Paplašināma daudzaģentu sistēmu modelēšanas vide

Ingars Ribners

Darba vadītājs: Guntis Arnicāns

2016.gada 13.aprīlis

Saturs

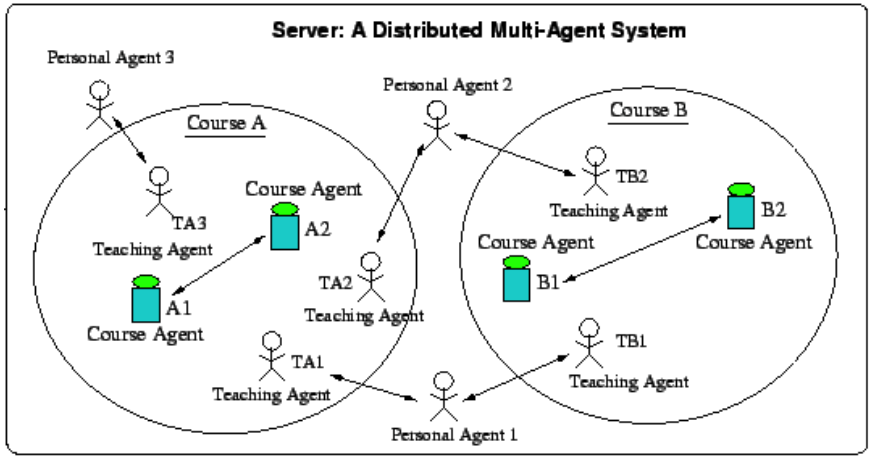
1. Darba tēma
2. Sistēmas konceptuālais modelis
3. Autonomā auto gatavošana GCDC-2016
4. Aģenta koriģētā arhitektūra

Darba tēma

1. Izstrādāt universālu un paplašināmu valodu, ar kuru var aprakstīt sistēmu modeļus **daudzaģentu paradigmā**;

2. Izstrādāt "izpildes" (simulācijas) vidi, kas spēj interpretēt aprakstītos modeļus un kuru būtu pietiekami vienkārši lietot ne-programmētājam, lai veidotu un izmantotu modeļus no gatavām komponentēm.

Daudzaģentu modelēšanas paradigma



Pētījumi par daudzaģentu sistēmām


1940-tie: John Von Neumann (1903-1957) – pašreplicējoša mašīna, Stanislaw Ulam (1909-1984) – šūnu automāti *(cellular automata)*

1969: John Conway (1937) – šīnu automātu speciālgadījums «dzīvības spēle» *(The Game of Life)*

1971: Thomas Schelling (1921) – rasu segregācijas modeļi

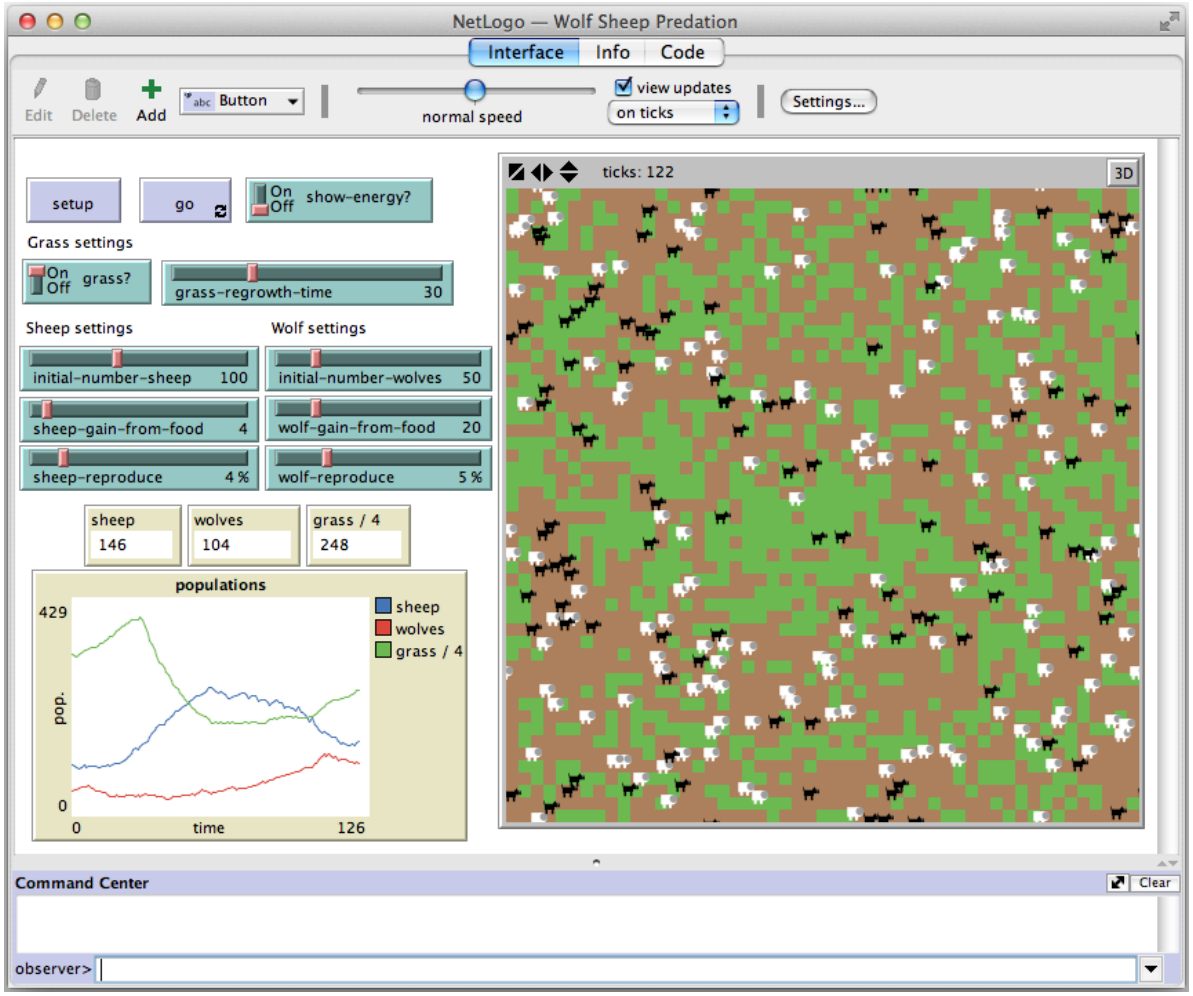
~1980: Robert Axelrod (1943) – sadarbības izcelšanās pētījumi (programmu sacensības «vairākkārtējā cietumnieku dilemmā» u.c.)

~1988: Craig Reynolds (1953) – bara (*flock*) izturēšanās algoritmi

uc.

Agent-Based Modeling Paradigm

System – a composition (structure) of autonomous entities – agents.



An example of typical agent based model (Netlogo)

Modeling paradigm inspired by nature

Ar daudzaģentu sistēmām saistītās pētījumu jomas

1. Agent-based modeling (computational model)
    - game theory,
    - complex systems,
    - emergence,
    - computational sociology,
    - evolutionary programming
    - Multi-agent systems (MAS)

2. MAS development environments (Netlogo,RePast,MASON Jason, JADE etc.)

3. MAS development methodologies (Gaia, RAP/AOR, O-MASE, AUML etc.)

4. Pieejas MAS semantikas aprakstīšanai (programmēšanas valodām):
    - Actor model – Carl Hewitt
    - π-calculus – Robin Milner
    - Communicating Sequential Processes (CSP) – Tony Hoare
    - Petri tīkli – Carl Adam Petri

Modelēšanas vides īpašības

Uzmanība uz šādiem aspektiem:

1. Liels aģentu skaits modelī (10^4- 10^5 aģenti)
2. Spēja modelēt dzīvas sistēmas
3. Hierarhiska aģentu–vižu struktūra
4. Materiālas mijiedarbības modelēšana
5. Individuāla aģenta vēsture (evolūcija, ģenētiskie algoritmi)

6. Klienta-servera arhitektūra

Simulācijas prototips (Erlang)

(one *Erlang/OTP 17.5* node, *PC i5-4460, 4-cores, 3.2GHz, 16GB RAM, Windows 8)****

Number of agents	Start time (ms)*	Memory used (MB)**	Memory used by agent (kB)
100	17	20,00	35,00
500	22	25,40	17,80
1000	37	32,10	15,60
5000	194	85,60	13,82
10000	530	152,40	13,59
50000	8007	726,90	14,21
100000	28451	1402,50	13,86

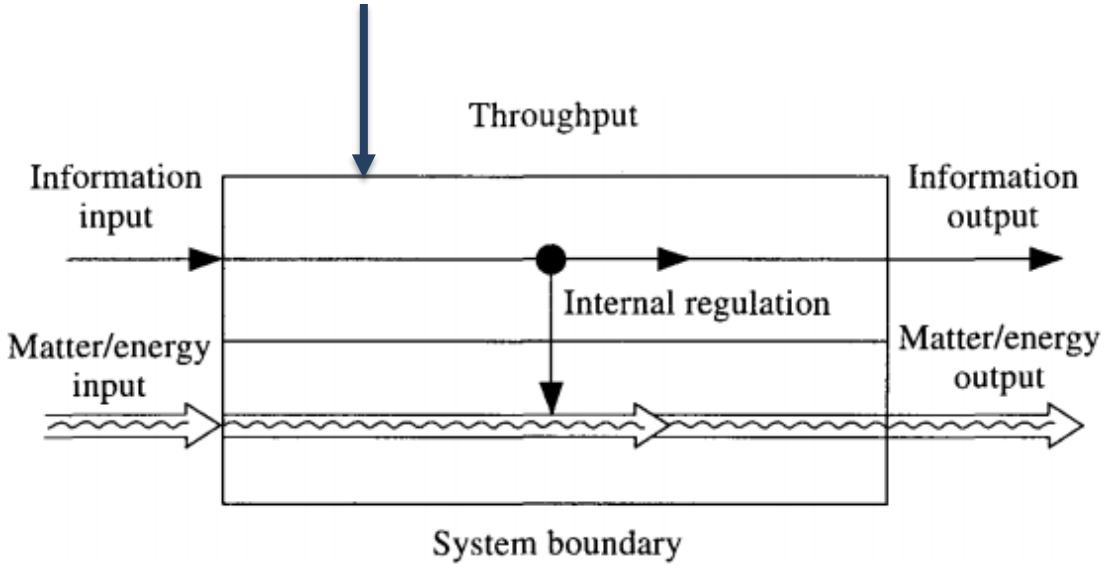
* Average from 4 measurements

** Approx. value from Windows 8 Task Manager

*** Erlang provides instant (or almost instant) scalability (for SMP, cluster, GPU)

Properties of Living Systems

Living System – a special subclass of Open System



James G. Miller, Living systems (1978,1991)

Properties of Living Systems

Eight hierarchical levels of living systems:

supranational systems.
societies,
communities,
organizations,
groups,
organisms,
organs,
cells,

James G. Miller, Living systems (1978,1991)

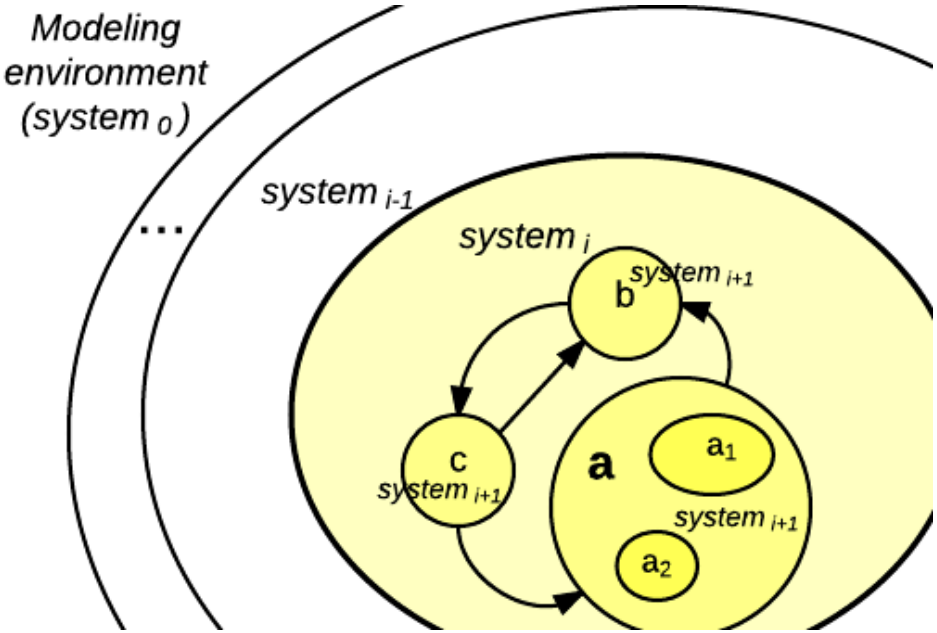
Properties of Living Systems

20 standard functional subsystems of a living system:

reproducer,
ingestor,
converter,
storage,
motor,
input transducer,
channel and net,
decoder,
memory,
encoder,
boundary,
distributor,
producer,
extruder,
supporter,
internal transducer,
timer,
associator,
decider,
output transducer.

