Pervasive Data Management in the Green Move System: a Progress Report

Emanuele Panigati Angelo Rauseo
Fabio A. Schreiber Letizia Tanca

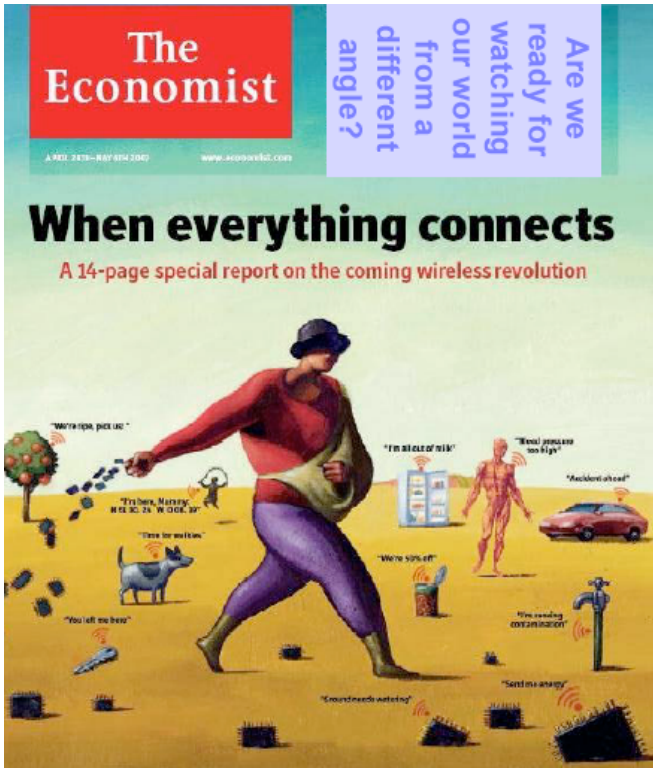
Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria

June 30, 2013

Summary of the content

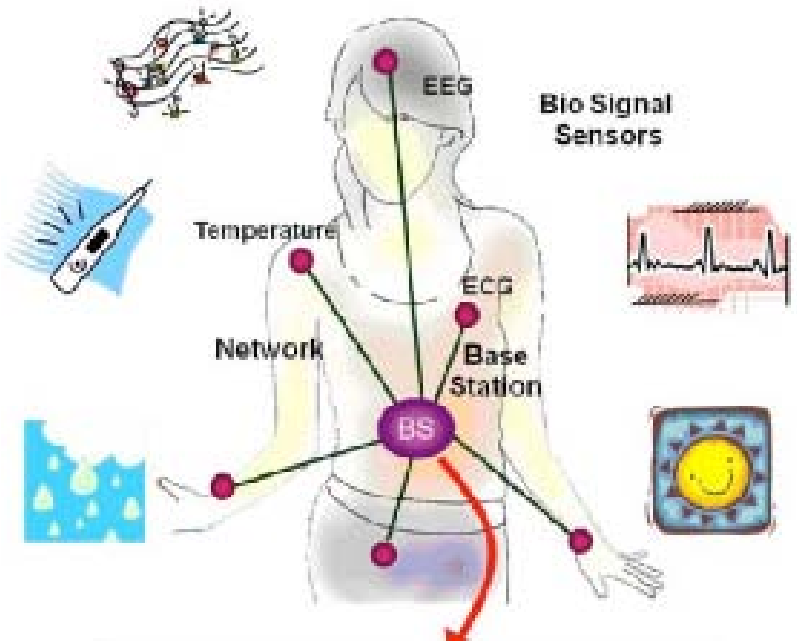
Situational, ubiquitous, context-aware computing

- From anytime, anyplace connectivity *for users* to connectivity and computing power available *for everything*



Situational, ubiquitous, context-aware computing

- From anytime, anyplace connectivity *for users* to connectivity and computing power available *for everything*
- RFID tags, wireless sensors and embedded devices allow the computer to *vanish in the background* becoming part of the infrastructure
- The heterogeneity of hundreds of devices must be made *transparent to the application and to the user*



Pervasive systems

- Health, well-being, climate change, energy and transportations are all areas whose progress relies on strategic use of pervasive systems, for climate and traffic, managing emergencies, governing smart cities, on-line alert systems etc.
- The perception of the environment promotes
  - interoperability
  - self-adaptivity
  - context-awareness
  - personalization
- In a pervasive system things "disappear", i.e. we are no more aware of their presence
- the system supports their management and we are free to use them without thinking
- $\Longrightarrow$ **we can focus on new goals**

A real case: the Green Move project

- An ongoing project financed by the Lombardia Region
- Involves 8 departments and research centers of Politecnico di Milano
- Designing and implementing a Zero-Emission-Vehicles (ZEV)-sharing system in Milan
- Coordination of vehicles owned by different public administrations or companies
- Services and information spread over the territory and among vehicle fleet owners and users

$\Rightarrow$ Huge amounts of heterogeneous data, coming from possibly large collections of participating entities, have to be collected, re-distributed and analyzed in a reasonable amount of time, in order to obtain useful and up-to-date information.

$\Rightarrow$ We adopt context-aware techniques to implement shared services, data gathering and information distribution.

A real case: the Green Move project

Green Move service configurations

condo-sharing for users who live in apartments and decide to share vehicles

firm-sharing for firms outsourcing their company vehicles to the Green Move sharing service

world-of-services users use a Green Move vehicle to reach a registered place offering dedicated services to Green Move customers (e.g. having the museum ticket charged on the Green Move monthly bill, skipping the queue at the ticket office)

generic users whose needs are not represented in any of the previous configuration; in this case the Green Move sharing service acts as a traditional vehicle sharing service

Integrated user experience

- The Green Move system aims at providing an integrated user experience among core and accessory services, like integrated services among commercial partners in the city, information and advertising based on users' interests and positions.
- Device interoperability and seamless data integration must be supported.
- The context-aware approach leads to the reduction of (noisy) information delivered to the users and to the satisfactory distribution of information, computation and network transmission loads among the various system's components.

