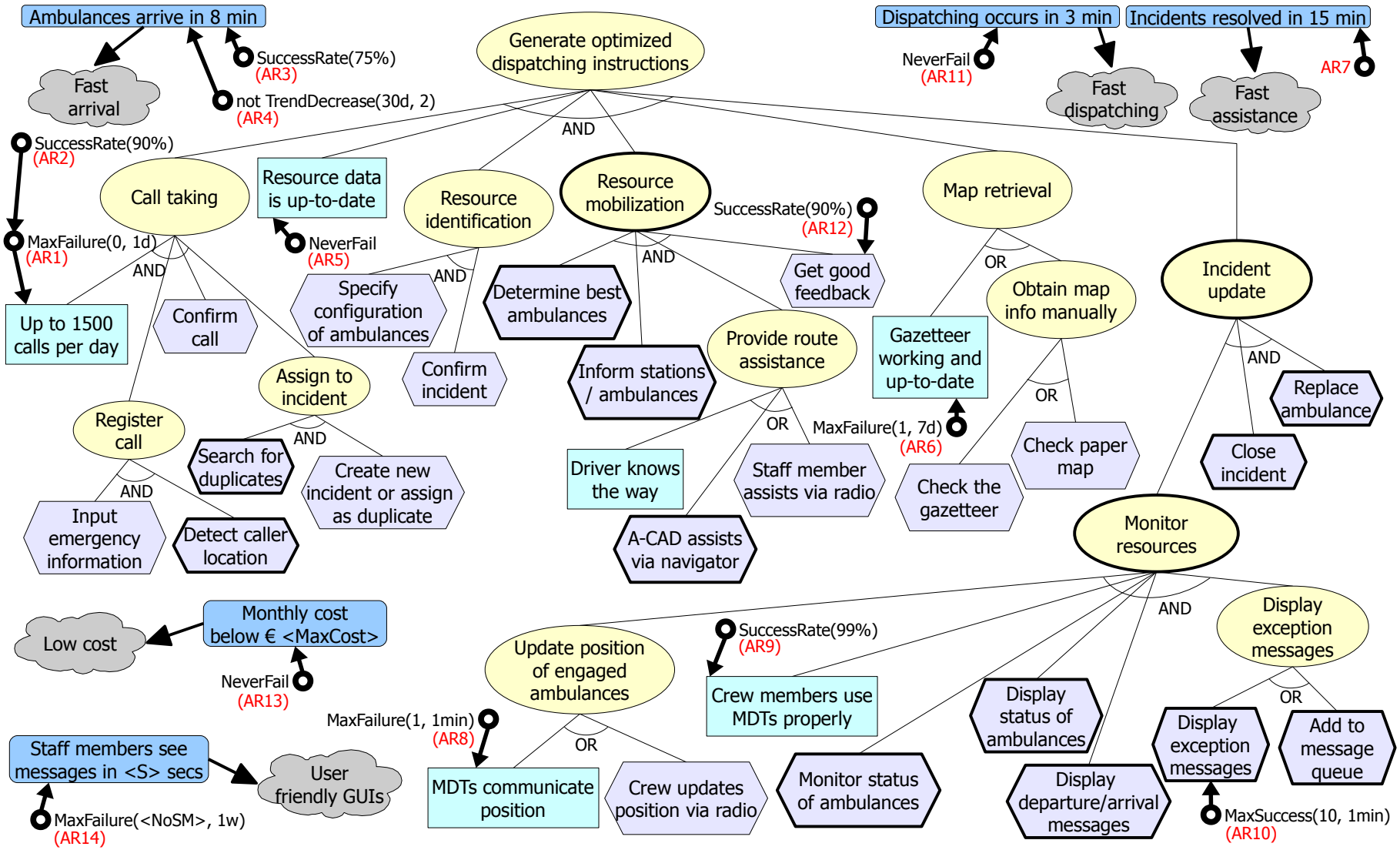
- Didactic: the Meeting Scheduler [ER'11, CSRD'12, PhD];
- Proof-of-concept: an Adaptive CAD, based on the LAS-CAD [SEfSAS2, CoopIS'12, A-CAD Tech. Rep., PhD];
- An external application: Russel Bjork's ATM simulation [SEAMS'13a];
- Comparison with an architecture-based approach (Rainbow): the *ZNN.com* slashdot effect [SEAMS'13b].