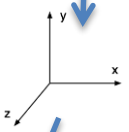
James G. Miller, Living systems (1978,1991)

Hierarhiska aģentu-vižu struktūra

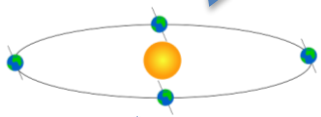


Hierarhijas piemērs

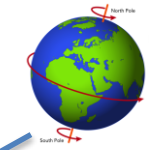
Time model



Space model



year cycle...

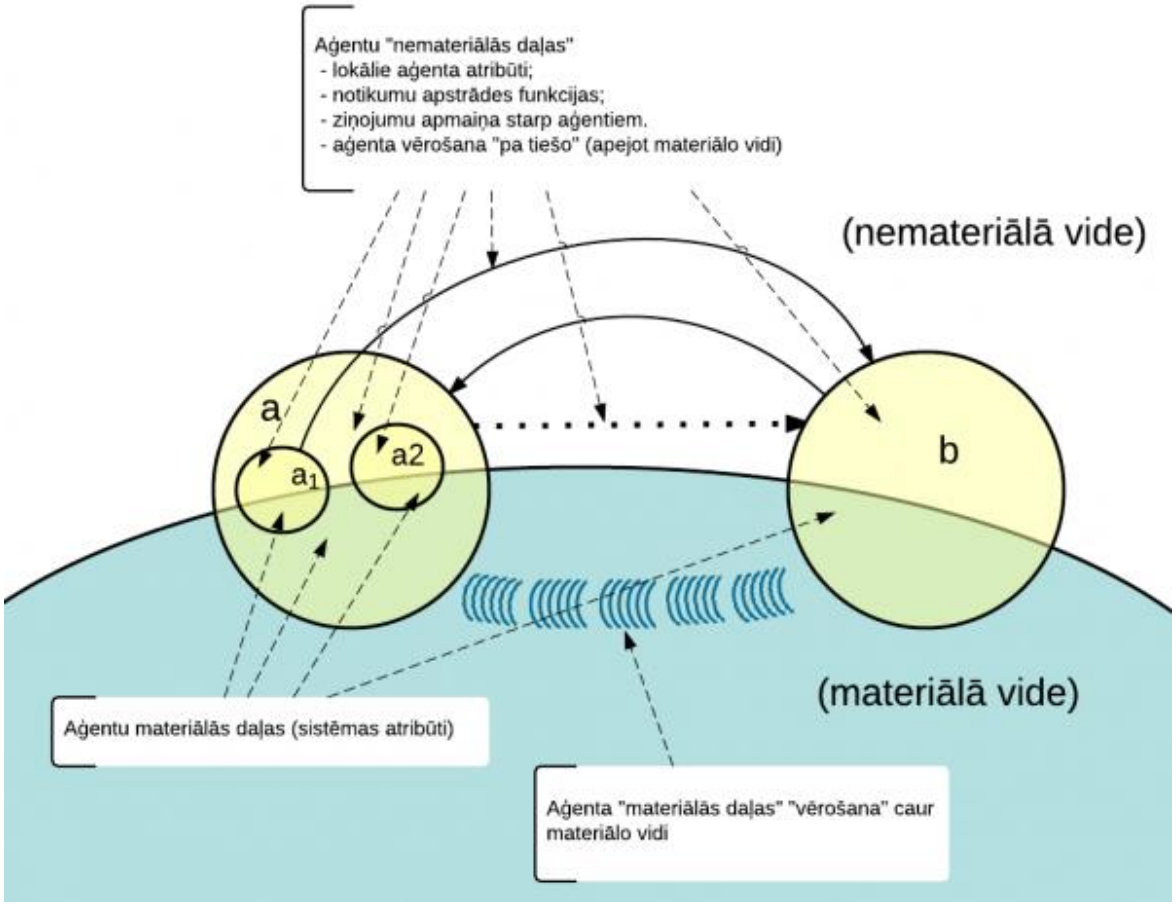


Day cycle, gravitation field...

Place, altitude...

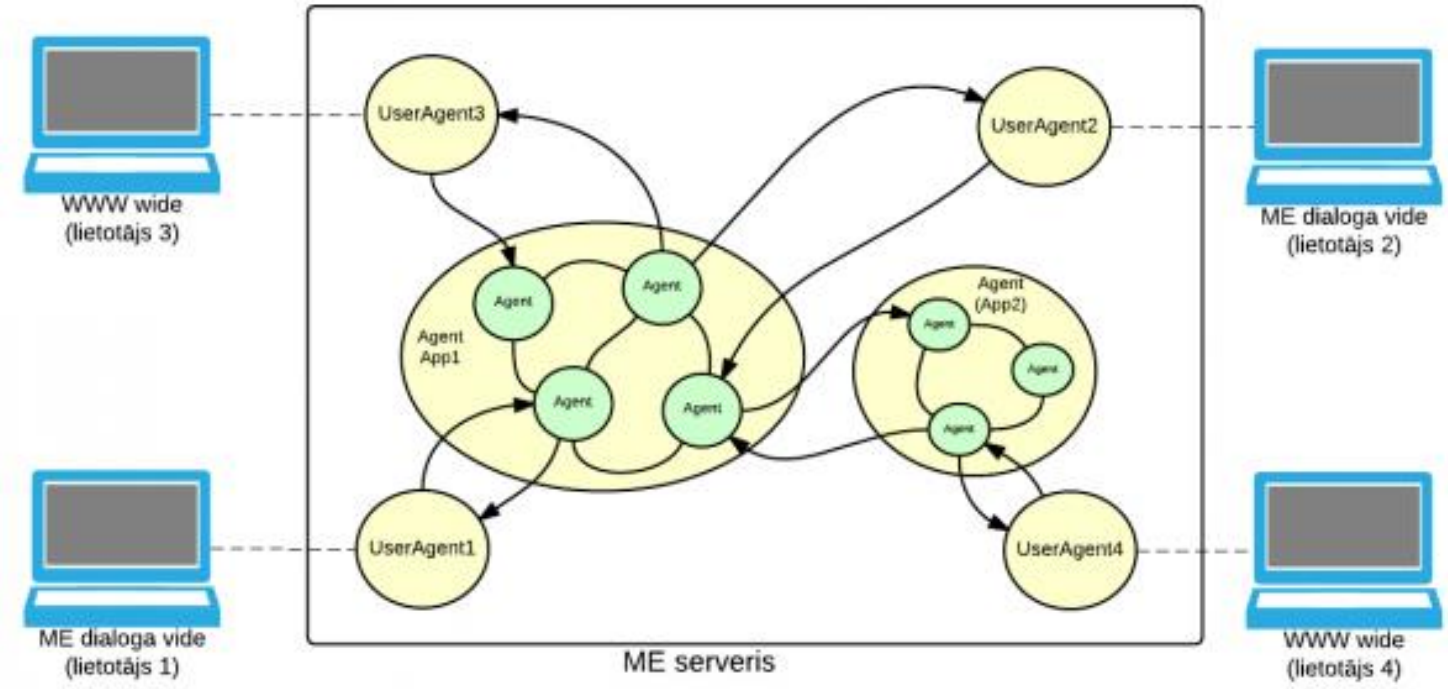
Temperature, gravitation acceleration, light, ... and their change pattern (t)

Materiāla un nemateriāla aģentu mijiedarbība



Aģenta materiālā un nemateriālā daļa / mijiedarbības veidi

Klienta-servera arhitektūra



Main Concepts

- Agent
- Event
- Environment
- Communication arena
- Artifact

Agent

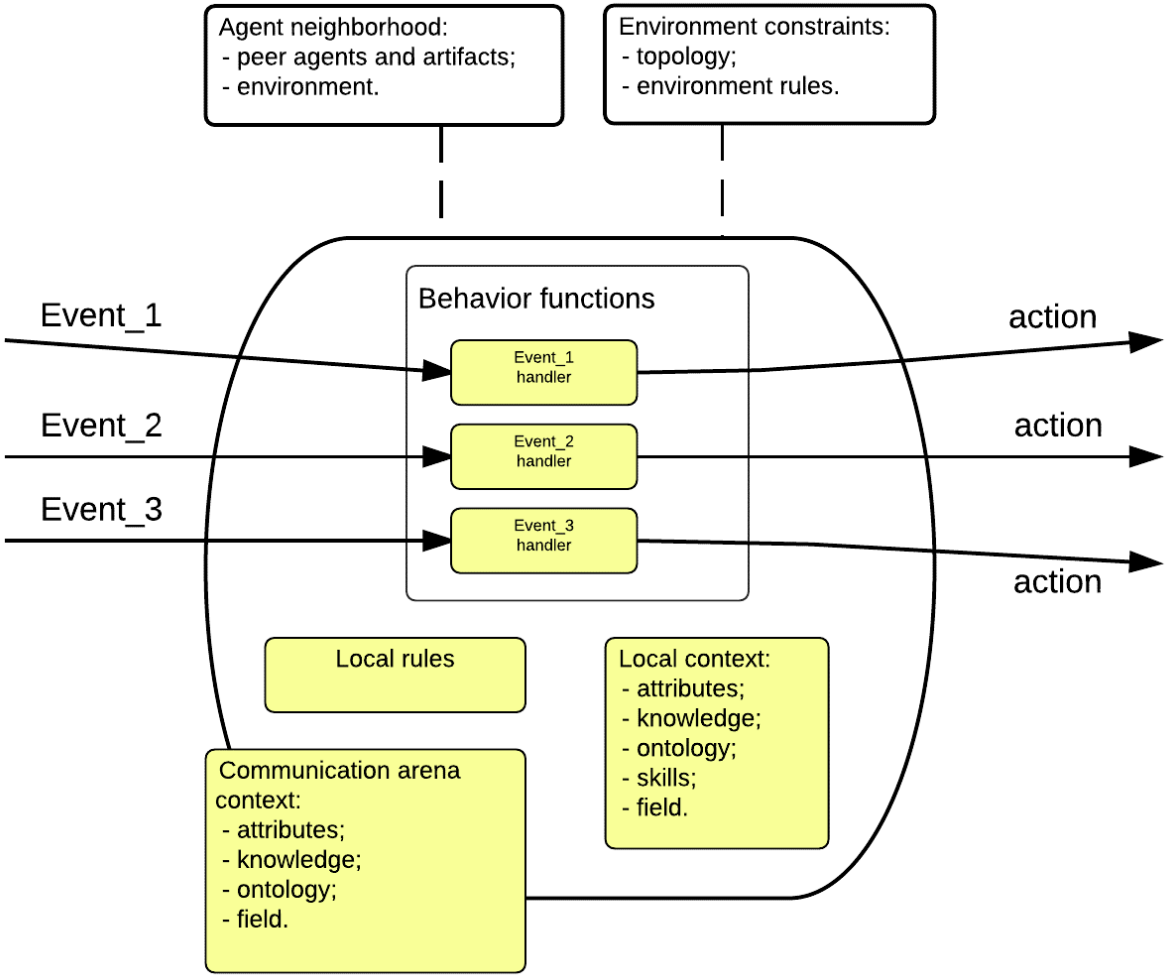
Agent – a key abstraction in this modeling approach.