System architecture

Three main components

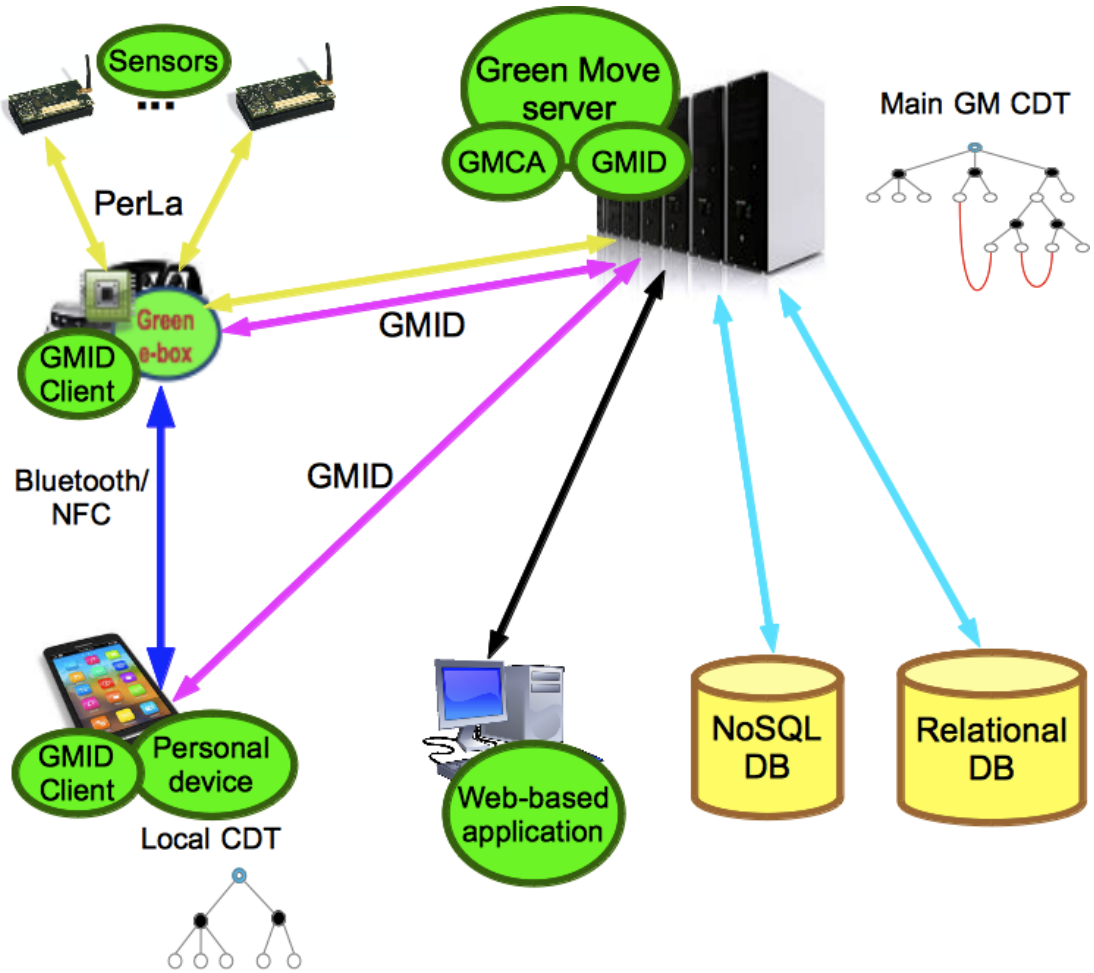
1. Central platform (Green Move server)
2. On-vehicle components (*Green-eBoxes*)
3. Users' personal devices

Information management main modules (server side)

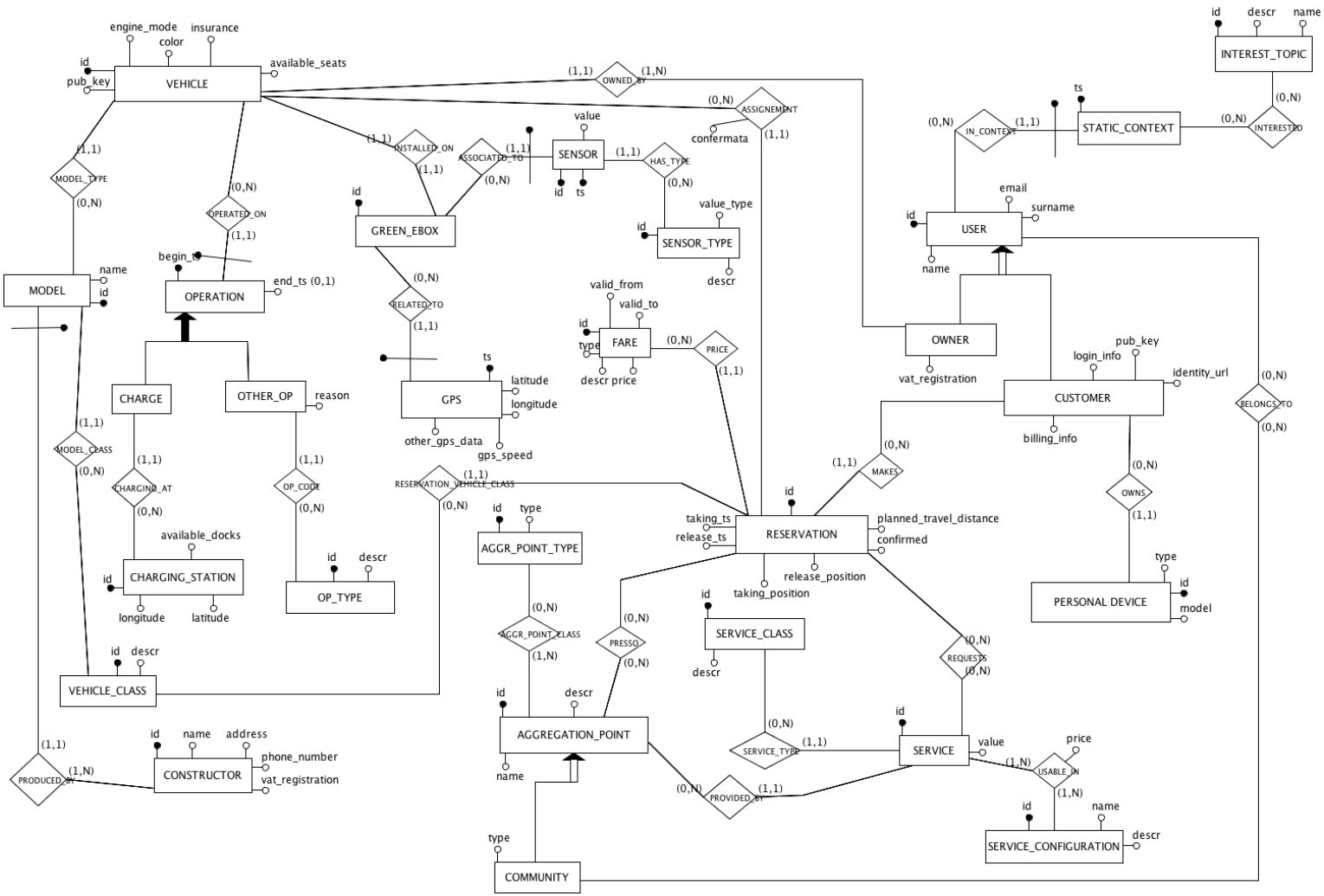
GMCA main, context-aware component; handle context-aware functions and vehicle to reservation assignments

GMID manage the information's distribution channel; deals with information pre-filtering.

