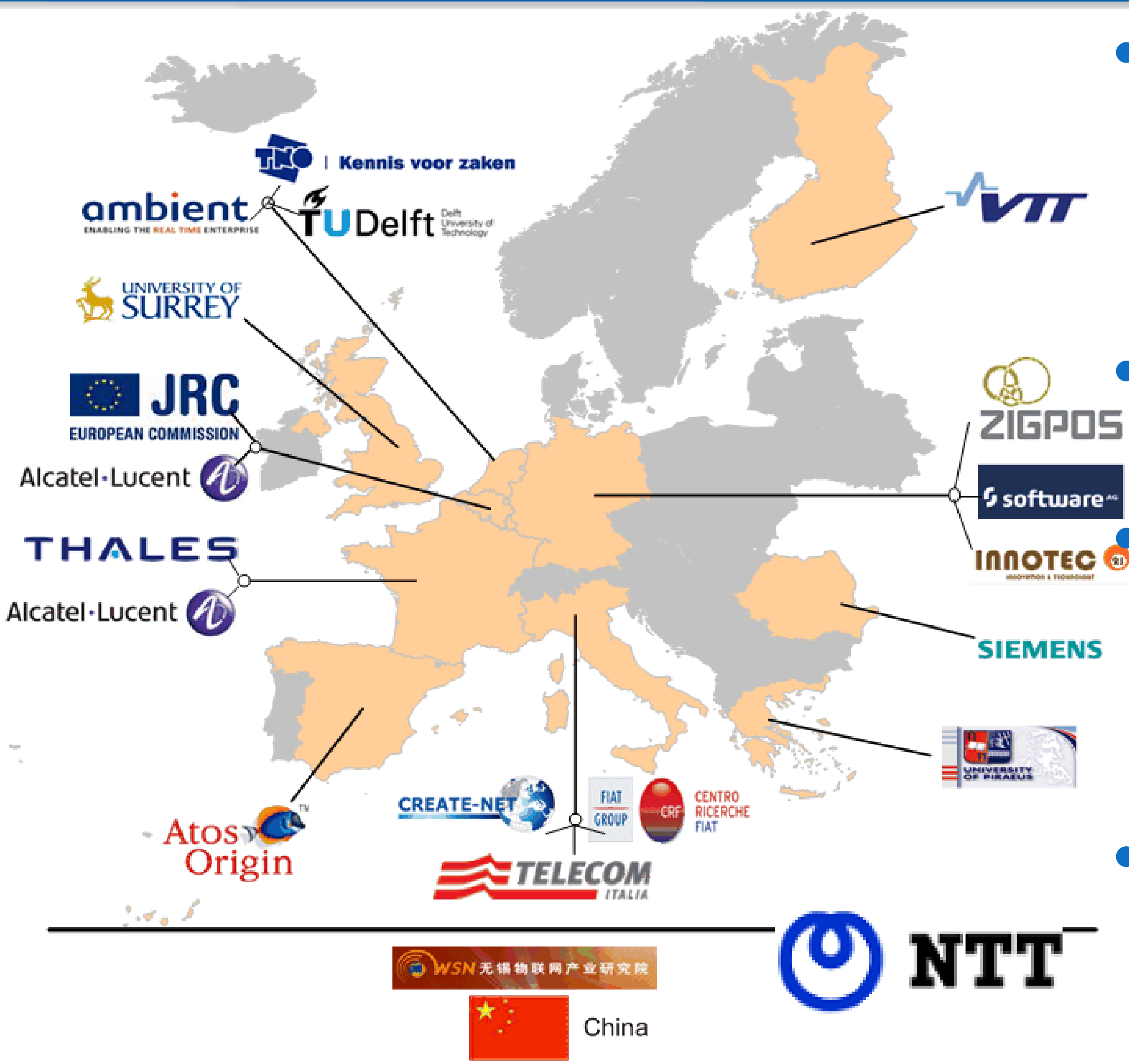
iCore: Internet Connected Objects for Reconfigurable Eco-system