In the remainder of this talk, we will see how Zanshin works (an overview) using the A-CAD as example.

Baseline: goal model with variability



残心 Step 1. Awareness Requirements



Some *AwReqs* of the A-CAD

Aggregate

AwReq AR1 should succeed 90% of the time, considering month periods

AR 2

Aggregate

QC *Ambulances arrive in 8 min* should have 75% success rate;

AR 3

Trend

The success rate of that same QC should not decrease 2 months in a row

AR 4

Basic

QC *Dispatching occurs in 3 min* should never fail;

AR 11

Aggregate

Task Get good feedback should succeed 90% of the time

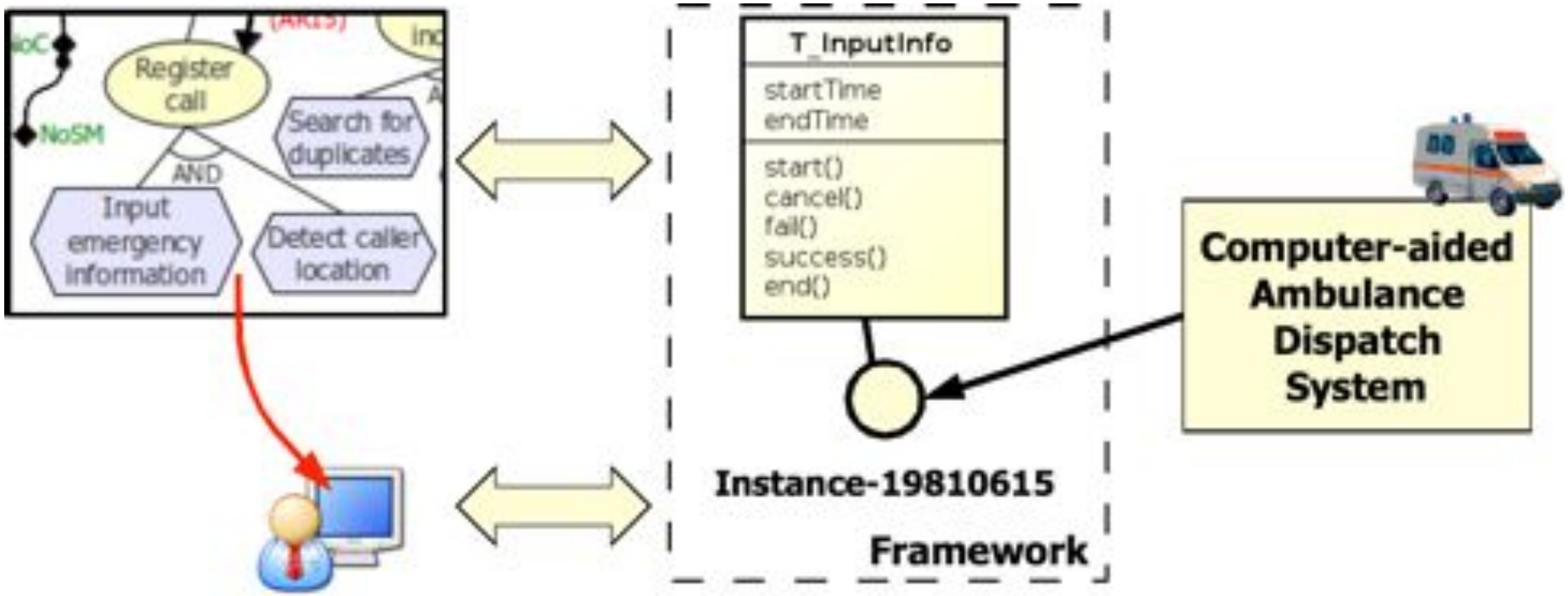
AR 12

Basic

Goal *Register call* should never fail;

AR 15

Run-time representation of requirements



Instance (for this dispatch)

"If the quality constraint that prescribes ambulances should arrive in 8 min fails, reconfigure".

