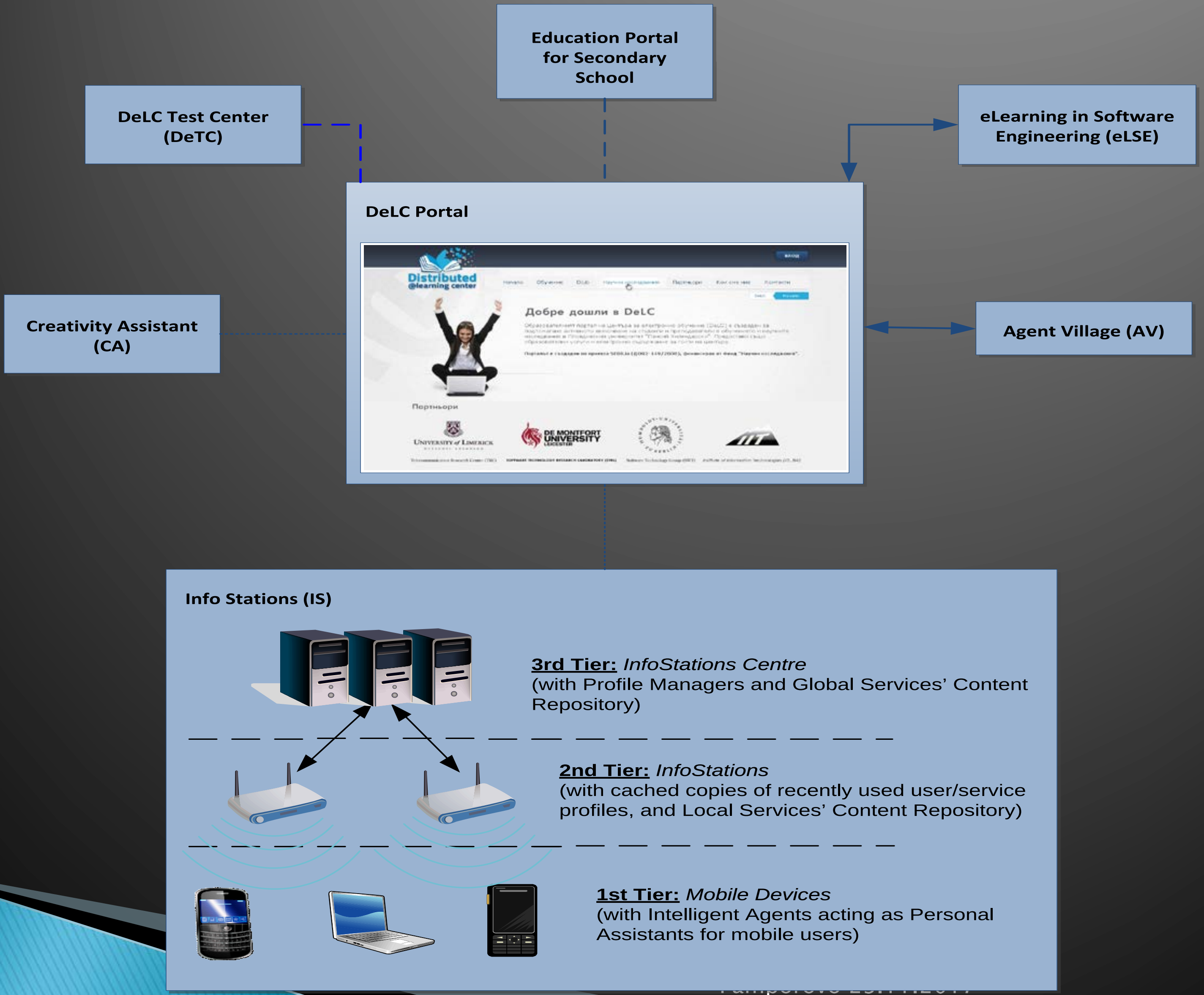
VES – Event management, models and mechanisms (Управление на събития във ВОП – модели и механизми)

Vladimir Valkanov
FMI, Plovdiv University

Motivation – DeLC Project

- Main goal: delivering e-services to support e-learning.
- DeLC architecture: distributed system, containing fixed and mobile nodes.
- Mobile node: providing mobile access to the services, through intelligent wireless network based on InfoStation architecture.