IERC AC4 SEMANTIC INTEROPERABILITY WORKSHOP
IERC AC4 Venice Meeting

Konstantinos Kotis (VTT) & Abdur Rahim Biswas (Create-Net)

iCore identification

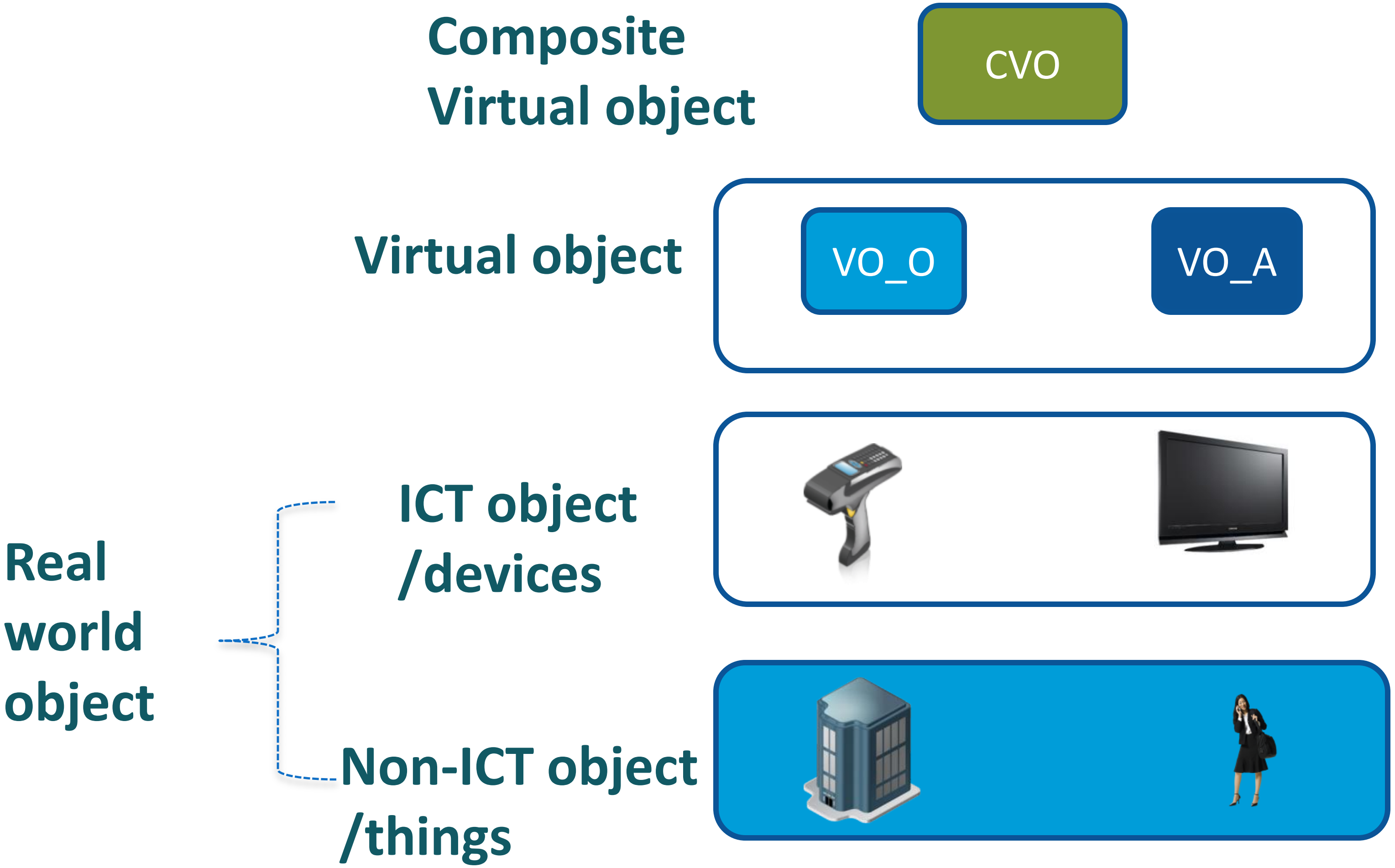


- **iCore-**Internet Connected Object for Reconfigurable Eco-system
- **3 yrs** EU FP7 Integrated Project (1st Oct 2011)
- 20 Partners with strong industrial representation (across 12 countries)
- EU + China and Japan