System architecture



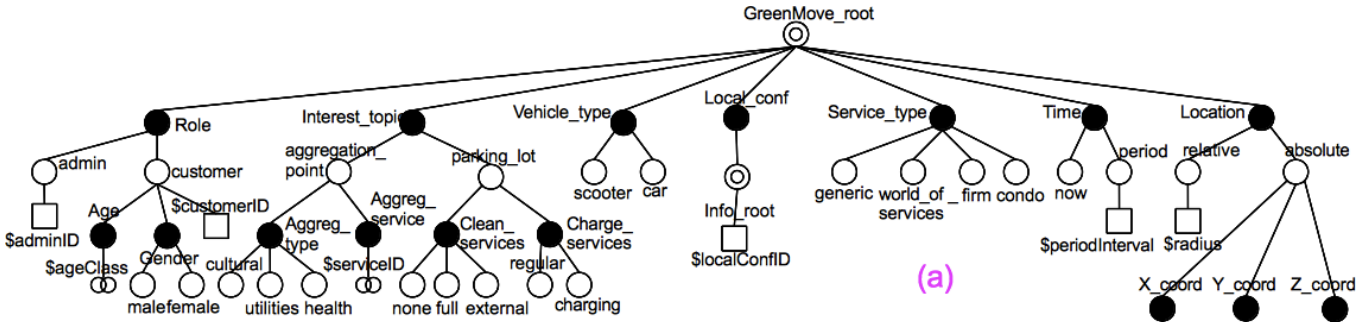
ER-Diagram


VEHICLE
engine_mode
color
insurance
available_seats
id
pub_key
MODEL
name
MODEL_TYPE
OPERATION
begin_ts
end_ts (0,1)
OPERATED_ON
CHARGE
OTHER_OP
reason
MODEL_CLASS
CHARGING_AT
OP_CODE
CHARGING_STATION
available_docks
longitude
latitude
OP_TYPE
VEHICLE_CLASS
PRODUCED_BY
CONSTRUCTOR
address
phone_number
vat_registration
GREEN_EBOX
INSTALLED_ON
ASSOCIATED_TO
SENSOR
HAS_TYPE
SENSOR_TYPE
value_type
descr
value
confermata
ASSIGNEMENT
OWNED_BY
RELATED_TO
GPS
other_gps_data
gps_speed
FARE
valid_from
valid_to
type
descr price
PRICE
RESERVATION
VEHICLE_CLASS
taking_ts
release_ts
planned_travel_distance
confirmed
release_position
taking_position
AGGR_POINT_TYPE
AGGR_POINT_CLASS
PRESSO
AGGREGATION_POINT
name
COMMUNITY
SERVICE_CLASS
SERVICE_TYPE
SERVICE
REQUESTS
USABLE_IN
price
SERVICE_CONFIGURATION
PROVIDED_BY
INTEREST_TOPIC
IN_CONTEXT
STATIC_CONTEXT
INTERESTED
USER
email
surname
OWNER
MAKES
CUSTOMER
login_info
identity_url
billing_info
BELONGS_TO
OWNS
PERSONAL DEVICE
model

Running Example

"Mr. Guido Verde" registered himself to the system since a Green Move condo-sharing facility is available at his condo, including a parking lot with a recharging station. Once registered, he decided to take full advantage of all services; he specified his data to the system and downloaded the Green Move application to his smartphone filling the private part of his profile. Beside more occasional usages, Mr. Verde typically uses the electric cars to take his granddaughter to school every morning, and sometimes stops, on the way home, at the supermarket for some shopping. Thanks to the private profile in the GMID client on his smartphone, Mr. Verde is also able of receiving interesting traffic information and ads according to the topic he has selected.

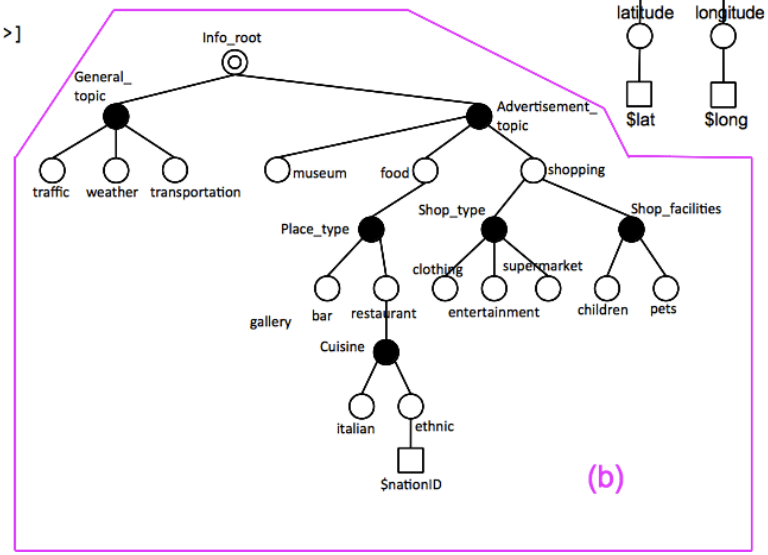
The Context Dimension Tree (CDT)



Context Element ce_1 : dimension = value[<par>]

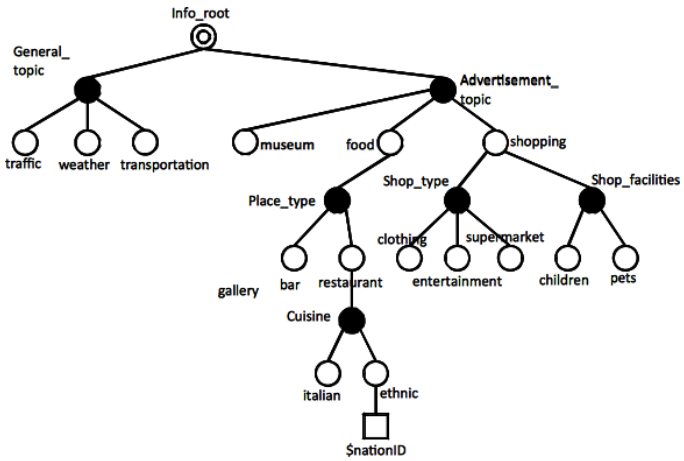
Context Instance $CI : \bigwedge_{i=1}^{n} ce_i$

LocalConfID : the ID for the local configuration matches the device ID



CIj = {Role = customer⟨1011⟩ ∧ Interest_topic = parking_lot ∧ Vehicle_type = car ∧ Service_type = condo}