Infrastructure of DeLC



Education Portal for Secondary School

DeLC Test Center (DeTC)

eLearning in Software Engineering (eLSE)

DeLC Portal

Creativity Assistant (CA)

Agent Village (AV)

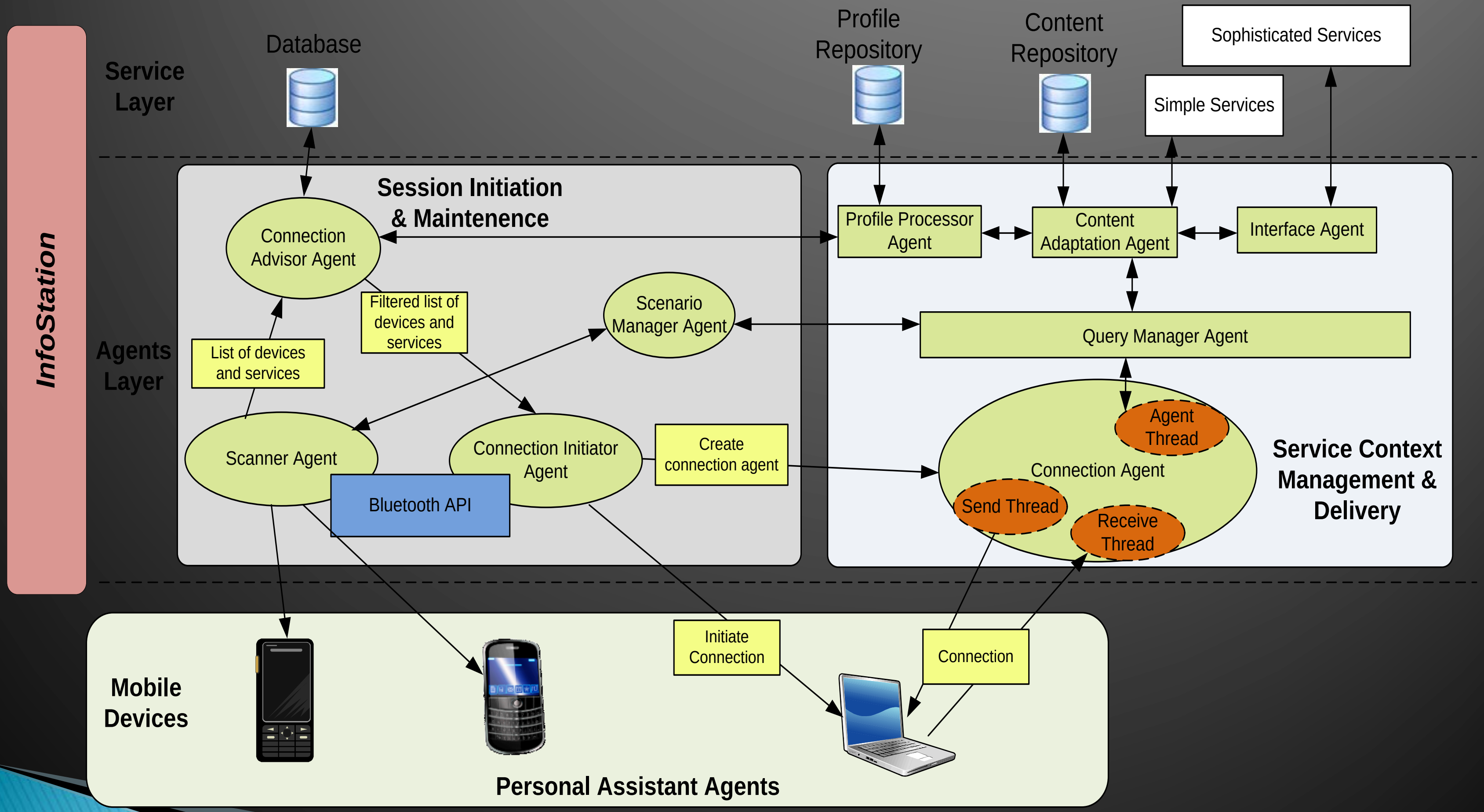
Info Stations (IS)

3rd Tier: *InfoStations Centre*
(with Profile Managers and Global Services' Content Repository)

2nd Tier: *InfoStations*
(with cached copies of recently used user/service profiles, and Local Services' Content Repository)

1st Tier: *Mobile Devices*
(with Intelligent Agents acting as Personal Assistants for mobile users)