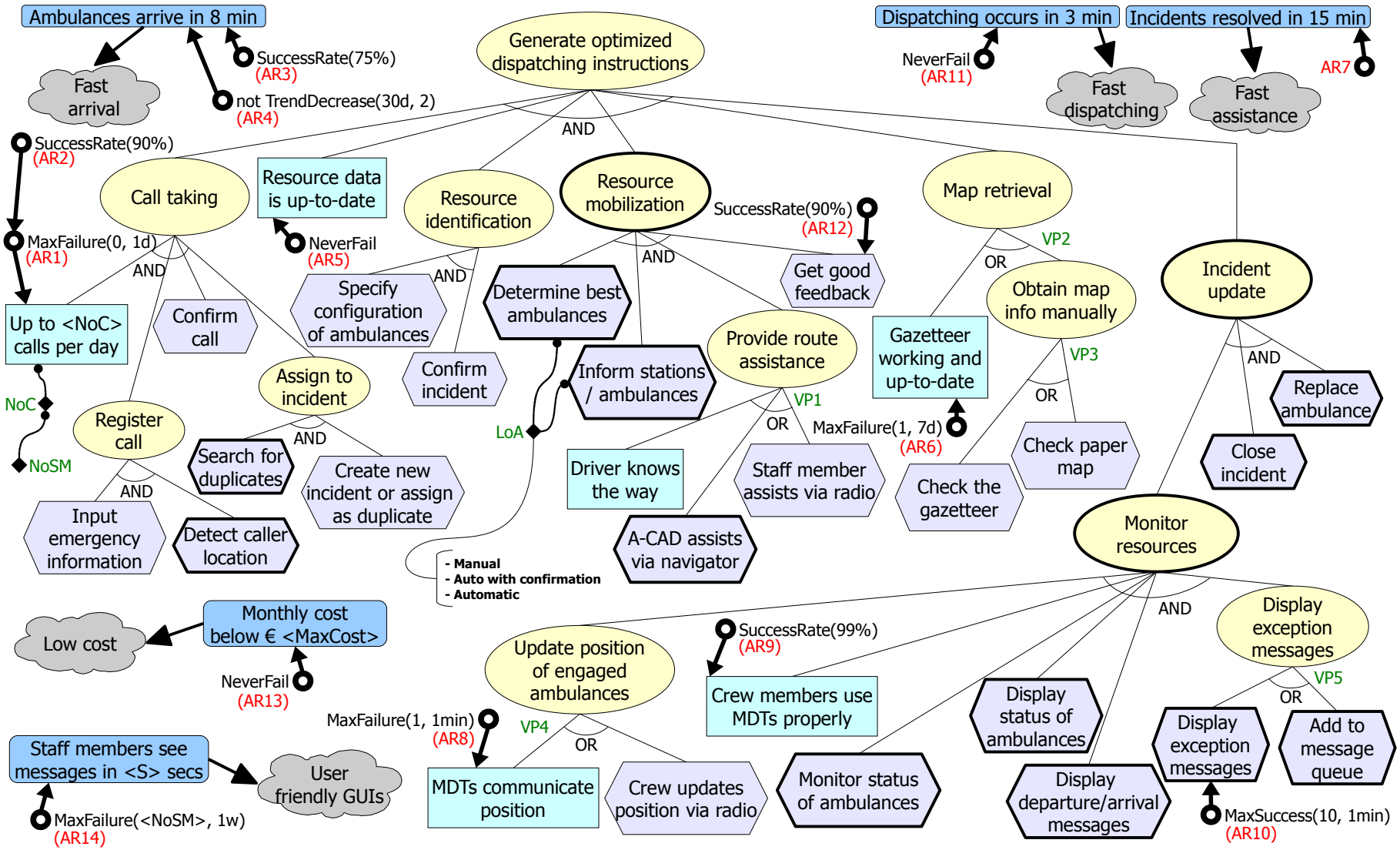
Reconfiguration

Class (from now on)

"If MDTs fail to communicate position every 13 seconds, relax it to every 20 seconds".