Local CDT for Information distribution

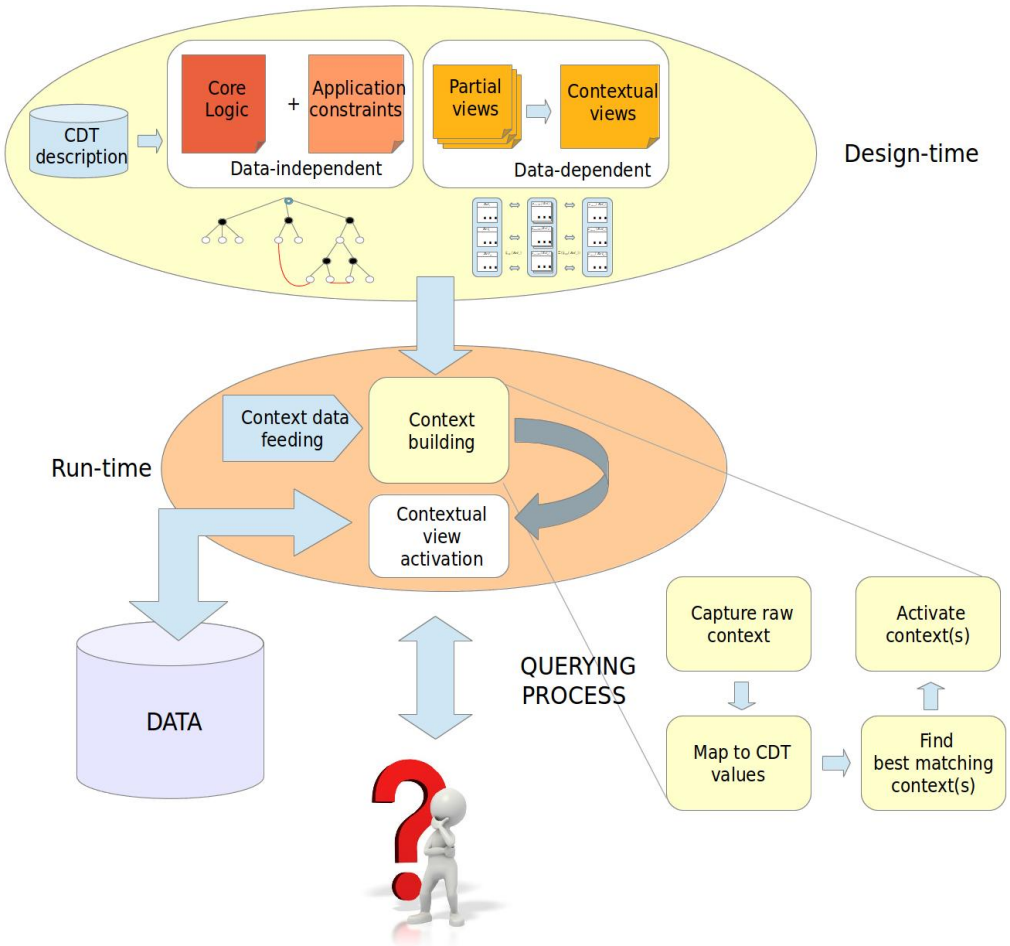


The Local CDT designed to support information distribution in Green Move

- The local CDT belongs to the user devices and is used to complete the data filtering process using private information
- The context instance *CCI* of a combined CDT is

$$CCI = CPI \wedge CLI = \bigwedge_{i=1}^{n} ce_i \wedge \bigwedge_{h=1}^{m} ce_h = \bigwedge_{k=1}^{n+m} ce_k$$

Green Move Context Awareness Framework



Context-aware tasks in Green Move

Actually three main tasks can take advantage of a context aware approach:

1. producing a personalized user experience ▸ Go
2. Context-aware sensors data retrieval and evaluation ▸ Go
3. information distribution to *Green Move* users ▸ Go

Tasks 1 and 2 are performed by the *GMCA* module, while task 3 is performed by the *GMID* module.

Personalized user experience

- The system tries to infer automatically the user preferences and his/hers habits from the actual user context and some context historical data.
- The whole process is supported by the PreMINE framework, a contextual preference mining framework.

Example

Mr. Verde usually reserves a car for taking his nephew to school from 8 to 10 AM, including a child seat in his request.
RESULT: Whenever he makes a reservation starting at 8 AM, the system will automatically propose to include the child seat in the request.

Context-aware sensors

To avoid useless transmissions, always ensuring timeliness constraints, context-aware sensor data management is performed using the PerLa framework, which provides:

- a middleware infrastructure suitable for collecting data from different sensors' types and data gathering peripherals
- *PerLa* (*Per*vasive *La*nguage), a declarative, SQL-like language for sensors querying, which retrieves and manages the data produced by the sensors, obtaining adaptability to different types of sensors and transparency w.r.t. the underlying sensor network configuration.

Example

Knowing Mr. Verde position, the PerLa module installed on the *Green-eBox* compute the actual user's context, allowing to properly set up the data retrieval and transmission frequencies, sending the data more often when necessary.

Context-aware sensors – Queries

Driving in the suburbs

```
CREATE CONTEXT Suburbs_Driving
ACTIVE IF lat > center_lat + max_dist AND long > center_long + max_dist
ON_ENABLE:
SELECT lat, long, batt_charge
SAMPLING EVERY 60 s WHERE batt_charge <= 0.5
EXECUTE IF EXIST lat, long, speed, batt_charge AND speed > 0
SET PARAMETER 'alarm' = TRUE WHERE batt_charge <= 0.35;
ON_DISABLE:
DROP Suburbs_Driving;
SET PARAMETER 'alarm' = FALSE;
REFRESH EVERY 5 m;
```

Driving downtown

```
CREATE CONTEXT Downtown_Driving
ACTIVE IF lat <= center_lat + max_dist AND long <= center_long + max_dist
ON_ENABLE:
SELECT lat, long, batt_charge
SAMPLING EVERY 120 s WHERE batt_charge <= 0.5
EXECUTE IF EXIST lat, long, speed, batt_charge AND speed > 0
SET PARAMETER 'alarm' = TRUE WHERE batt_charge <= 0.35;
ON_DISABLE:
DROP Downtown_Driving;
SET PARAMETER 'alarm' = FALSE;
REFRESH EVERY 5 m;
```

Information distribution to *Green Move* users I

Information message composition

- a short caption
- an (optional) image
- a data structure (e.g. an XML-like file) describing the topics dealt within this specific advertisement or information (chosen among the ones described in the local CDT)

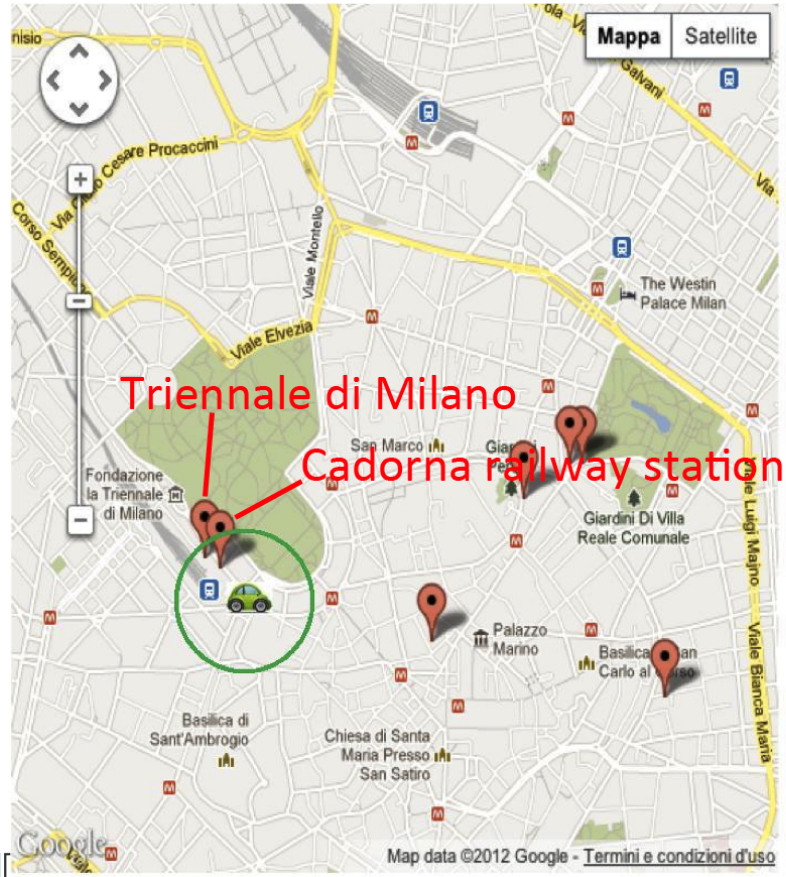
Information distribution to *Green Move* users II