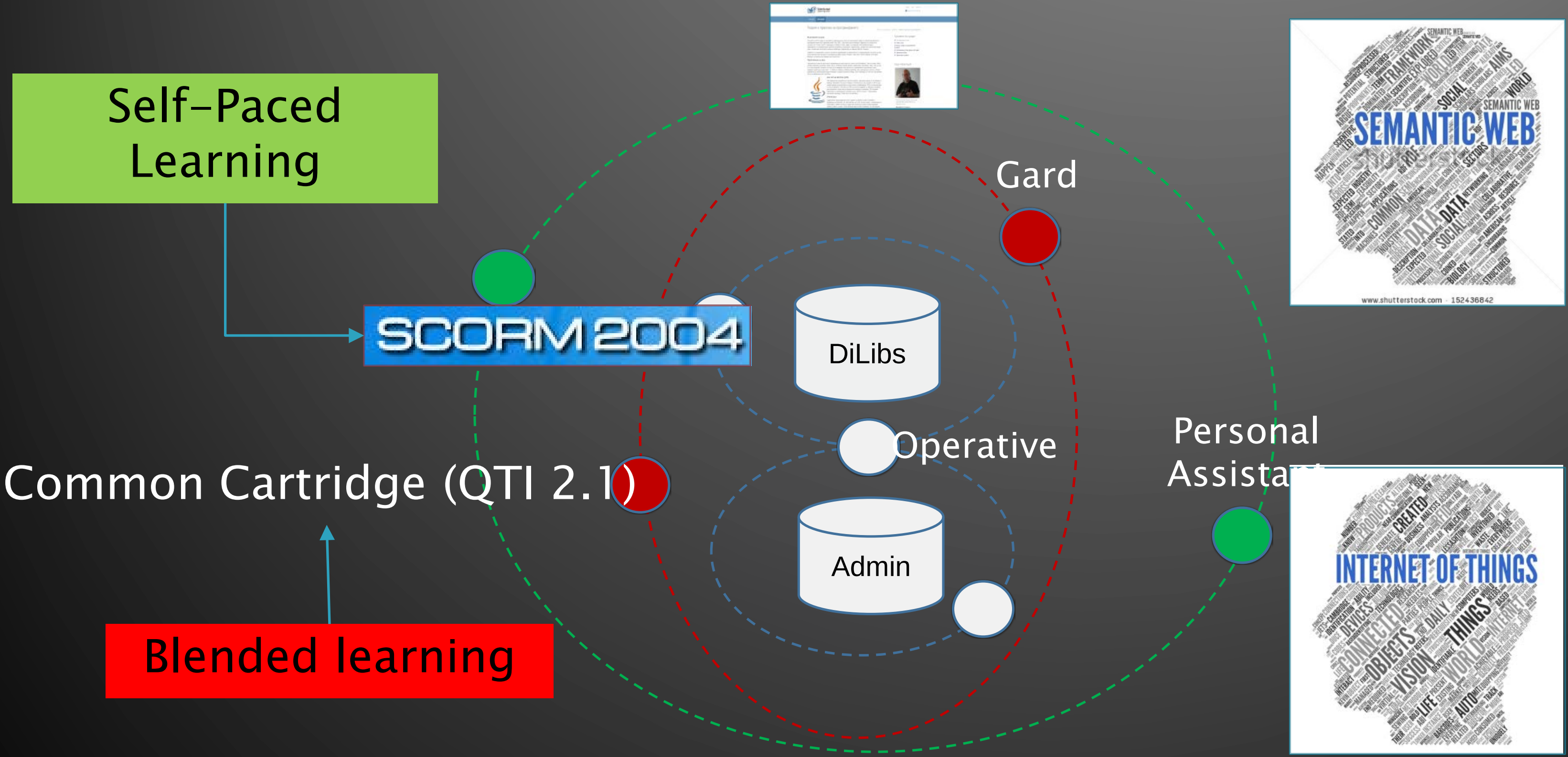
Mobile node middleware



From DeLC to VES

- New tendency in the development of Internet and Web:
  - Internet of Things – stimulating the origin of cyber-physical systems which will lead to essential consequences in the fallowing years.
  - Semantic web.
- Develop VES with the following features:
  - Intelligent.
  - Context-aware.
  - Scenario-oriented.
  - Controlled infrastructure.
- Lifelong learning support

Virtual Education Space



Self-Paced Learning

SCORM 2004

DiLibs

Gard

Operative

Admin

Personal Assista

Common Cartridge (QTI 2.1)

Blended learning

VES in more details: Building a Virtual Education Space, WMSCI 2015, July 12 – 15 – Orlando, Florida, USA

Virtual eLearning Space specifics

- Functionality has to be supported by agents.
- VES has a context-aware architecture (adaptation, personalization).
- Functionality of VES is a non constant set of resources.
- There is a basic functionalities without with VES cannot exits.
- Specific components in VES have to react to the changes of the environment.

Main goal

- To expand the middleware with intelligent agents, which are able to detect and manage time aspects of delivering educational services and content in distributed InfoStation network.
- Preparing DeLC for being part of Virtual eLearning Space (VES).