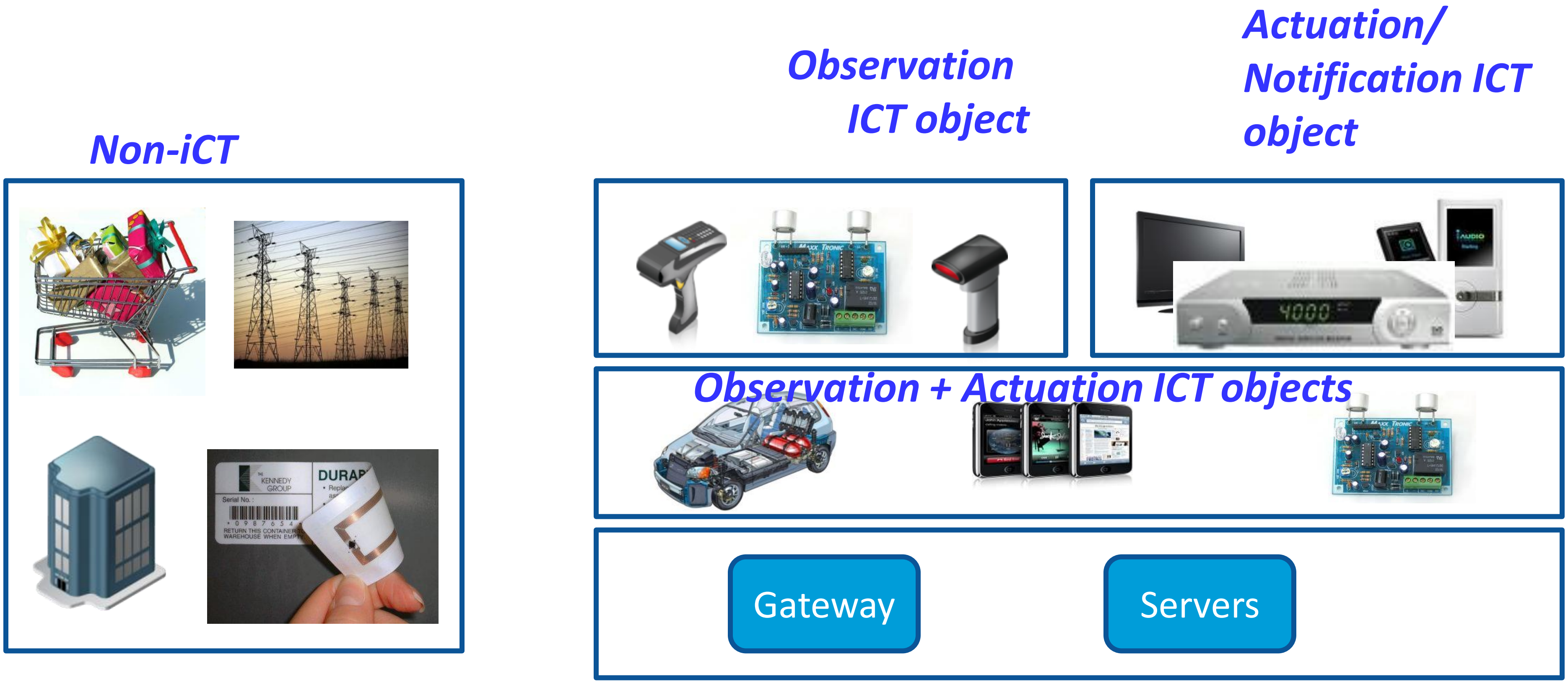
Concepts

- Making life easier for application developer, service provide and operators
- Increase the vertical **reusability of objects** outside the scope in which they were originally deployed
- Address **interoperability** issues
- **Open cognitive framework** for the Internet of Things (IoT) addressing three levels:
  - **Virtual Objects (VOs):** Virtual representations of real-world objects
  - **Composite Virtual Objects** (CVOs): Mash-ups of multiple VOs
  - **Users/stakeholders** perspectives