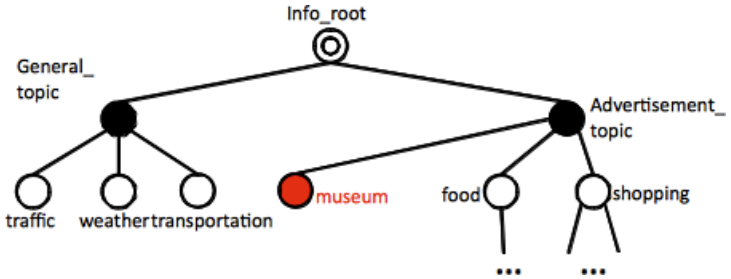
The system needs to personalize the information to be sent to each user. In order to obtain this result, the system performs two steps:

- *pre-filtering* (on the server) using age, gender, time and distance among client gps position (user *public* information) and ad/message geolocalized descriptor using the primary CDT contextual information
- *filtering* (on the user's personal device) using all the detailed information available about user preferences (user *private* information) using the local CDT contextual information

After these two steps, the user's personal device displays only a little set of interesting information to the user.

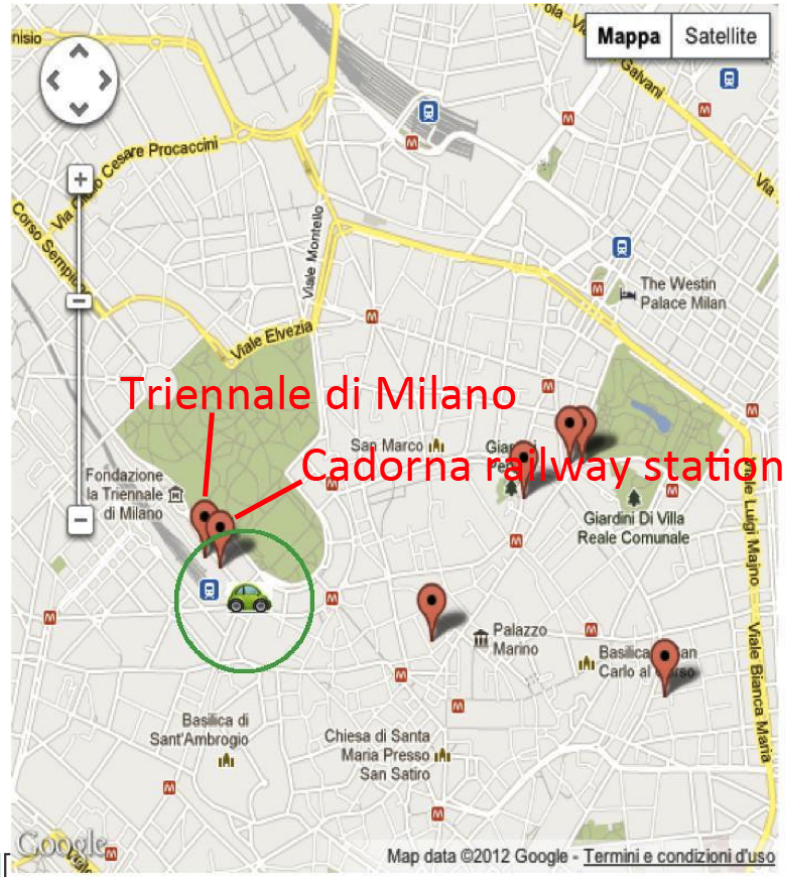
Information distribution – Example

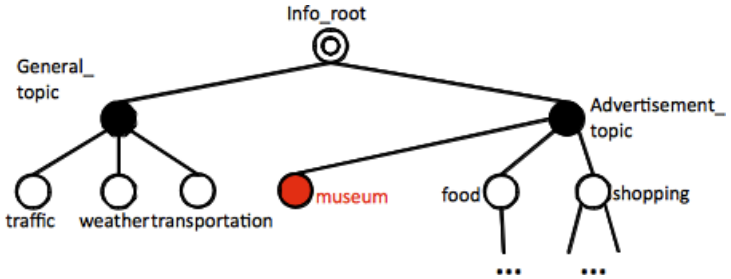




$$CI = \{Role = customer\langle 1011\rangle \wedge X\_coord = latitude\langle 45.468\rangle \wedge Y\_coord = longitude\langle 9.177\rangle \wedge Advertisement\_topic = museum\}$$

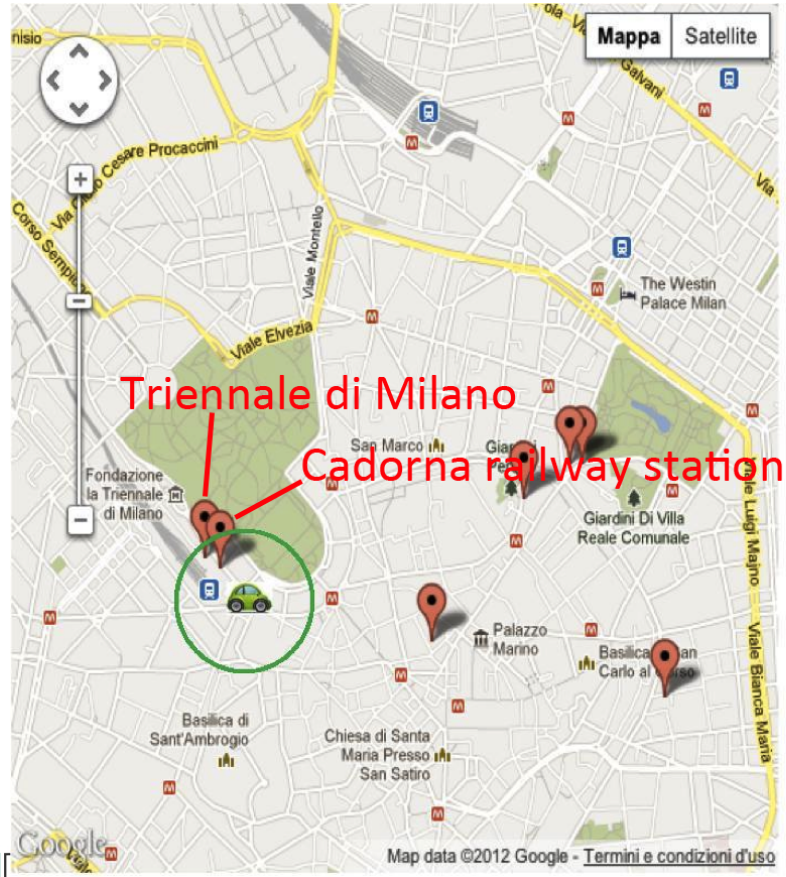
Information distribution – Example

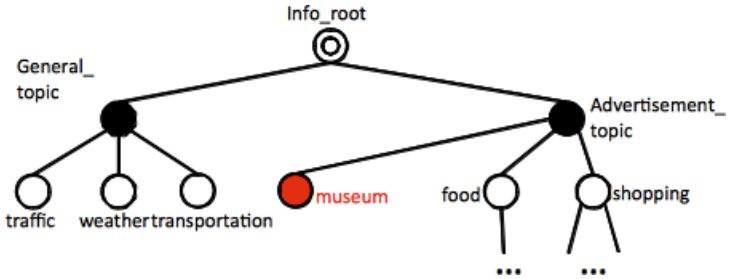




$$CI_{server} = \{Role = customer\langle 1011\rangle \wedge X\_coord = latitude\langle 45.468\rangle \wedge Y\_coord = longitude\langle 9.177\rangle\}$$