Agent means a *subject* that acts in some environment.

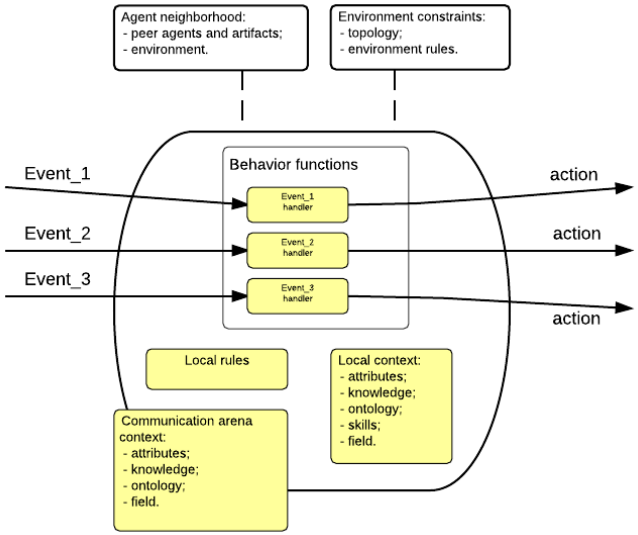
Main qualities of **agent** as described by *Jennings et.al (1998)*:

- Situatedness;
- Autonomy;
- Flexibility (responsiveness, pro-activeness, social behaviour)

Aģenta struktūra (modelis iter#1)



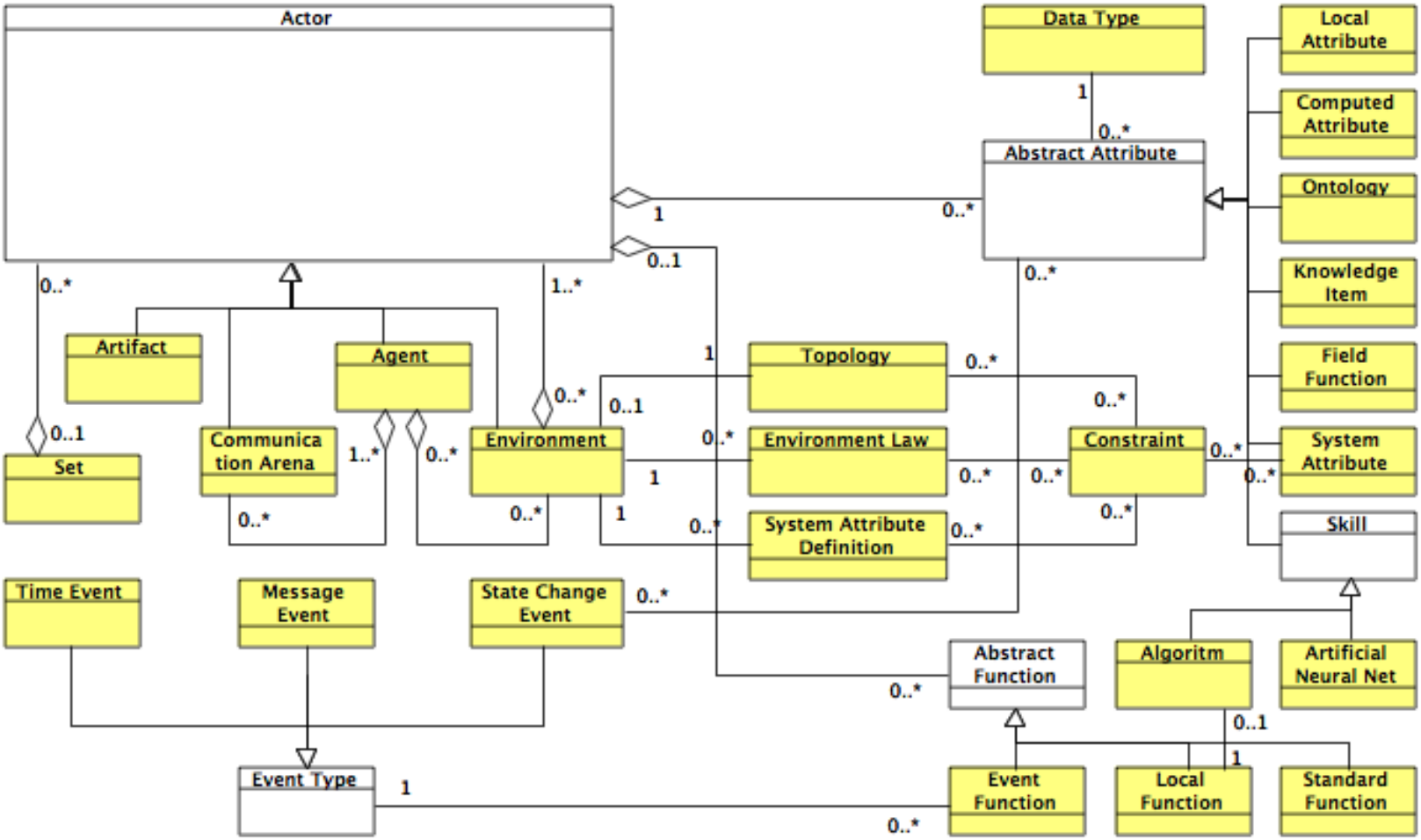
Piemērs: Auto



a.name = “Car”
a.lights = off
a.masa = 1.2 (kg)