Terminology



Real and physical objects model

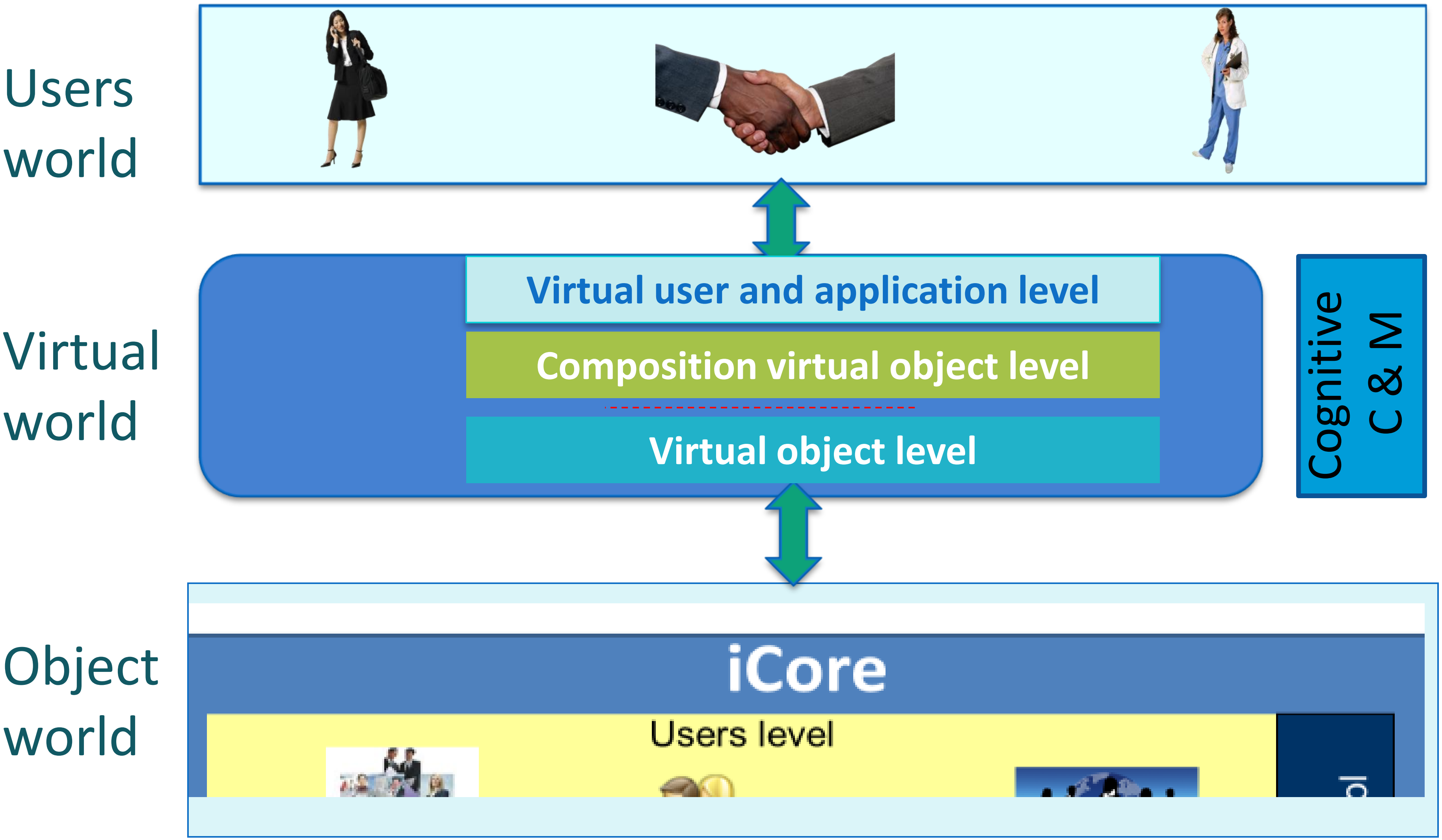


- Objects directly or indirectly connected or associated with internet
- Characterization of iCore real and physical objects?