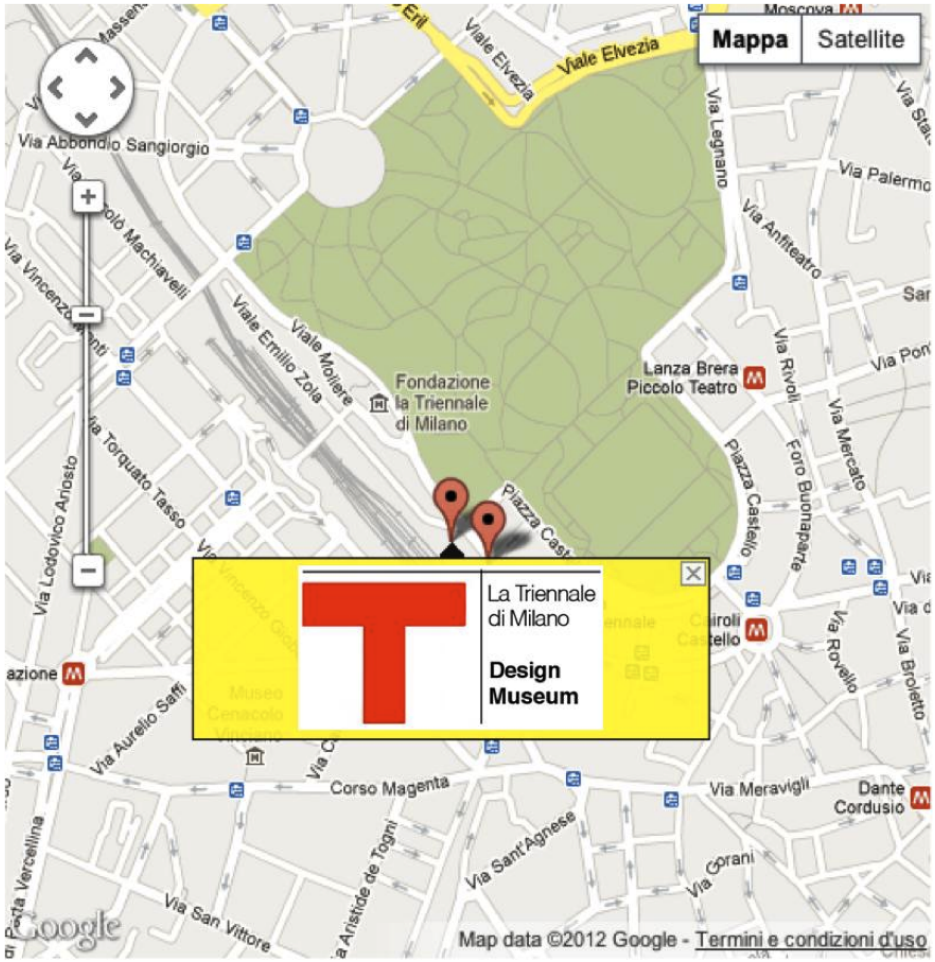
Information distribution – Example





$$CI_{local} = \{Advertisement\_topic = museum\}$$

Information distribution – Example



Answer Set Programming (ASP) prototype

We are currently working on a rapid-development prototype based on the Green Move specifications for data management. The main implementation language is Answer Set Programming (ASP), in particular the DLV system, to implement

- context management and data tailoring for database access,
- modeling the vehicle reservation system and using contextualized historical data to assign and make forecasts about the availability of the vehicles in a given time interval.

Example (I)

context_elem(customer) ∧ parameter(customerId, ``1'') ∧ sensed(scooter)

id	name	surname	age	gender	email	customer	owner	username
1	Guido	Verde	middle_age	M	...	TRUE	TRUE	gm
2	Guido	Maluccio	young	M	...	TRUE	TRUE	bd
3	Giovanni	Storti	late	M	...	TRUE	TRUE	aggi
...								
11	Homer	Simpson	middle_age	M	...	FALSE	TRUE	rwarr
...								

Table : User data (complete)

id	name	surname	age	gender	email	customer	owner	username
1	Guido	Verde	middle_age	M	...	TRUE	TRUE	gm

Table : User data (according to the current context)

Example (II)

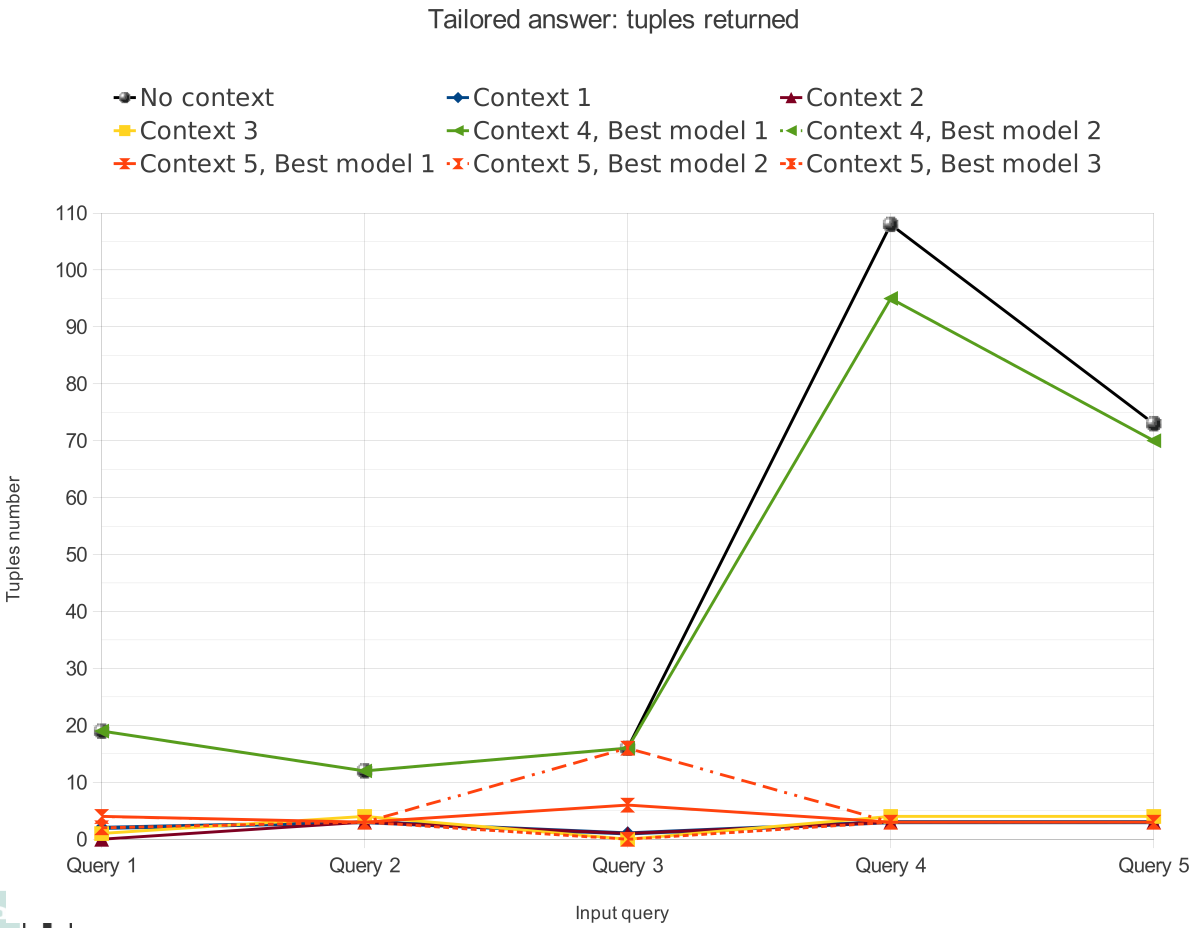
`sensed(admin)` $\wedge$ **sensed(car)**