Time aspects

- Scenarios
  - They specify the functionality of our communication environment (InfoStations)
- During the execution of a service different local events could happened :
  - Getting in/out of range of an IS.
  - Change communication protocol.
  - Change the mobile device.
- The existing middleware could react to various events, but it is unable to represent them in time order.
  - No management mechanism.
- In the scenario point of view the problem is to manage scenario change and execution.

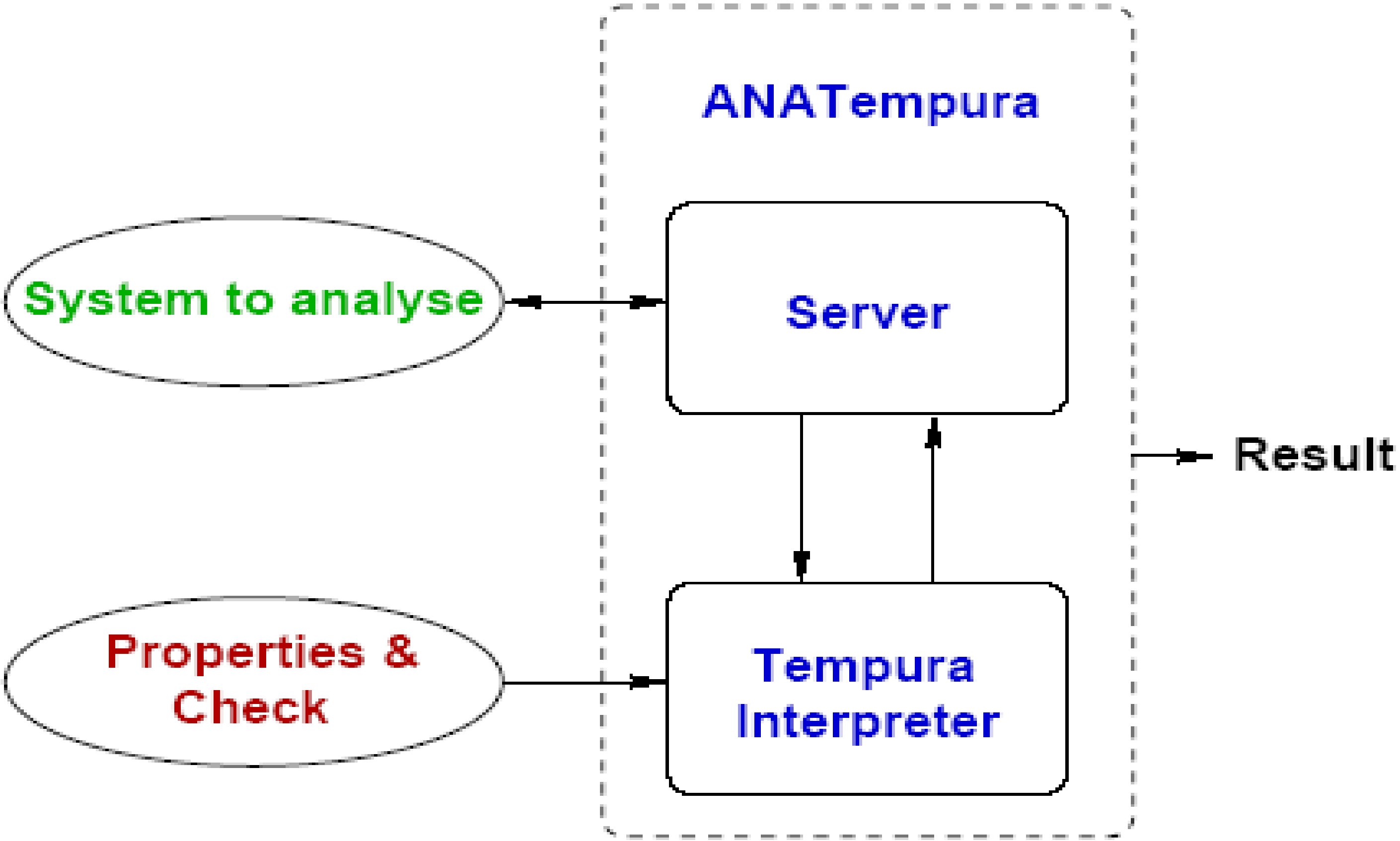
Choosing a proper formalism

- Interval Temporal Logic (ITL)
  - Ben Moszkowski – Computer Laboratory, University of Cambridge
- What is ITL:
  - First order logic with added time dependent operators like “sometimes” , “always”, “next” …
  - Considering time as a discrete sequence of points in time called intervals.
  - For ITL there is an interpreting mechanism and its program realisation called Tempura.