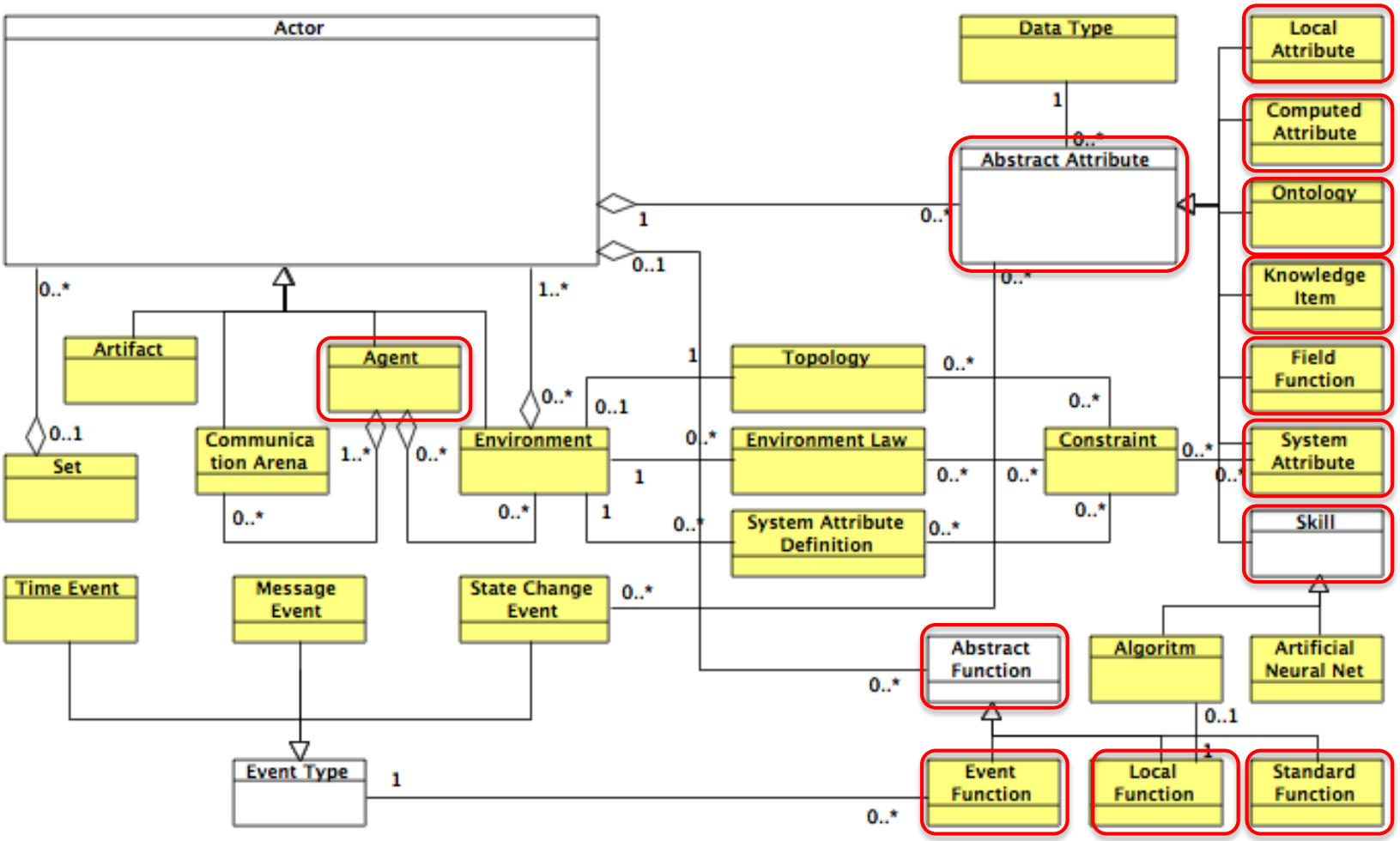
Main Concepts

Agent, Event, Environment, Communication arena, Artifact



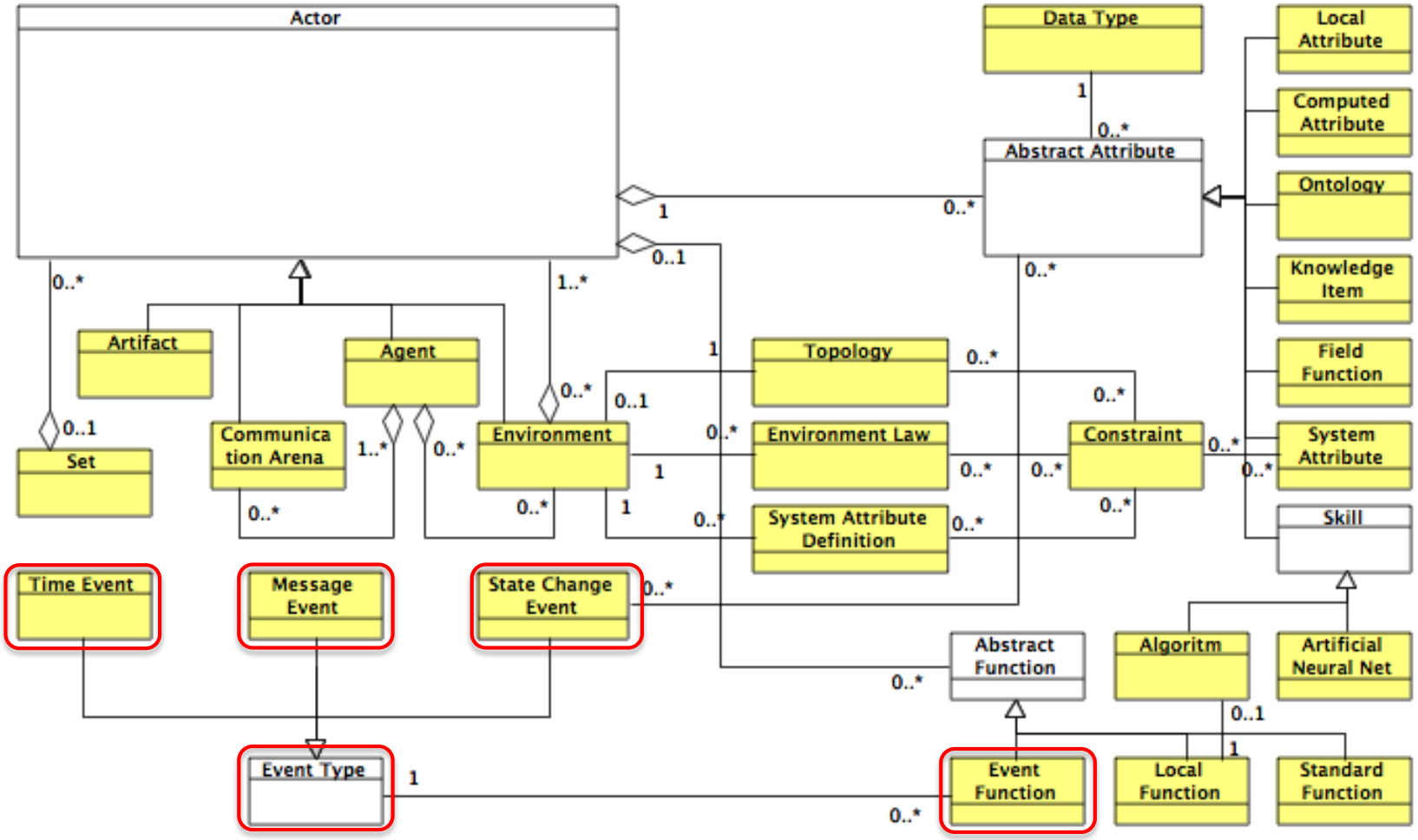
Main Concepts

Agent, Event, Environment, Communication arena, Artifact



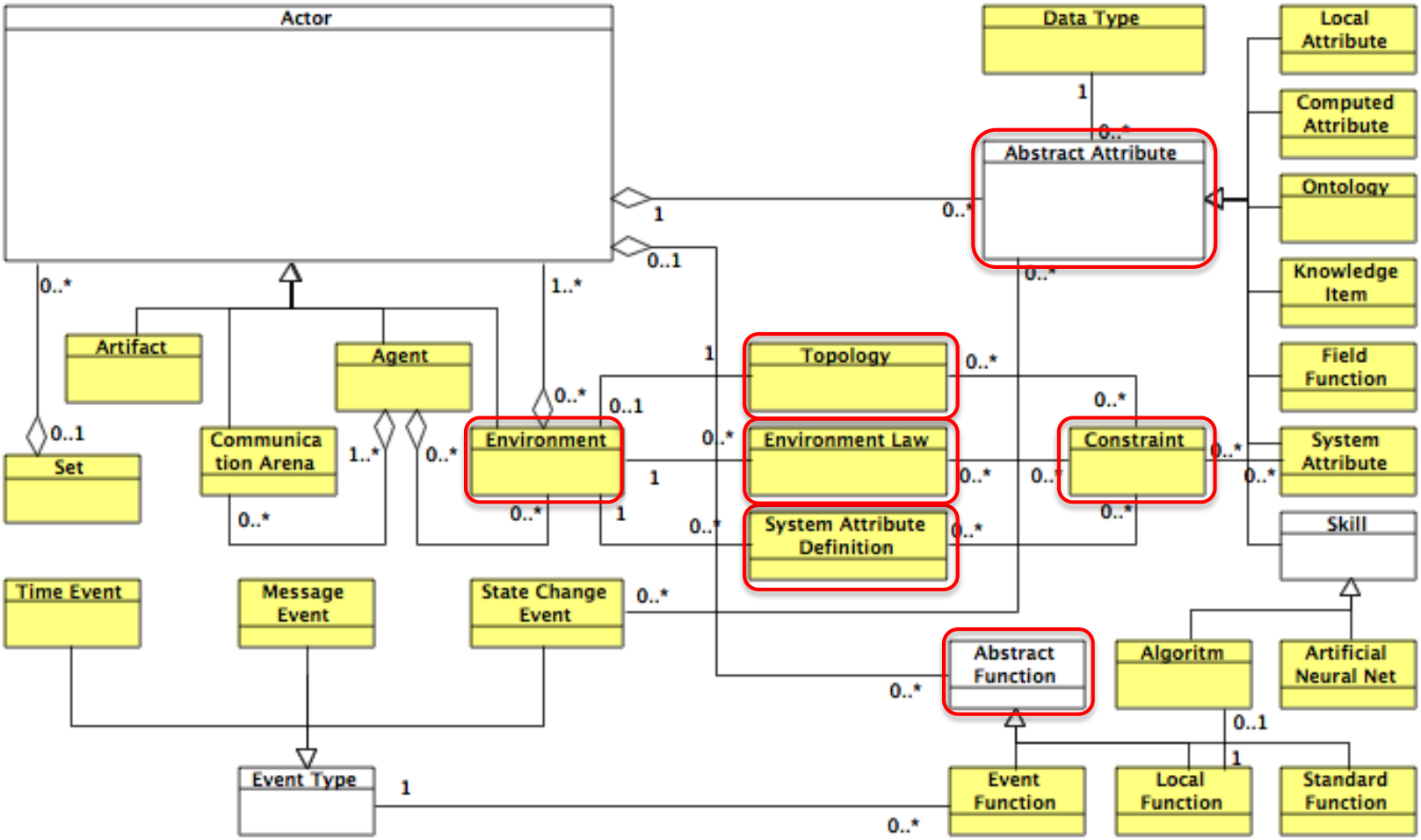
Main Concepts

Agent, **Event**, Environment, Communication arena, Artifact



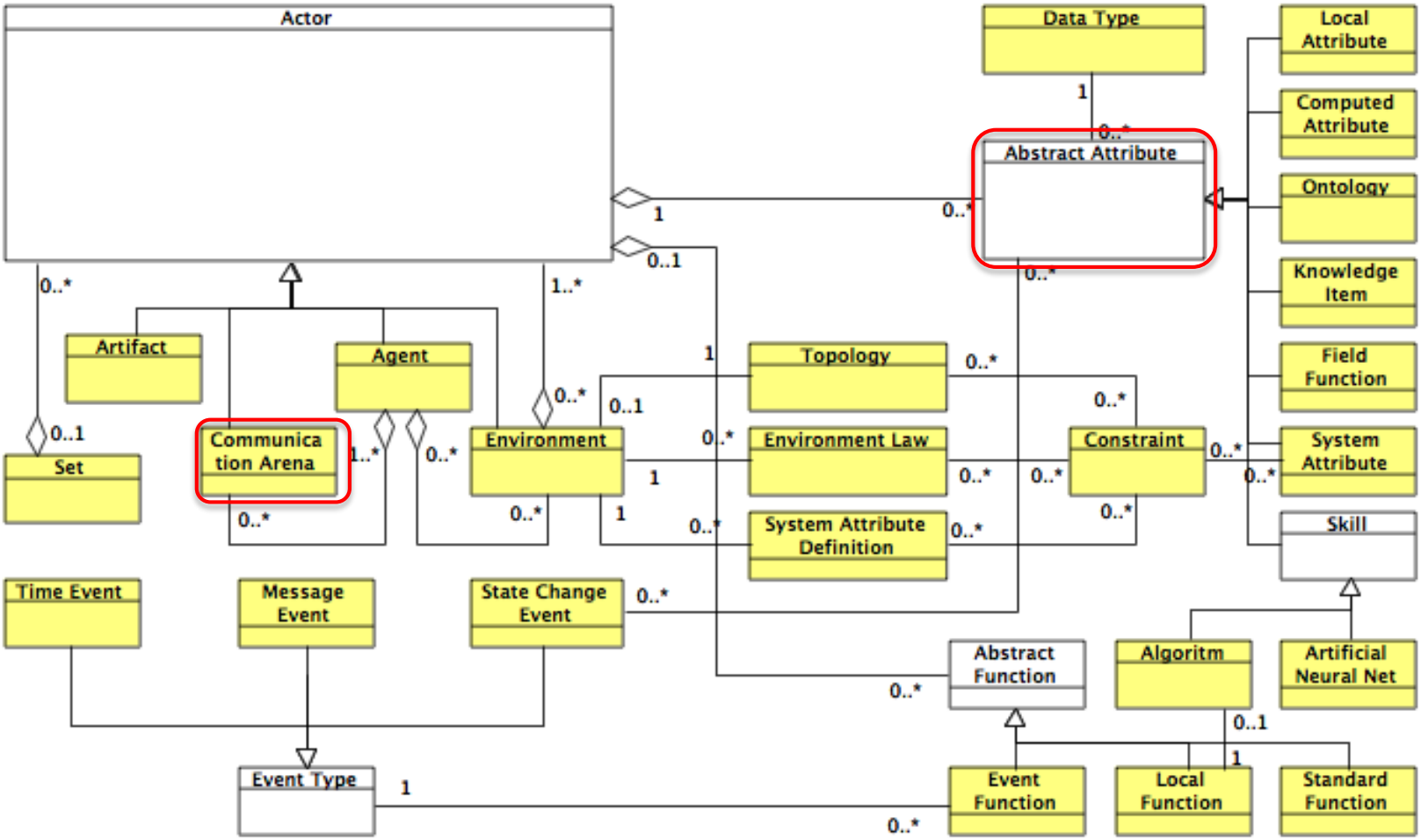
Main Concepts

Agent, Event, **Environment**, Communication arena, Artifact



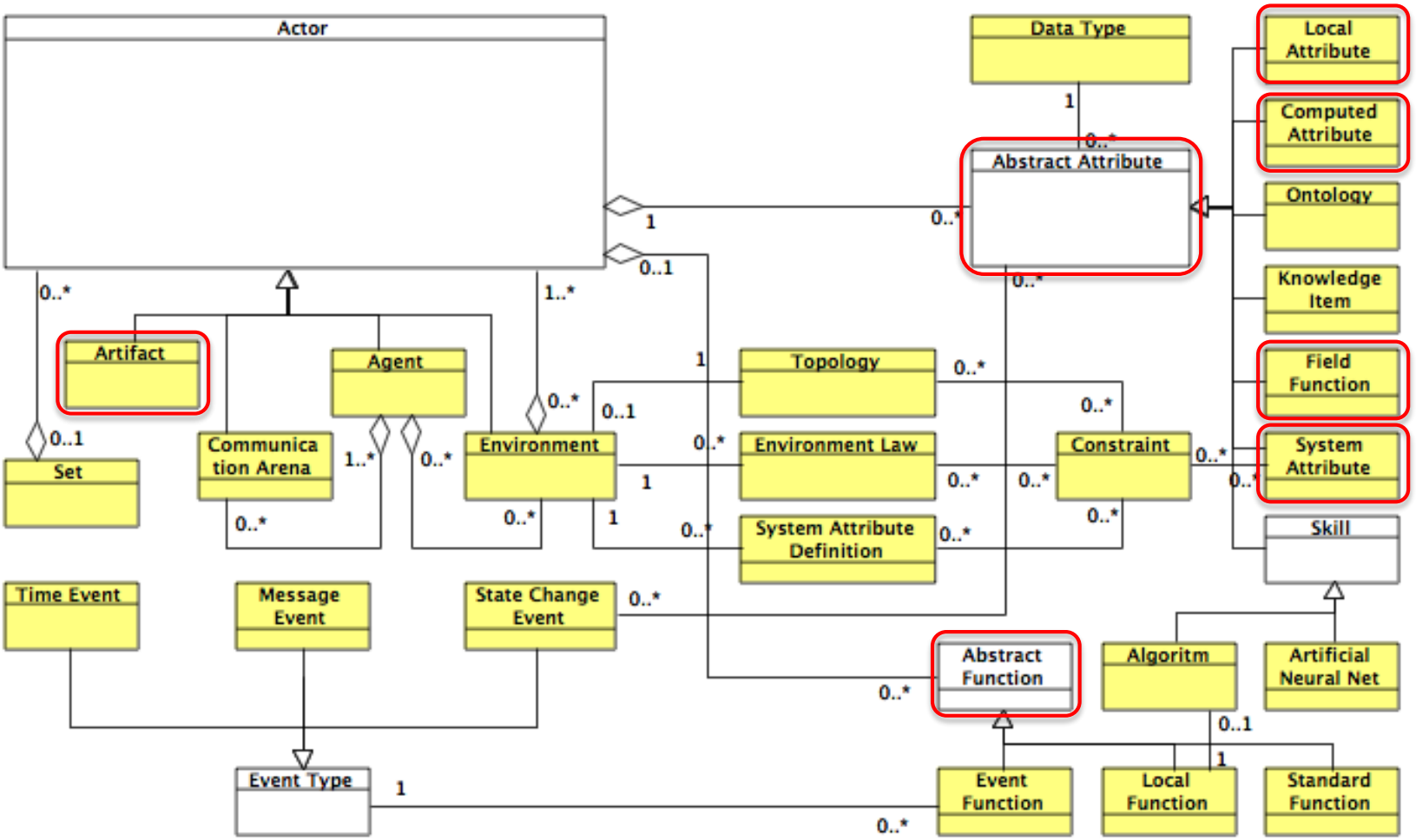
Main Concepts

Agent, Event, Environment, **Communication arena**, Artifact



Main Concepts

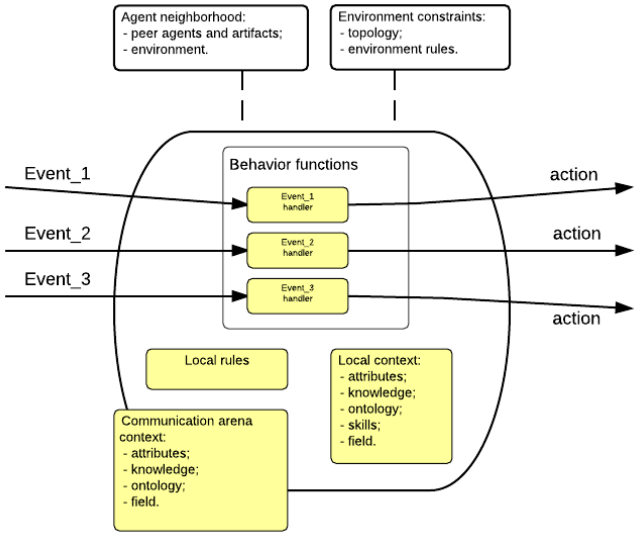
Agent, Event, Environment, Communication arena, **Artifact**



Publikācijas

05.2015 I.Ribners, G.Arnicans.
Concept of Client-Server Environment for Agent-Based Modeling and Simulation of Living Systems
(CICSyN - 2015, Rīga)

Materiālā aģenta mijiedarbība



a.name = “Car”
a.lights = off
a.masa = 1.2 (kg)