id	name	surname	age	gender	email	customer	owner	username
1	Guido	Verde	middle_age	M	...	TRUE	TRUE	gm
2	Guido	Maluccio	young	M	...	TRUE	TRUE	bd
3	Giovanni	Storti	late	M	...	TRUE	TRUE	aggi
				...				
11	Homer	Simpson	middle_age	M	...	FALSE	TRUE	rwarr
				...				

Table : User data (complete)

id	name	surname	age	gender	email	customer	owner	username
1	Guido	Verde	middle_age	M	...	TRUE	TRUE	gm
2	Guido	Maluccio	young	M	...	TRUE	TRUE	bd
3	Giovanni	Storti	late	M	...	TRUE	TRUE	aggi
				...				
11	Homer	Simpson	middle_age	M	...	FALSE	TRUE	rwarr
				...				

Experiments



- The "No context" case is the upper bound for the size of the answer set
- The answer set returned through the context-aware approach is in general smaller

Conclusions

- Urban mobility and energy saving are among the driving factors in future smart-city planning and development.
- The Green Move project aims at realizing a zero-emission-vehicle (ZEV) sharing service, that also includes pervasive information distribution to assist the driver both in driving and in personal activities.
- The Green Move context-and-preference-aware information collection/dissemination service provides the right information to the right person at the right moment, overcoming heretogeneity and reducing information noise

Further references

- D. Beretta, E. Quintarelli, and E. Rabosio. *Mining context-aware preferences on relational and sensor data*
- C. Bolchini, E. Quintarelli, and L. Tanca. *CARVE: Context-aware automatic view definition over relational databases*
- C. Cappiello and F. A. Schreiber. *Quality- and energy-aware data compression by aggregation in wsn data streams*
- L. Carrara and G. Orsi. *A new perspective in pervasive advertising*
- A. Rauseo, D. Martinenghi, and L. Tanca. *Context-aware data tailoring through answer set programming*
- F. Schreiber, R. Camplani, M. Fortunato, M. Marelli, and G. Rota. *PerLa: A language and middleware architecture for data management and integration in pervasive information systems*
- G. Cugola, A. Margara, A. Morzenti, E. Panigati, M. Rossi, S. Savaresi, F. A. Schreiber, L.Tanca, et al. *Green Move: towards next generation sustainable smartphone-basedvehicle sharing*, In Proceedings of SustainIT2012, Pisa, 2012
- E. Panigati, A. Rauseo, F. A. Schreiber, L. Tanca. *Context-aware Information Management in the Green Move System*, submitted to Fifth Interop-Vlab.it Workshop (co-located with itAIS2012), Rome, 2012

ASP code excerpts

```
%  GUESS  %
% Assign a vehicle to a reservation
assignement(VId, ResrvId) v -assignement(VId, ResrvId) :-
        reservation(ResrvId, _, _, _, _, _),
        vehicle(VId, _).

%  STRONG CONSTRAINTS  %
% If two reservations overlaps,
  two different vehicles must be assigned to them
:- overlaps_resrv(ResrvId1, ResrvId2),
        assignement(VId1, ResrvId1),
        assignement(VId2, ResrvId2), VId1 == VId2.
[...]

%  WEAK CONSTRAINTS  %
% Weak constraints are listed from stronger to weaker
% (syntax: [weight:level])

% Try to satisfy all reservations
:~ reservation(ResId, User, SDay, SHour, EDay, EHour),
        not assignement(VId, ResId), vehicle(VId, _). [1:2]

% Try not to assign vehicles which need to be charged
:~ reservation(ResrvId, _, _, _, _, _),
        assignement(VId, ResrvId),
        vehicle(VId, ChargePerc), ChargePerc <= 30. [1:1]
```