Tempura

- Imperative programming language which use subset of ITL:
  - First interpreter was written in Prolog
  - C/C++ version:
    - Roger Hale, Ph.D. thesis in Cambridge , 1984-1985 г.,
  - Maintenance: Antonio Cau, STRL, De Montfort University.
- AnaTempura
  - The centralize surrounding environment of Tempura.

AnaTempura


ANATempura
System to analyse
Server
Result
Properties & Check
Tempura Interpreter

Approach

- Three possible ways:
  - Wrapping Tempura with I/O Java classes.
  - Creating a complete new Java version of ITL interpreter.
  - Reengineering the existing C-based version of Tempura.

Why reengineering?

- Missing documentation and specification of the basic algorithms used in the interpreter.
- Homogeneous environment.
- Using proven system and already prepared test cases.

Reengineering in steps

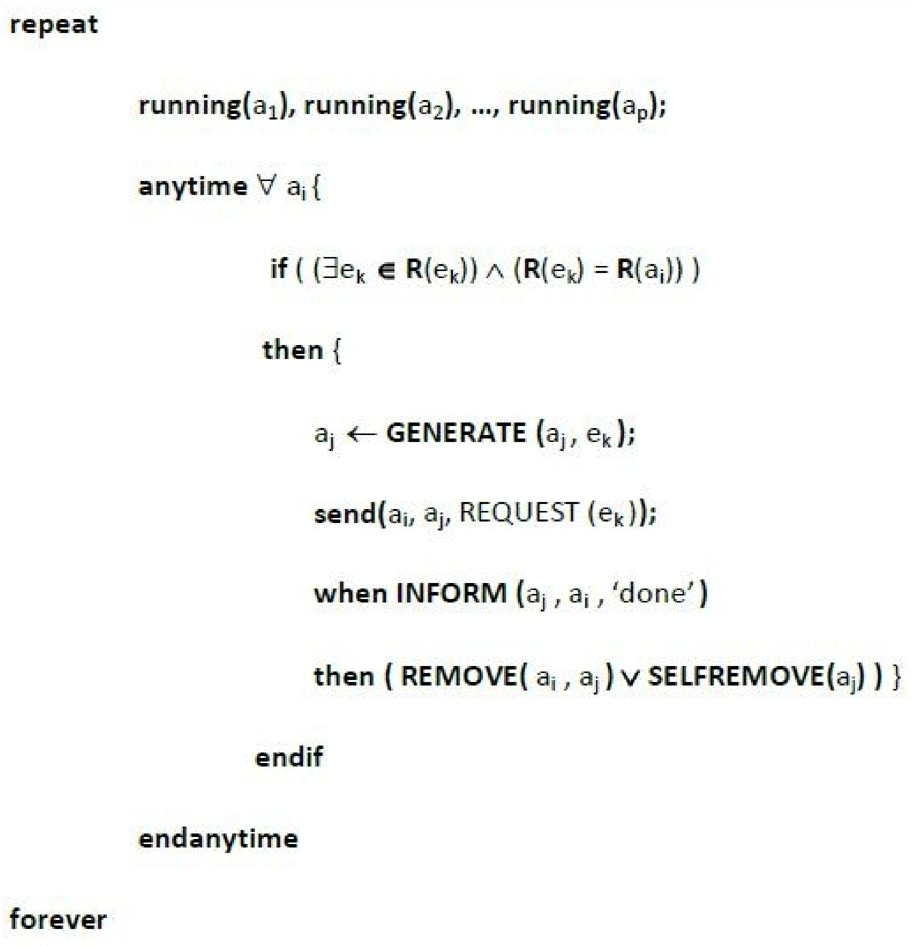
- Iterative hand-made translation
  - C to Java without changing the imperative structure of the system.
  - Imperative Java to OO Java (jTempura).
  - OO Java to AO Java (JADE based AjTempura) fallowing VES specification.

From jTempura to AjTempura

- To support the reengineering process we create a model:
  - C3A model
    - Abstract model for Context-Aware Agent Architecture
    - Functionality is supported by agents (persistent agents *PA* and operative agents *OA*).
    - PA supports the basic functionality of the system.
    - OA are generated by PA.