Evolution

残心 Step 2. System Identification



Parameters, indicators and relations

Basic

QC *Dispatching occurs in 3 min* should never fail;

AR 11

1. Consider *AwReqs* as indicators of requirements convergence;

2. Elicit system parameters that can be tuned at runtime;

- LoA: Level of Automation;
- VP1: goal *Provide route assistance*;
- VP2: goal Map retrieval;
- VP3: goal Obtain map info manually;
- VP4: goal Update position of engaged ambulances;
- MST: Minimum Search Time.

Parameters, indicators and relations

3. Elicit the (qualitative) effect that changes in parameters have in indicators;

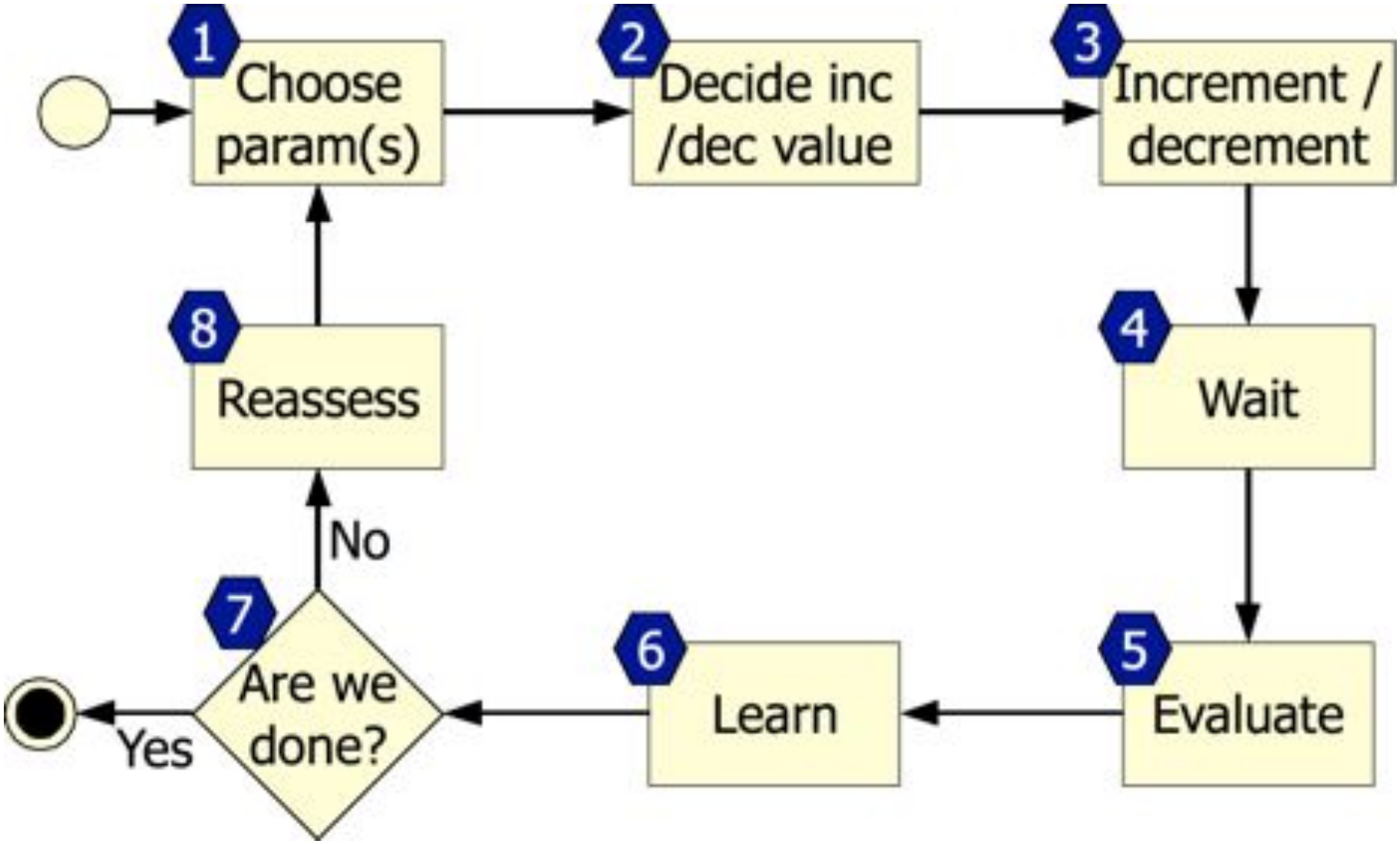
- $\Delta(AR11/LoA) > 0$
- $\Delta(AR11/VP1) < 0$
- $\Delta(AR11/VP2) < 0$
- $\Delta(AR11/VP3) < 0$
- $\Delta(AR11/VP4) < 0$
- $\Delta(AR11/MST)[0, 180] > 0$