a.speed = ? // sistēmas attribūts
a.{x,y} = ? // sistēmas attribūts

Papildināt esošo aģenta arhitektūru, lai tā derētu arī materiālās aģenta mijiedarbības modelēšanai.

Uzdevums

Jāsaprot, kā pareizi vadīt materiālus objektus un šo modeli jāizmanto aģenta arhitektūrā.

Jomas:

- Vadības teorija (Control Theory)
- Sistēmas ar atgriezenisko saiti (Feedback Systems)
- Kiberfizikālās sistēmas (KFS)
- Roboti
- Kibernētika
- uc.

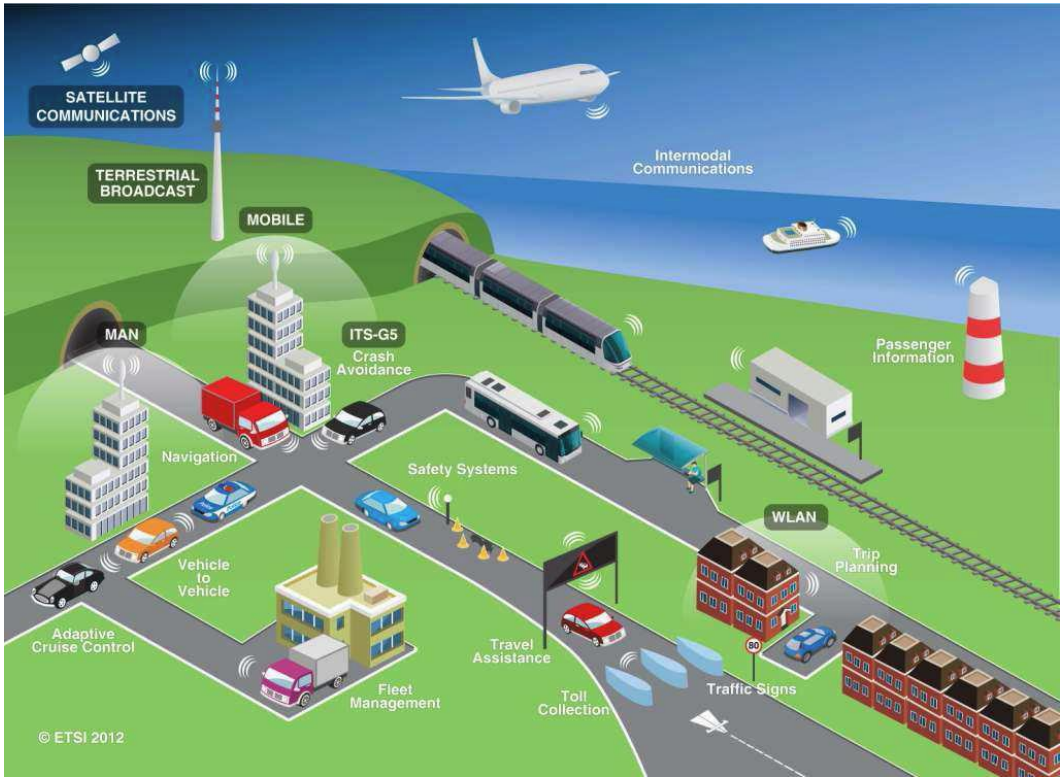
22-31 May, 2016

Helmond (Netherlands)

http://www.gcdc.net/

The project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 612035.

Inteligent transport Systems (ITS)



GCDC – focus no cooperative ITS (C-ITS)

ETSI ITS-G5 network stack

- V2V
- V2I

History (GCDC 2011)

GCDC-2016 scenarios

1. Road work

2. Intersection

3. Operative vehicle

1.

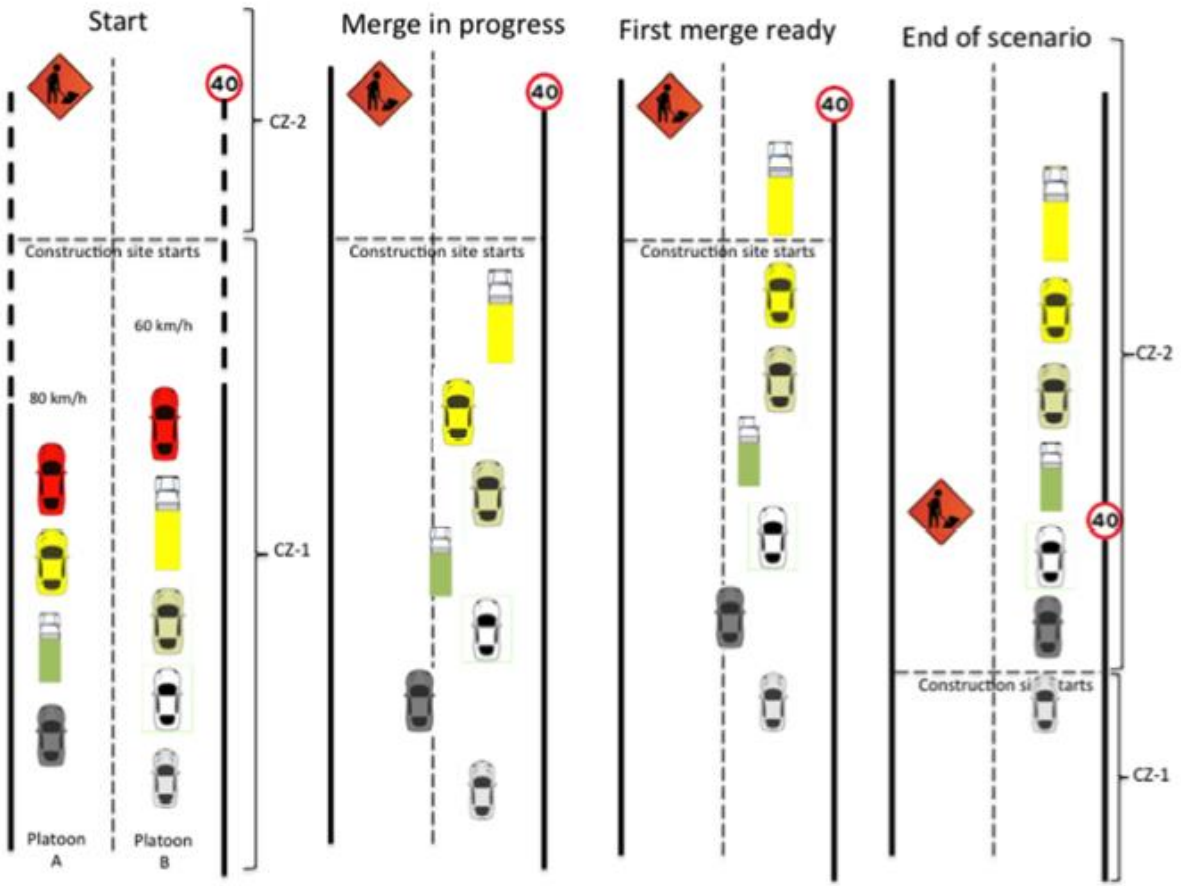


Figure 1: Scenario 1 description

2.

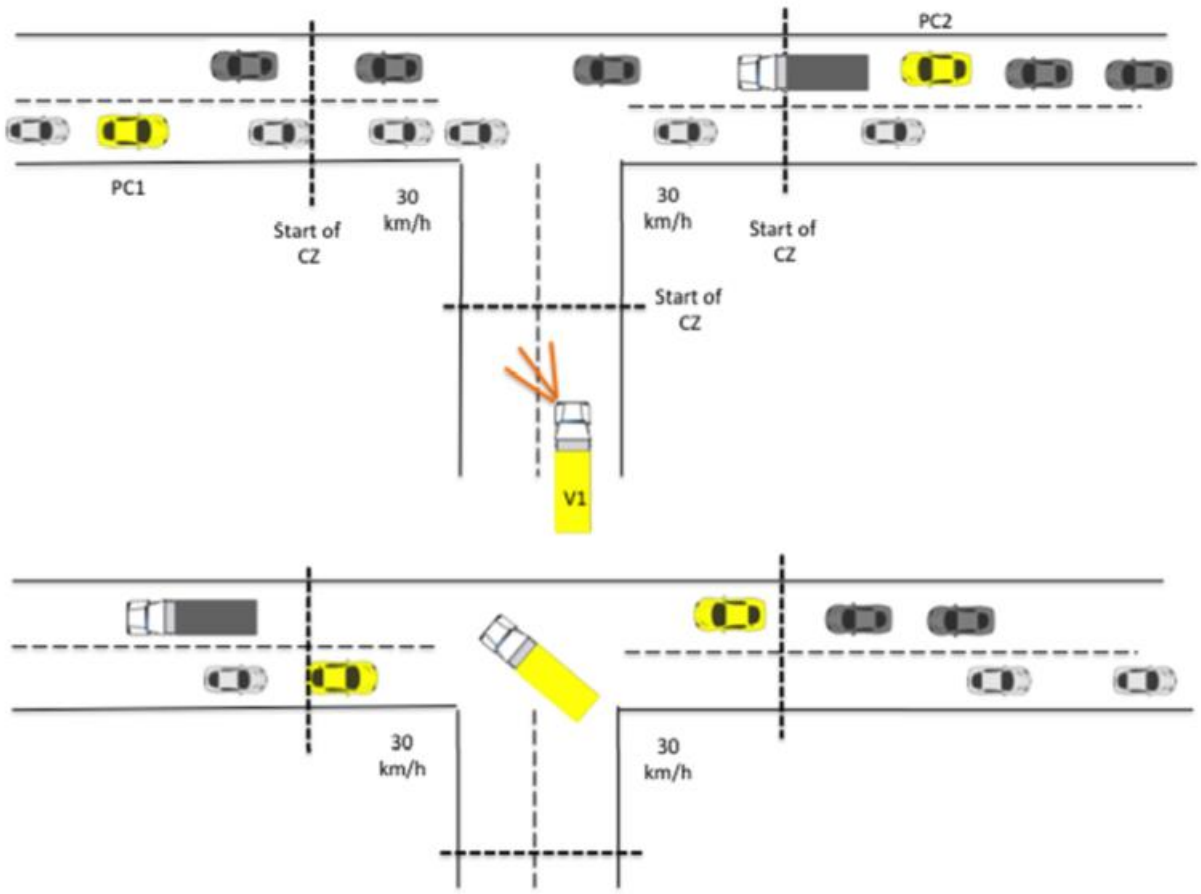


Figure 2: Scenario 2 description

3.