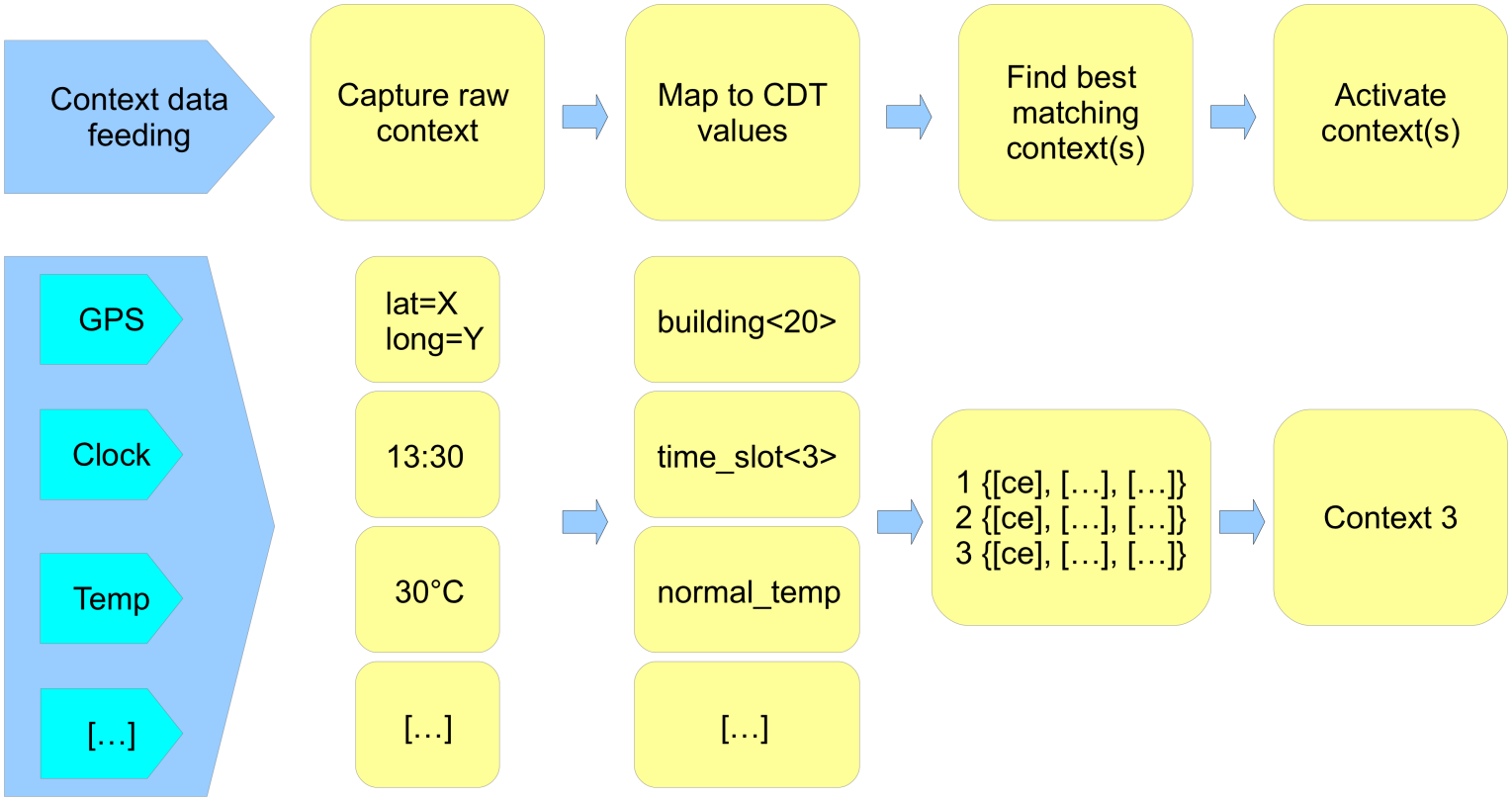
```
%dimension(Dimension)
dimension(role).
dimension(interest_topic).
dimension(vehicle_type).
dimension(local_conf). [...]

%value(Value).
value(root).
value(admin).
value(customer).
value(aggregation_point).
value(parking_lot).
value(scooter).
value(car).
value(info_root). [...]

%dim2val(Dimension, Value).
dim2val(role,admin).
dim2val(role,customer).
dim2val(interest_topic,aggregation_point).
dim2val(interest_topic,parking_lot).
dim2val(vehicle_type,scooter).
dim2val(vehicle_type,car).
dim2val(local_conf,info_root). [...]

%val2dim(Value, Dimension).
val2dim(root,role).
val2dim(root,interest_topic).
val2dim(root,vehicle_type).
val2dim(root,local_conf). [...]
```

Context building



Example (III)

$$\texttt{context\_elem(customer)} \wedge \texttt{parameter(customerId,``1'')} \wedge \texttt{sensed(scooter)}$$

id	seats	engine_type	model_id	owner_id
1	2	electric	1	1
2	2	electric	2	1
3	4	hybrid	3	5

Table : Vehicle data (complete)

id	seats	engine_type	model_id	owner_id
1	2	electric	1	1

Table : Vehicle data (according to the current context)

Notes:
model_id $= 1 \rightarrow$ **vehicle_class** $=$ scooter

Example (IV)

sensed(admin) $\wedge$ **sensed(car)**

id	seats	engine_type	model_id	owner_id
1	2	electric	1	1
2	2	electric	2	1
3	4	hybrid	3	5

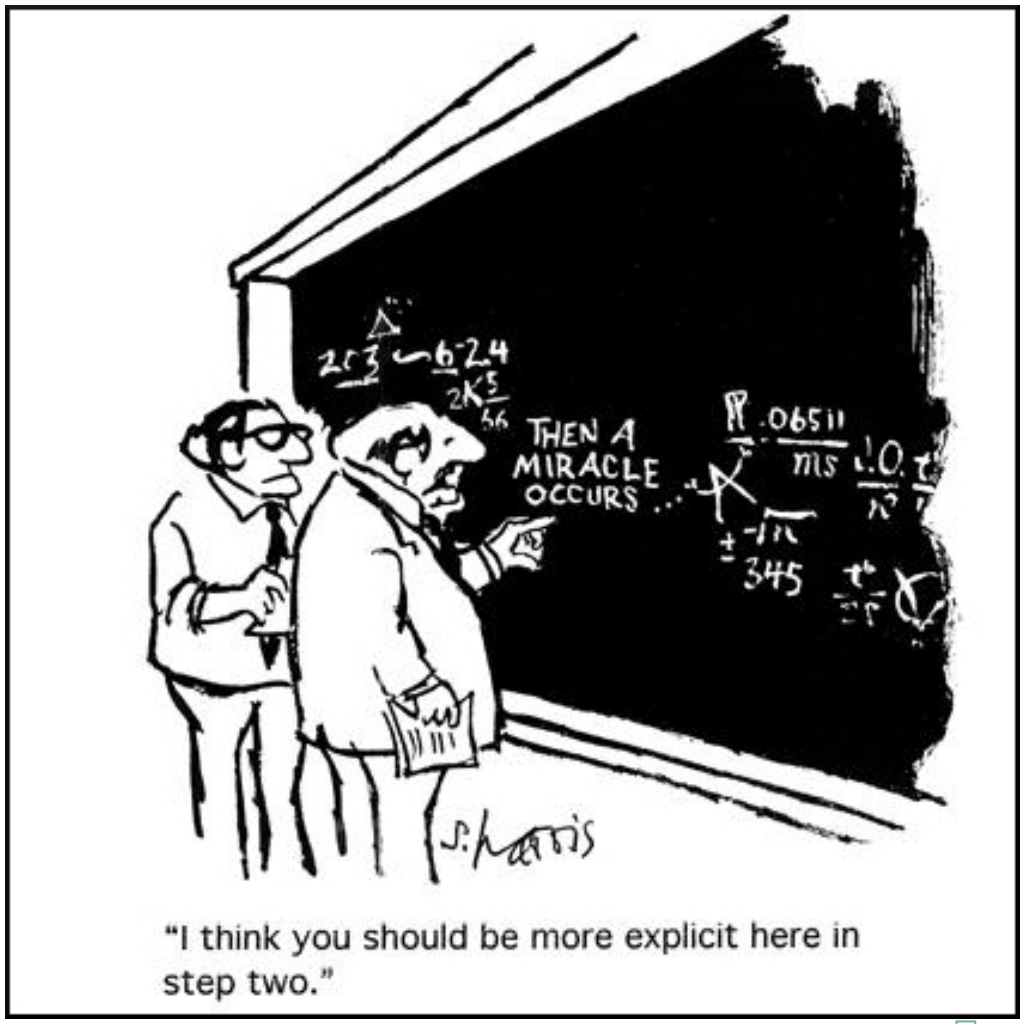
Table : Vehicle data (complete)

id	seats	engine_type	model_id	owner_id
2	2	electric	2	1
3	4	hybrid	3	5

Table : Vehicle data (according to the current context)

Notes:
model_id = 2 or 3 $\rightarrow$ **vehicle_class** = car

Thanks


THEN A MIRACLE OCCURS ...
"I think you should be more explicit here in step two."