$|\Delta(AR11/VP2)| >$
$|\Delta(AR11/LoA)| >$
$|\Delta(AR11/VP3)| >$
$|\Delta(AR11/VP1)| >$
$|\Delta(AR11/VP4)| >$
$|\Delta(AR11/MST)|$

4. Refine these relations, e.g., establishing an order of effect, if possible;

5. Choose a reconfiguration algorithm…

Qualia: customizable/extensible algorithm



残心 Step 3. Evolution Requirements

- Reconfiguration – solution space:
  - "If dispatching doesn't occur in 3 minutes, reduce the time spent searching for duplicates";
- Evolution – problem space:
  - "If the goal of registering a call is not achieved successfully, retry in 5 seconds or disable caller location detection";
  - "If the assumption that resource data is up-to-date is false, delegate the solution to a staff member";
  - "If MDTs fail to communicate position every 13 seconds, relax it to every 20 seconds".

Adaptation specification

AR	Adaptation strategies (EvoReqs)	Applicability
AR1	1. Warning("AS Management"); 2. Reconfigure(Ø).	1. Once per session; 2. Always.
AR5	1. Delegate("Staff Member").	1. Always
AR8	1. RelaxReplace(D_MDTPos_20s); 2. RelaxReplace(AR8, AR8_45s); 3. RelaxReplace(AR8_45s, AR8_30s); 4. Retry(6000); 5. Reconfigure(Ø [Immediate resol.]).	1. Once per session; 2. Once per session; 3. Once per session; 4. 3x per session; 5. Always.
AR11	1. Reconfigure({Oscillation Value Calculation, Oscillation Resolution Check}); 2. Reconfigure({Ordered Effect Parameter Choice} [order = descending]).	1. Always; 2. Always.

Operationalized by an E C A algorithm.

Try it yourself!

https://github.com/sefms-disi-unitn/Zanshin

Source code for the Zanshin framework, simulations console and the “Zanshin-managed” ATM software.

https://github.com/sefms-disi-unitn/Zanshin/wiki

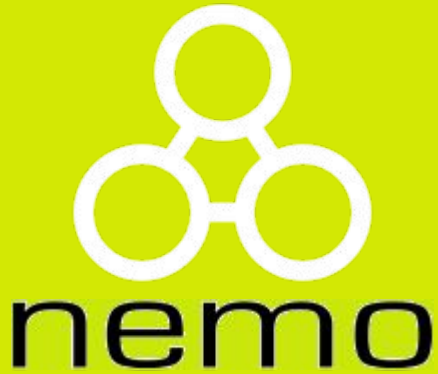
In the wiki, instructions on how to download Zanshin, how to run its simulations, how to create your own models and simulations and how to run the ATM experiment.

Current / future directions

- Further development of the prototype framework;
- Revision of its base model (ontology of requirements);
- LawReqs: studying the relation between regulatory compliance and adaptive systems [SEAMS'13c];
- Move towards architecture: model-driven approach;
- Handling the (many) limitations of Zanshin:
  - Too much responsibility on the system developer (integration) and analysts (consistency, correctness);
  - No support for legacy / 3rd party systems;
  - No integration with domain-specific models;
  - Qualia: empty procedures.