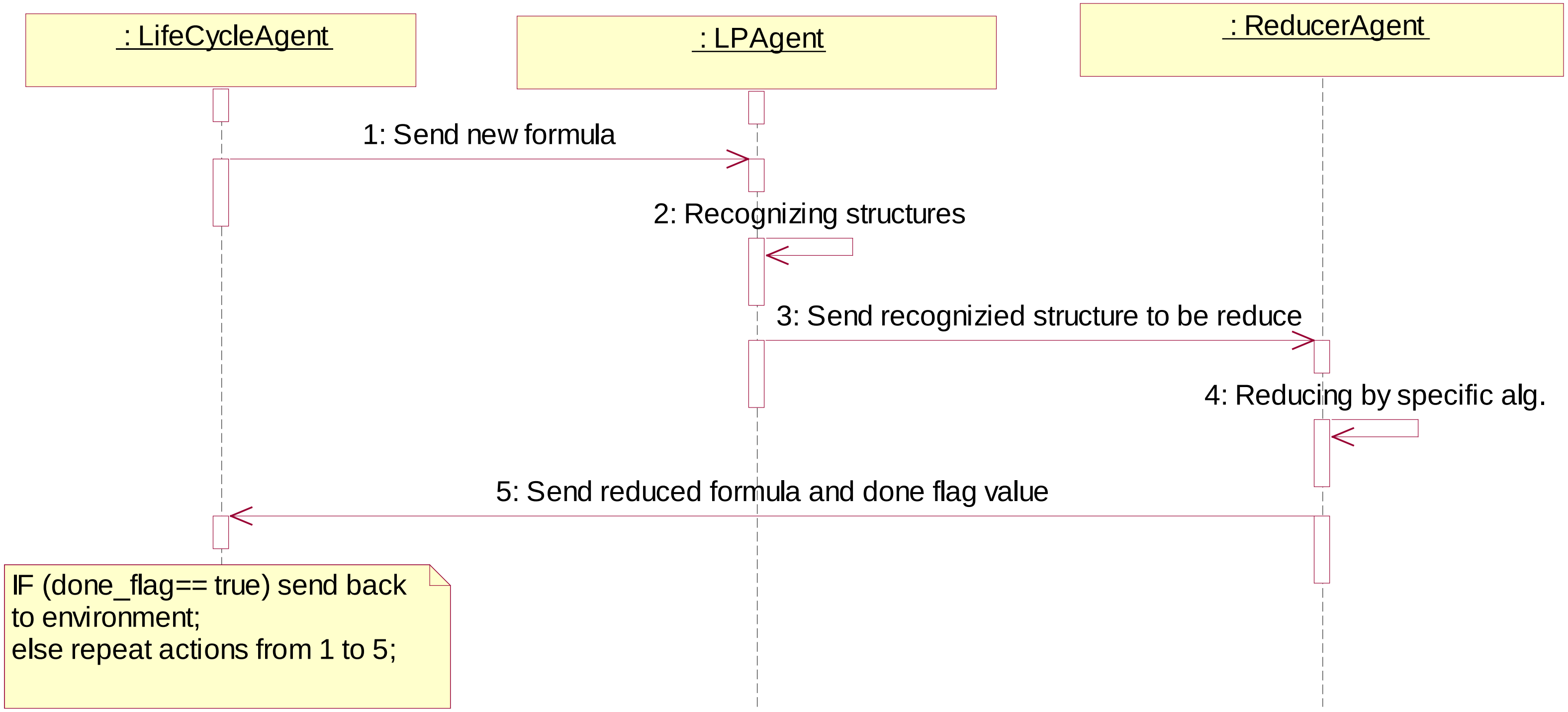
More details : AjTempura – First realization of C3A model, IEEE IS`14, Warsaw, Poland 2014