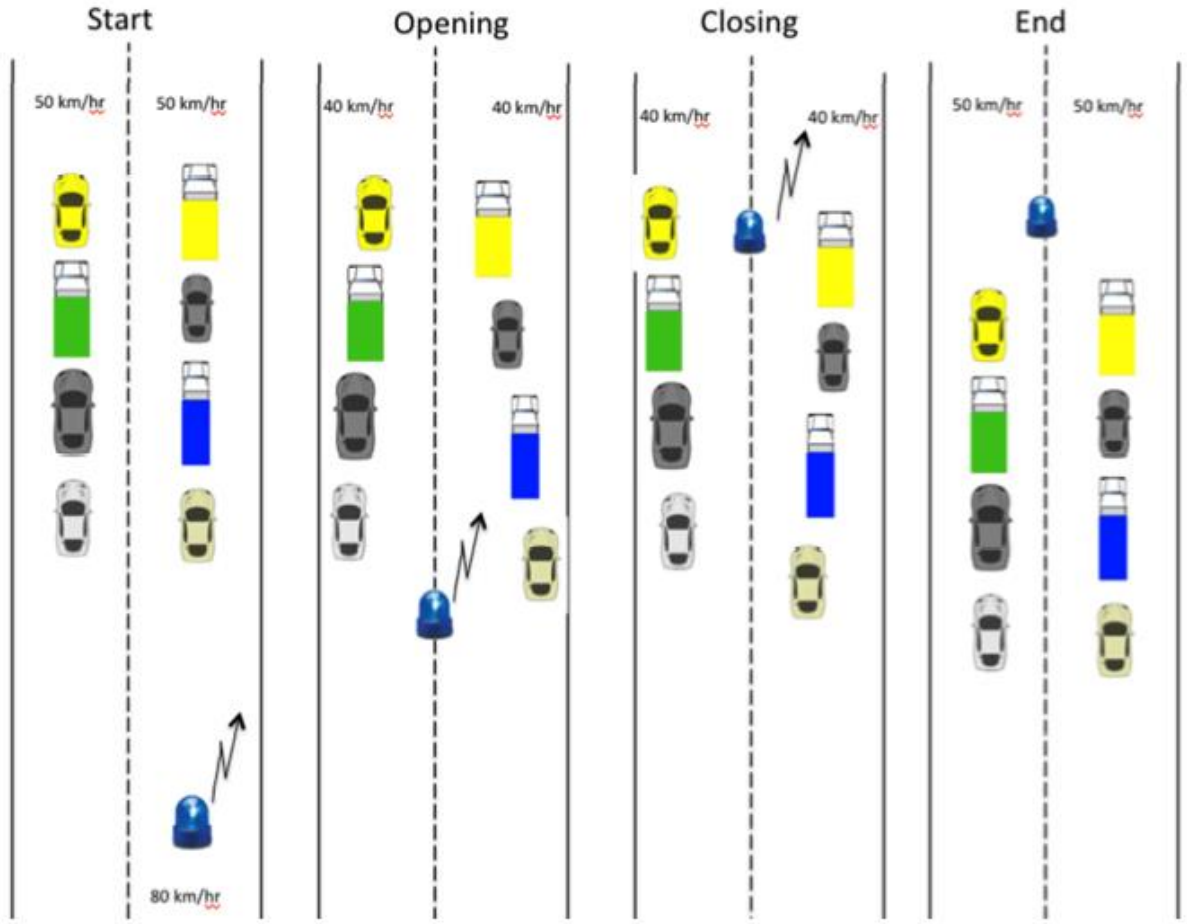


Figure 4: Scenario 3 description

Team

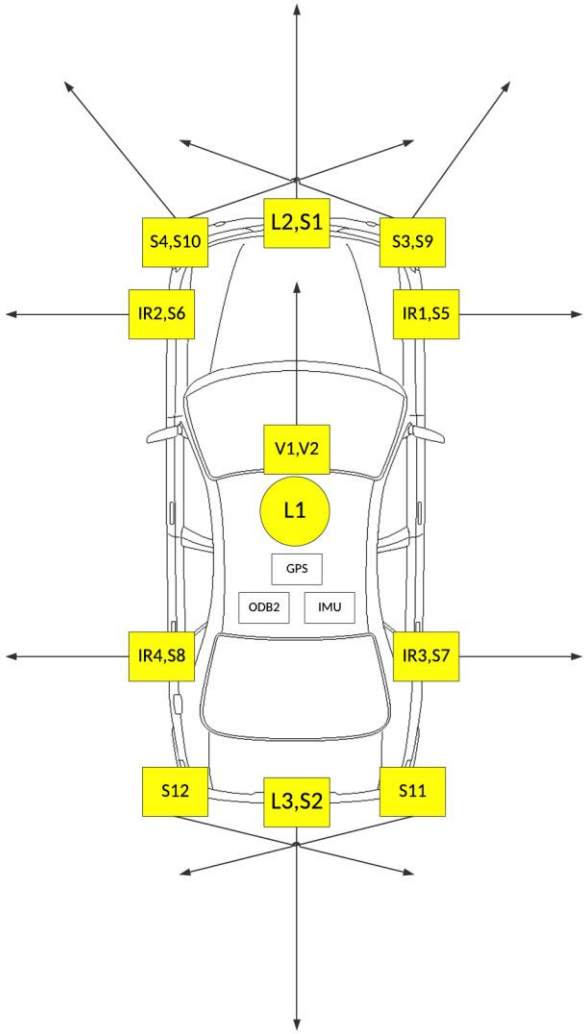
- Institure of Electronics and Computer Science (EDI)
- University of Latvia
- Riga Technical University

Mazda 6 (2004)

Sensors

Mandatory

- RTK-GPS (10Hz, 1cm, IMU)
- ITS-G5 stack (BTP/GeoNetworking/IEEE 801.11p)



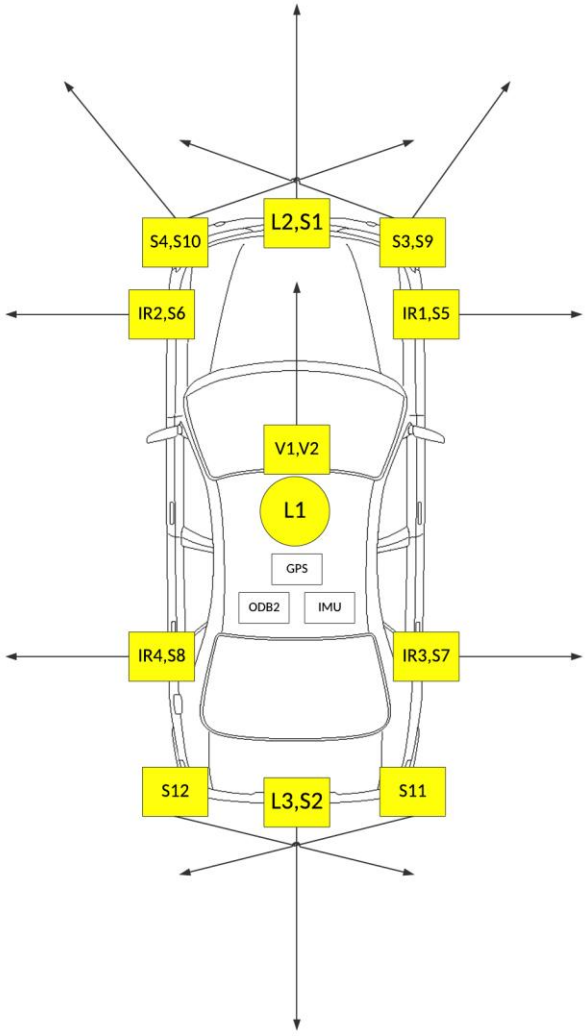
Sensors

Mandatory

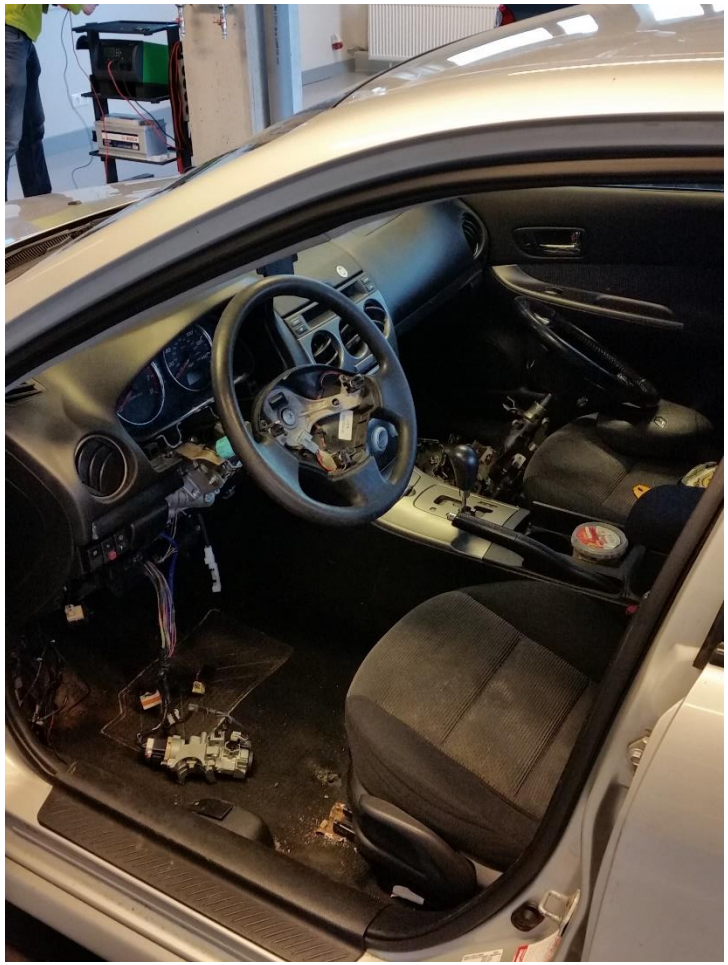
- RTK-GPS (10Hz, 1cm, IMU)
- ITS-G5 stack (BTP/GeoNetworking/IEEE 801.11p)

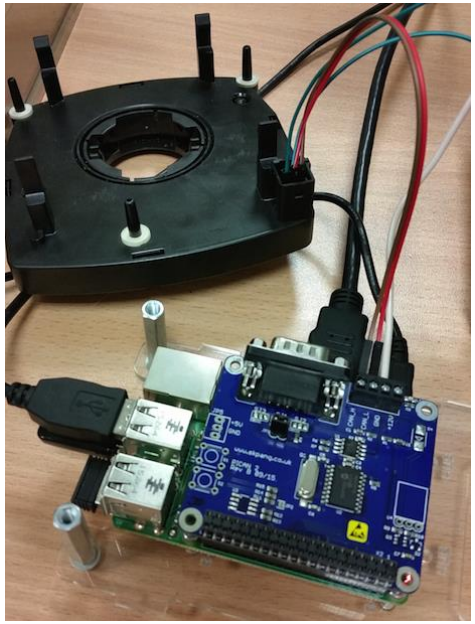
Optional

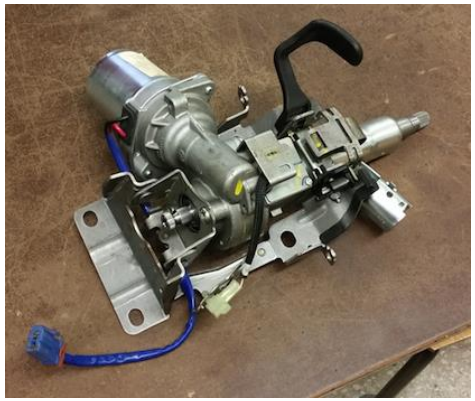
- LiDAR
- Stereo-vision
- Radar
- Ultrasound sensors
- et c.