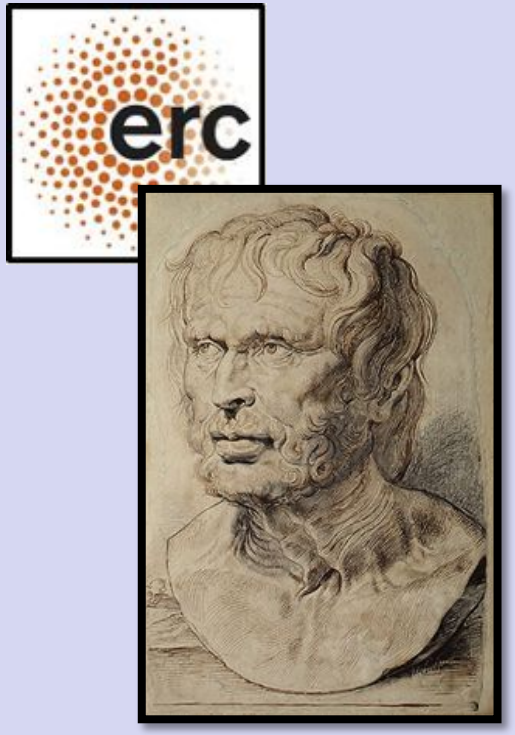
References

- **SEfSAS2**: V. E. S. Souza, A. Lapouchnian, W. N. Robinson, and J. Mylopoulos, "Awareness Requirements," in Software Engineering for Self-Adaptive Systems II, vol. 7475, R. Lemos, H. Giese, H. A. Müller, and M. Shaw, Eds. Springer, 2013, pp. 133–161.
- **ER'11**: V. E. S. Souza, A. Lapouchnian, and J. Mylopoulos, "System Identification for Adaptive Software Systems: a Requirements Engineering Perspective," in in Conceptual Modeling -- ER 2011, vol. 6998, M. Jeusfeld, L. Delcambre, and T.-W. Ling, Eds. Springer, 2011, pp. 346–361.
- **CSRD'12**: V. E. S. Souza, A. Lapouchnian, K. Angelopoulos, and J. Mylopoulos, "Requirements-driven software evolution," Computer Science - Research and Development, pp. 1–19, Springer, 2012.
- **CoopIS'12**: V. E. S. Souza, A. Lapouchnian, and J. Mylopoulos, "Requirements-Driven Qualitative Adaptation," in in On the Move to Meaningful Internet Systems: OTM 2012, vol. 7565, R. Meersman, H. Panetto, T. Dillon, S. Rinderle-Ma, P. Dadam, X. Zhou, S. Pearson, A. Ferscha, S. Bergamaschi, and I. F. Cruz, Eds. Rome, Italy: Springer, 2012, pp. 342–361.
- **A-CAD Tech. Rep.**: V. E. S. Souza, "An Experiment on the Development of an Adaptive System based on the LAS-CAD," available online: http://www.inf.ufes.br/~vitorsouza/a-cad/. Trento, Italy, 2012.
- **PhD**: V. E. S. Souza, "Requirements-based Software System Adaptation," PhD Thesis, University of Trento, Italy, 2012.
- **SEAMS'13a**: Tallabaci and V. E. S. Souza, "Engineering Adaptation with Zanshin: an Experience Report," in Proc. of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, 2013, pp. 93–102.
- **SEAMS'13b**: K. Angelopoulos, V. E. S. Souza, and J. Pimentel, "Requirements and Architectural Approaches to Adaptive Software Systems: A Comparative Study," in Proc. of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, 2013, pp. 23–32.
- **SEAMS'13c:** S. Ingolfo and V. E. S. Souza, "Law and Adaptivity in Requirements Engineering," in Proc. of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, 2013, pp. 163–168.



http://nemo.inf.ufes.br/

Nemo is supported by Brazilian agency FAPES (www.fapes.es.gov.br) through the PRONEX grant #52272362 and others.



This work has been supported by the ERC advanced grant 267856 “Lucretius: Foundations for Software Evolution” (April 2011 – March 2016) – http://www.lucretius.eu.

“Vanilla” requirements

#	Requirement
REQ-1	The system shall allow staff to register calls they receive from citizens;
REQ-2	The system shall, whenever possible, detect the location of the caller and associate it with the call registry (public phones have associated locations, cell phones might be triangulated, etc.);
REQ-3	The system shall allow staff to dismiss calls as non-emergencies;
REQ-4	The system shall assist staff in identifying, through the information from the call, if it refers to an open incident in the system;
REQ-5	The system shall allow staff to assign calls to open incidents as duplicates or create new incidents for calls;
	.
	.
	.

Initial goal model