C3A Lifecycle

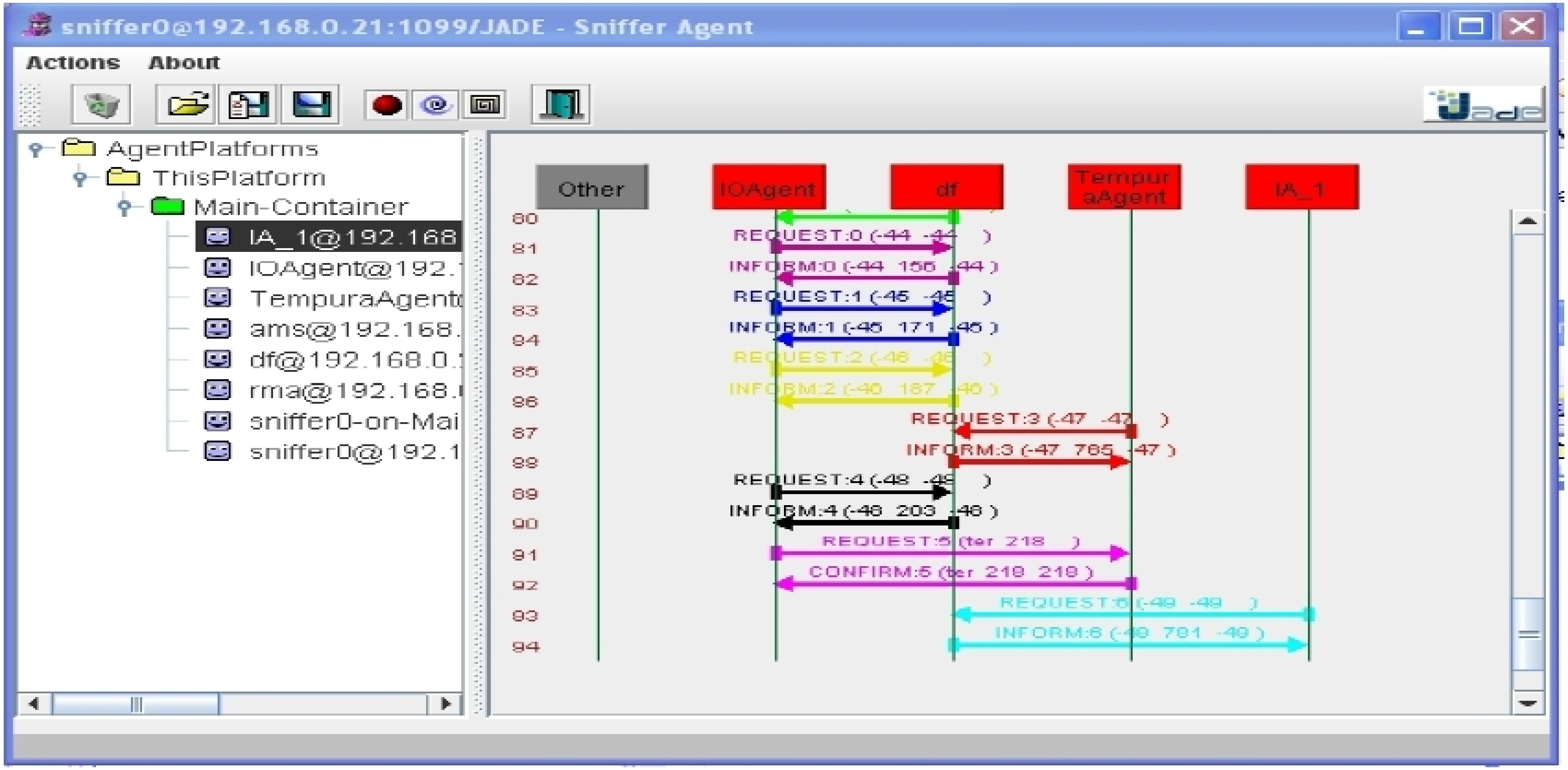


```
repeat
    running($a_1$), running($a_2$), …, running($a_p$);
    anytime $\forall a_i$ {
        if ( ($\exists e_k \in R(e_k)$) $\wedge$ ($R(e_k) = R(a_i)$) )
        then {
            $a_j \leftarrow$ GENERATE ($a_j$ , $e_k$);
            send($a_i$, $a_j$, REQUEST ($e_k$));
            when INFORM ($a_j$ , $a_i$ , 'done')
            then ( REMOVE( $a_i$ , $a_j$ ) $\vee$ SELFREMOVE($a_j$) ) }
        endif
    endanytime
forever
```