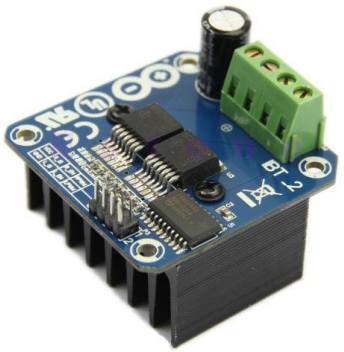
Steering control







- Renault Clio II steering motor
- IBT-2
- Steering angle sensor



Braking control

- Renault Laguna parking brake mechanism (as base)
- IBT-2
- Potentiometer for brake position feedback

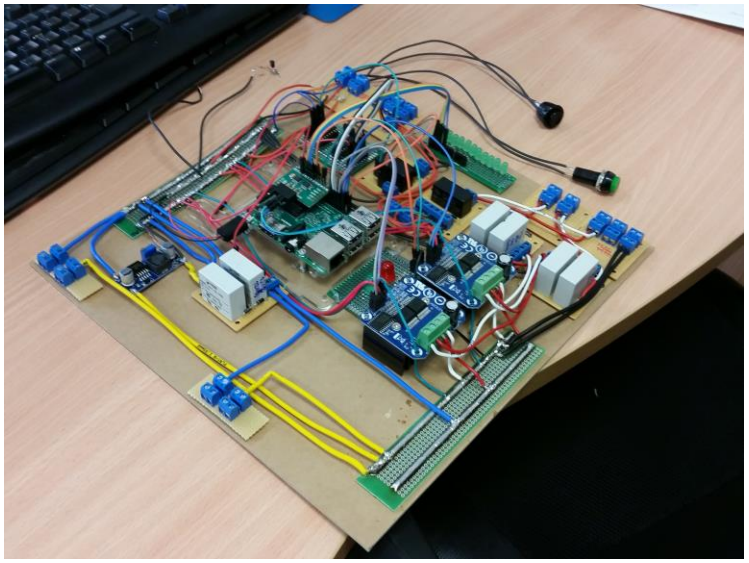
Gas control

By two voltages - V1, V2

Actuator control

Actuator control board

- Gas
- Brake
- Steering



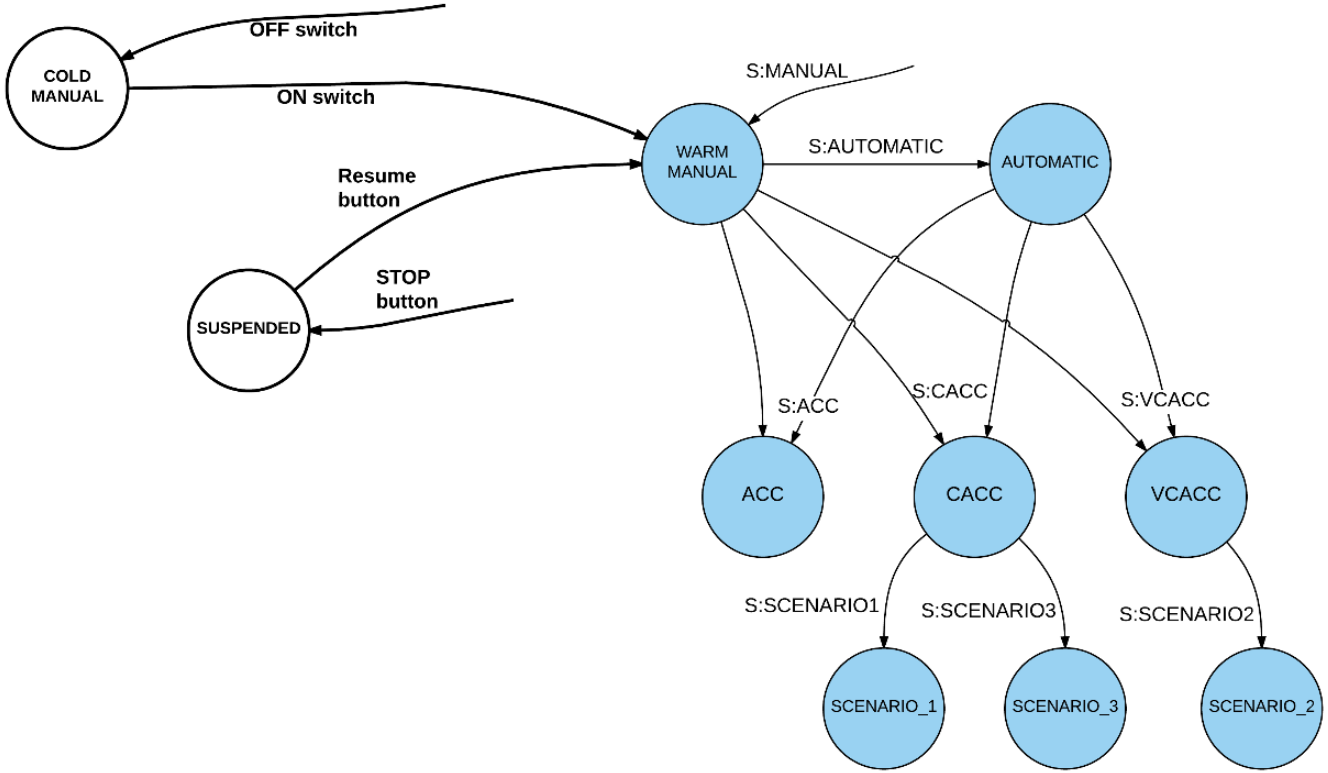
Communication module

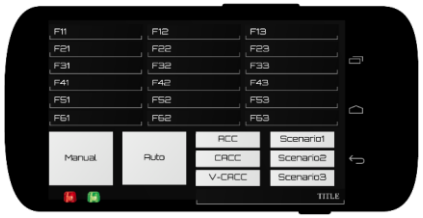
- BTP (transport)
- GeoNetworking (network)
- IEEE 802.11p (data-link)

PC Engines APU1D board:

- CPU: AMD G series T40E, 1 GHz dual Bobcat core with 64 bit support
- DRAM: 2 or 4 GB DDR3-1066 DRAM
- Storage: Boot from SD card (connected through USB), external USB or m-SATA SSD..
- 12V DC, about 6 to 12W depending on CPU load. Jack = 2.5 mm, center positive
- Connectivity: 3 Gigabit Ethernet channels (Realtek RTL8111E)
- I/O: DB9 serial port, 2 USB external + 2 internal, three front panel LEDs, pushbutton
- Expansion: 2 miniPCI express (one with SIM socket), LPC bus, GPIO header, I2C bus, COM2 (3.3V RXD / TXD)

States of the system





Stereo vision

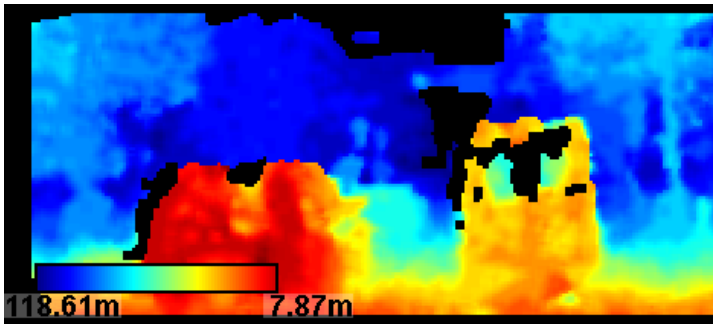
Capture
(2 frames)

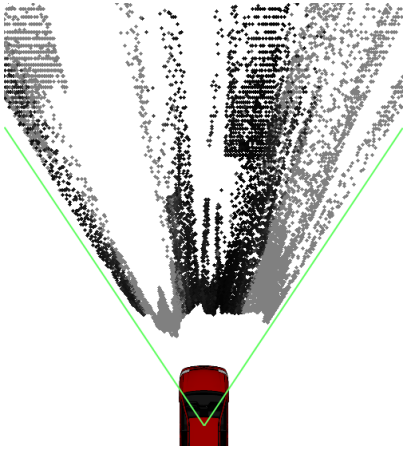
Calculate
point-cloud

Detect & follow
objects

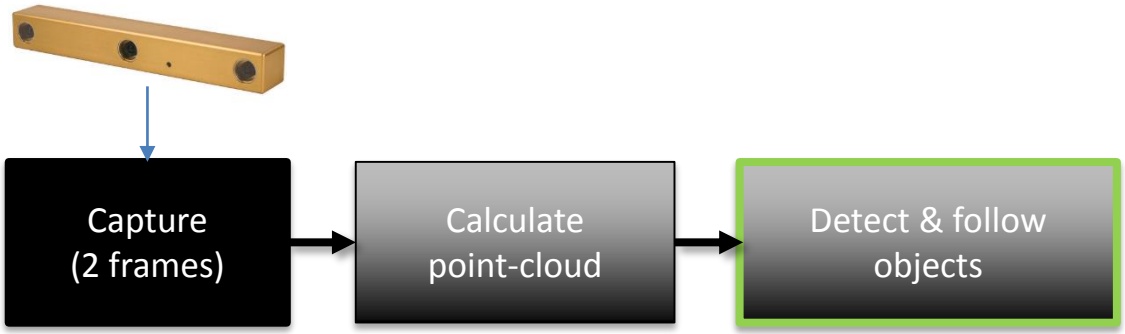
Stereo vision

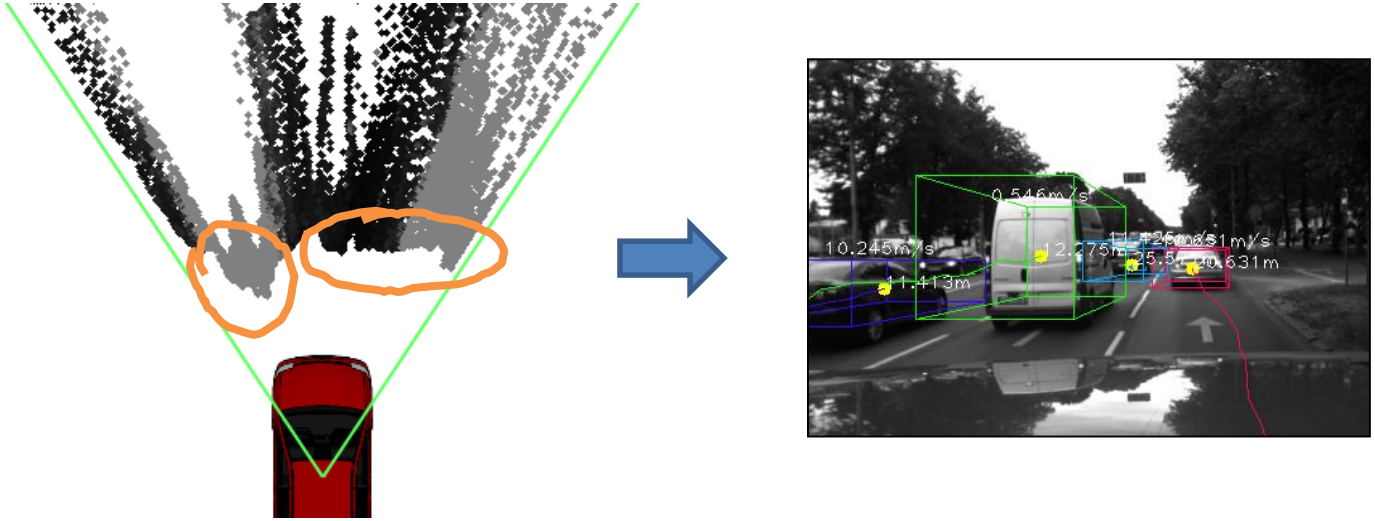
Capture (2 frames) → Calculate point-cloud → Detect & follow objects