AjTempura life-cycle

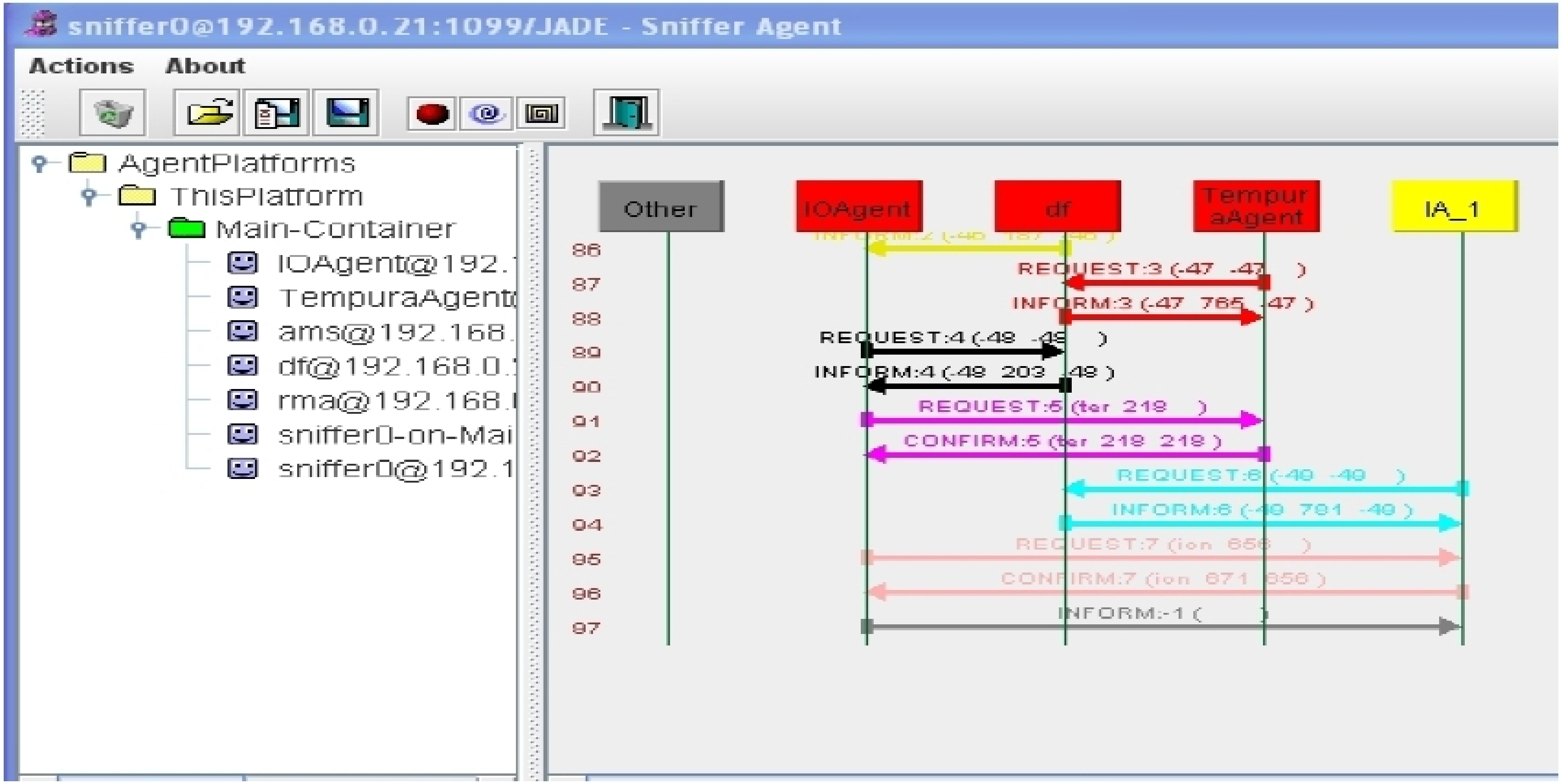

: LifeCycleAgent
: LPAgent
: ReducerAgent
1: Send new formula
2: Recognizing structures
3: Send recognizied structure to be reduce
4: Reducing by specific alg.
5: Send reduced formula and done flag value

IF (done_flag== true) send back to environment;
else repeat actions from 1 to 5;



AgentPlatforms
ThisPlatform
Main-Container
IA_1@192.168
IOAgent@192.1
TempuraAgent
ams@192.168.
df@192.168.0.1
rma@192.168.
sniffer0-on-Mai
sniffer0@192.1

```
80
81  REQUEST:0 (-44  -44   )
82  INFORM:0 (-44  156  -44 )
83  REQUEST:1 (-45  -45   )
84  INFORM:1 (-45  171  -45 )
85  REQUEST:2 (-46  -46   )
86  INFORM:2 (-46  187  -46 )
87  REQUEST:3 (-47  -47   )
88  INFORM:3 (-47  765  -47 )
89  REQUEST:4 (-48  -48   )
90  INFORM:4 (-48  203  -48 )
91  REQUEST:5 (ter  218   )
92  CONFIRM:5 (ter  218  218 )
93  REQUEST:6 (-49  -49   )
94  INFORM:6 (-49  781  -49 )
```


sniffer0@192.168.0.21:1099/JADE - Sniffer Agent
Actions
About
AgentPlatforms
ThisPlatform
Main-Container
IOAgent@192.1
TempuraAgent@
ams@192.168.
df@192.168.0.1
rma@192.168.I
sniffer0-on-Mai
sniffer0@192.1
Other
IOAgent
df
Tempur
aAgent
IA_1
INFORM:2 (-46 187 -46)
REQUEST:3 (-47 -47)
INFORM:3 (-47 765 -47)
REQUEST:4 (-48 -48)
INFORM:4 (-48 203 -48)
REQUEST:5 (ter 218)
CONFIRM:5 (ter 218 218)
REQUEST:6 (-49 -49)
INFORM:6 (-49 781 -49)
REQUEST:7 (ion 656)
CONFIRM:7 (ion 671 656)
INFORM:-1 ( )
86
87
88
89
90
91
92
93
94
95
96
97

THANK YOU FOR THE ATTENTION