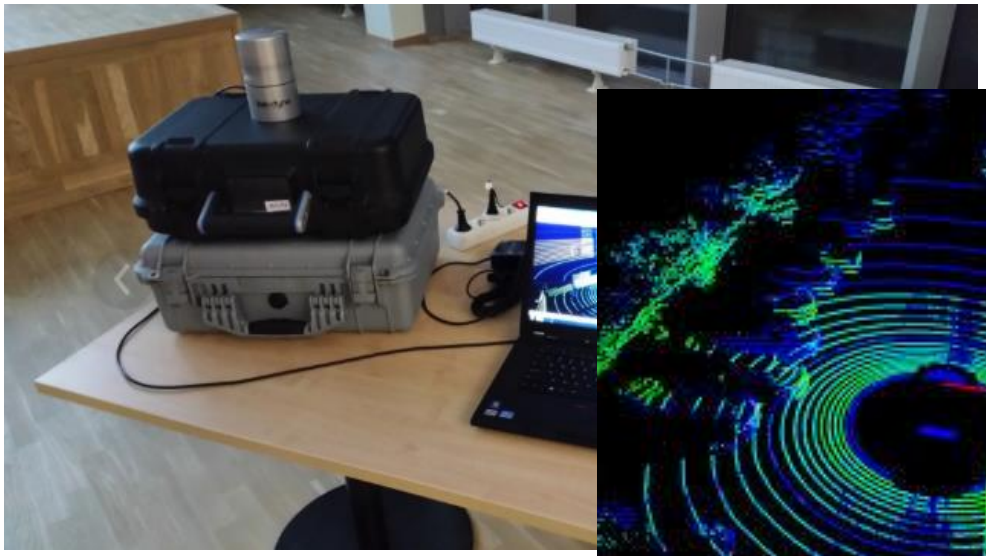


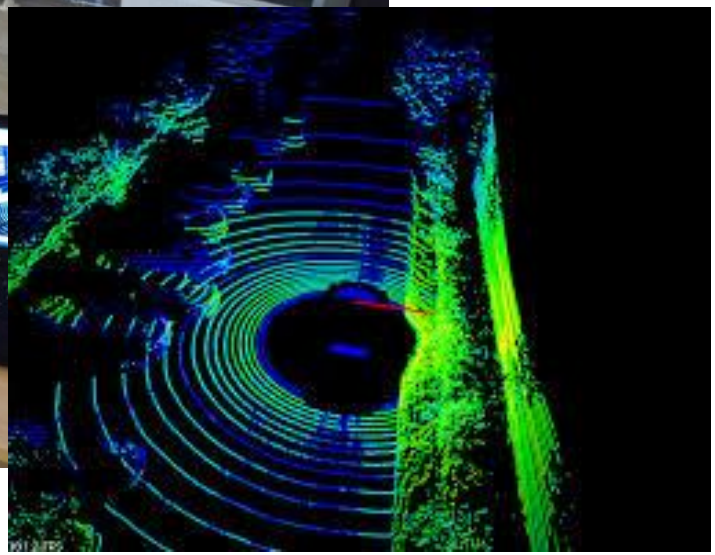
Stereo vision

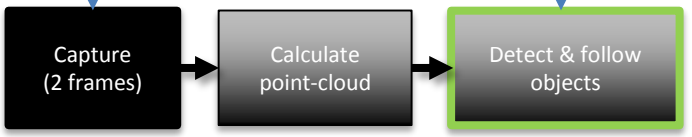


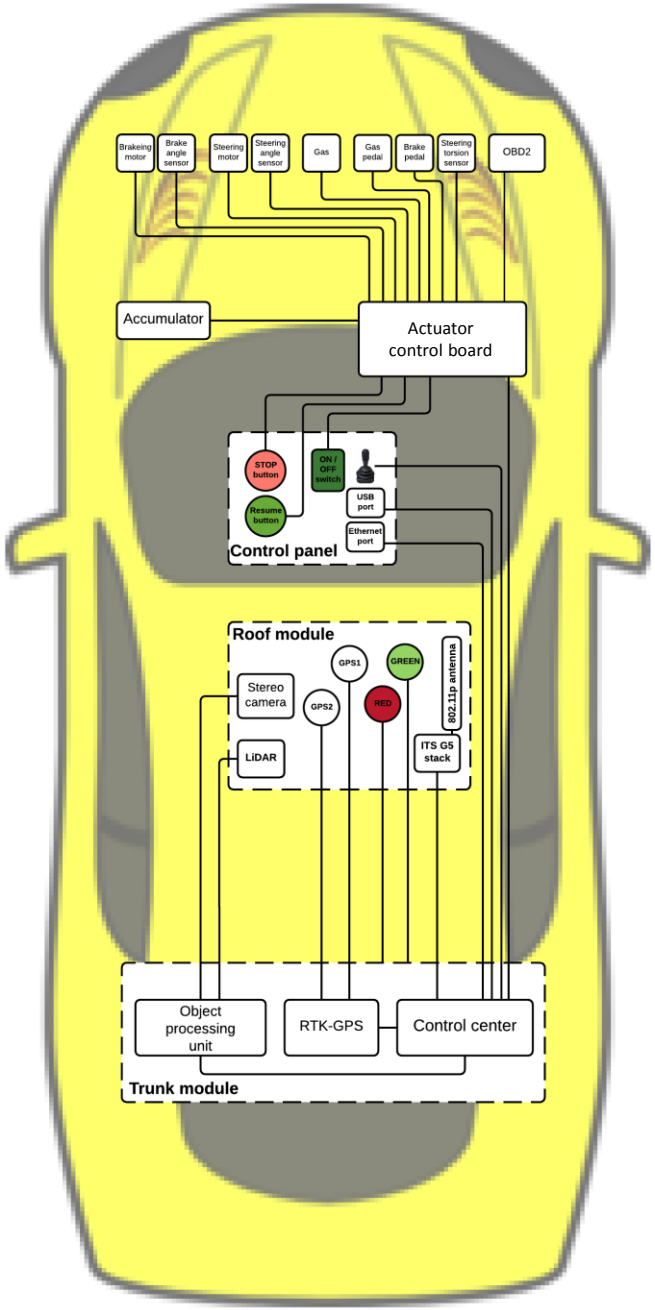


LiDAR (HDL-32E)








Brakeing motor
Brake angle sensor
Steering motor
Steering angle sensor
Gas
Gas pedal
Brake pedal
Steering torsion sensor
OBD2
Accumulator
Actuator control board
STOP button
ON / OFF switch
Resume button
USB port
Ethernet port
Control panel
Roof module
GPS1
GREEN
Stereo camera
GPS2
RED
802.11p antenna
ITS G5 stack
LiDAR
Object processing unit
RTK-GPS
Control center
Trunk module

IDIADA Safety workshop

IDIADA Safety workshop

Dates: 29.03.2016 - 01.04.2016

Features that were tested:

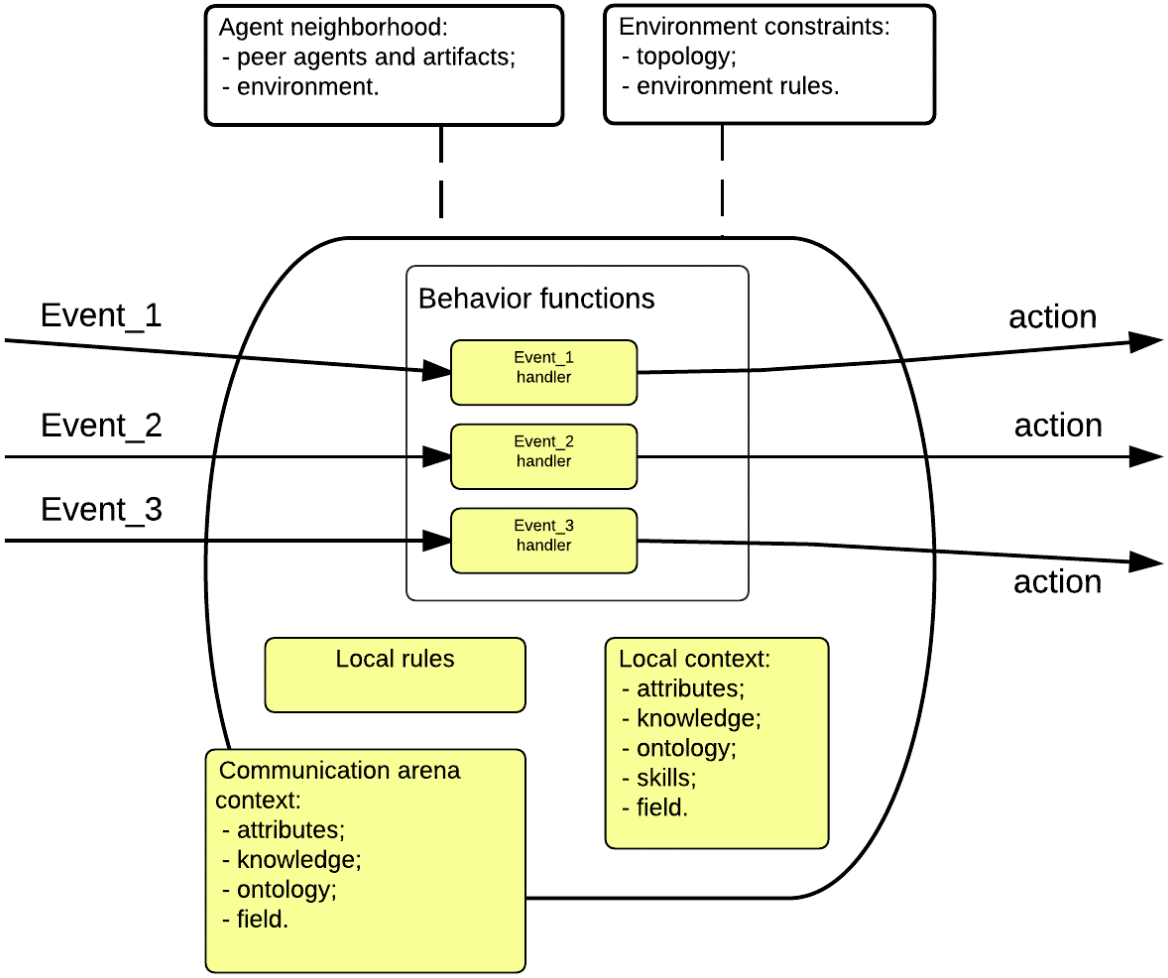
- Overall vehicle safety
- Brake state
- Manual takeover functionality

Next actvities

29.03-01.04 IDIADA safety workshop
(Barcelona, Spain)

23.05-31.05 GCDC-2016
(Helmond Automotive Campus, Netherlands)

Aģenta struktūra (modelis iter#2)



Aģenta struktūra (modelis iter#2)

Lai varētu modelēt aģentu materiālo mijiedarbību, mūsu aģenta modeli būtu jāpapildina sekojoši:

1. Katru sistēmas atribūtu jāsadala divās daļās:
   - **mērķa vērtība**
   - **aktuālā vērtība**

Aģenta algoritmi (behaviors) maina tikai atribūtu mērķa vērtības. Atbilstošās aktuālās vērtības ir funkcija no

- mērķa vērtībām
- aģenta spējām
- vides ierobežojumiem
- laika

Aģenta struktūra (modelis iter#2)

2. Aģenta modelim jāsastāv no diviem vienlaicīgi strādājošiem procesiem:

- Plāna izpildes process (izpilda plānu) - RT
- Plānošanas process (plāno/pārplāno) - ne-RT

Saturs

1. Darba tēma
2. Sistēmas konceptuālais modelis
3. Autonomā auto gatavošana GCDC-2016
4. Aģenta koriģētā arhitektūra

Jautājumi?