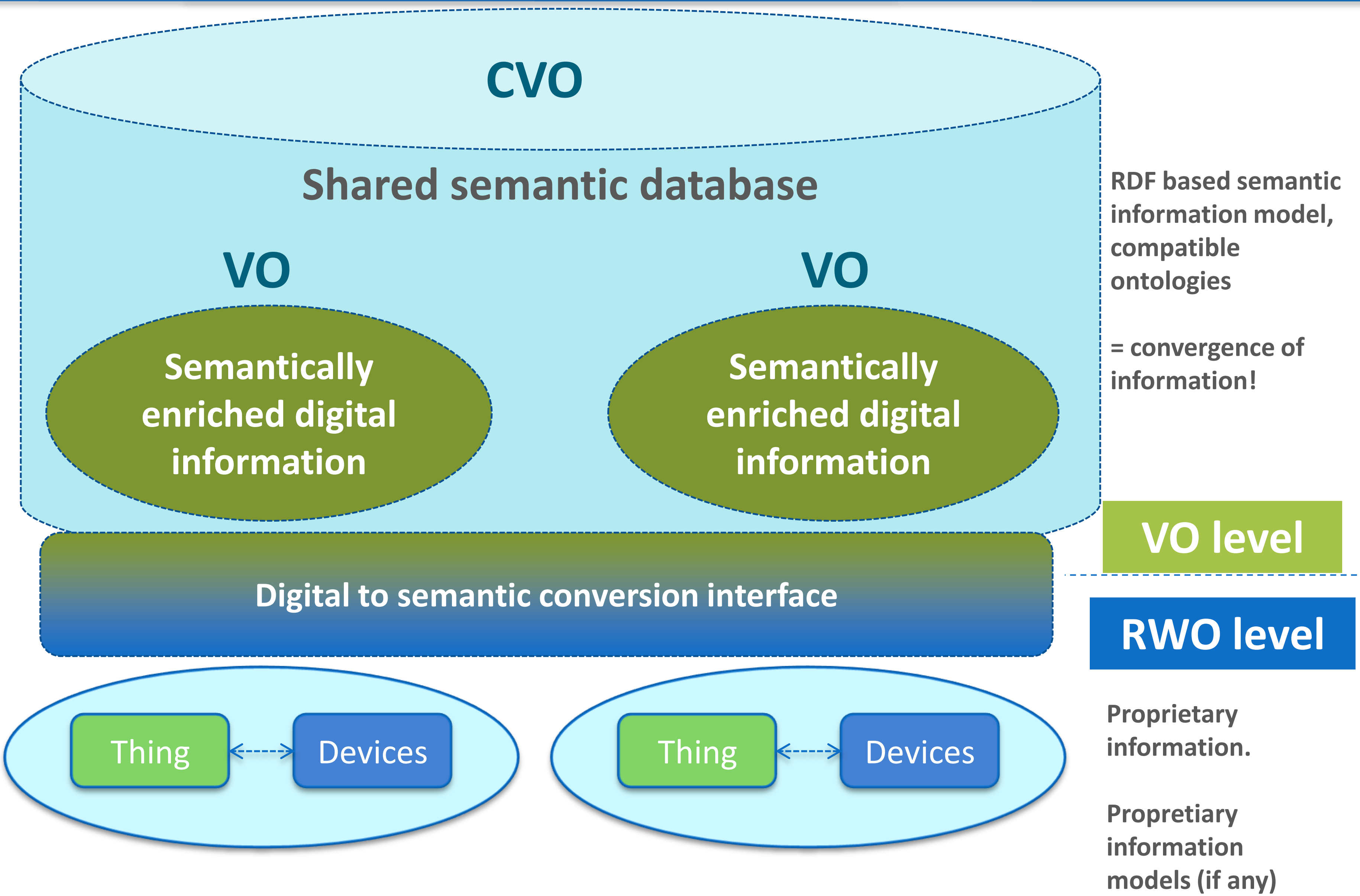
Characterization of RWO information

- Features and attributes of Non-ICT object which is observable and actuable (e.g. temperature of room)
- Functionalities and capabilities of ICT objects (e.g. smart phone sensing, actuation functionalities)
- Resources of ICT objects (computing, memory, battery, communication modules, etc.)
- Contextual information of ICT objects
- Ownership/vendor/models/ performance cost, etc.

Conceptual framework



Convergence of semantic VOs



The problem (…current status in market)

- Except for UPnP and DLNA-based media sharing, today one **MUST**:
  - Buy all the devices from one vendor, **OR**
  - Connect “smart” devices (phones, TVs) from different vendors through installing a particular software client (from one vendor) on each of them (limited list of supported platforms), **OR**
  - Use a particular gateway box, then can connect devices from different vendors (from a limited list of supported by the gateway)
- **In all three cases, a single vendor is responsible for all of the “interoperability”**

Vision (challenge current status)

- Ability to have gradually growing IoT environments, contrasted to a need to install and interconnect all IoT devices and software at once.

- Ability to interconnect devices from different vendors.

- Ability of 3rd parties to develop software applications for IoT environments, contrasted to applications coming only from the devices' vendors.
- Ability to develop applications that are generic in the sense of running on various IoT device sets (different vendors, same purpose), contrasted to developing applications for a very particular configuration of devices

In summary: ***"App store for smart environments***

e.g. smart home, smart city, smart island, …

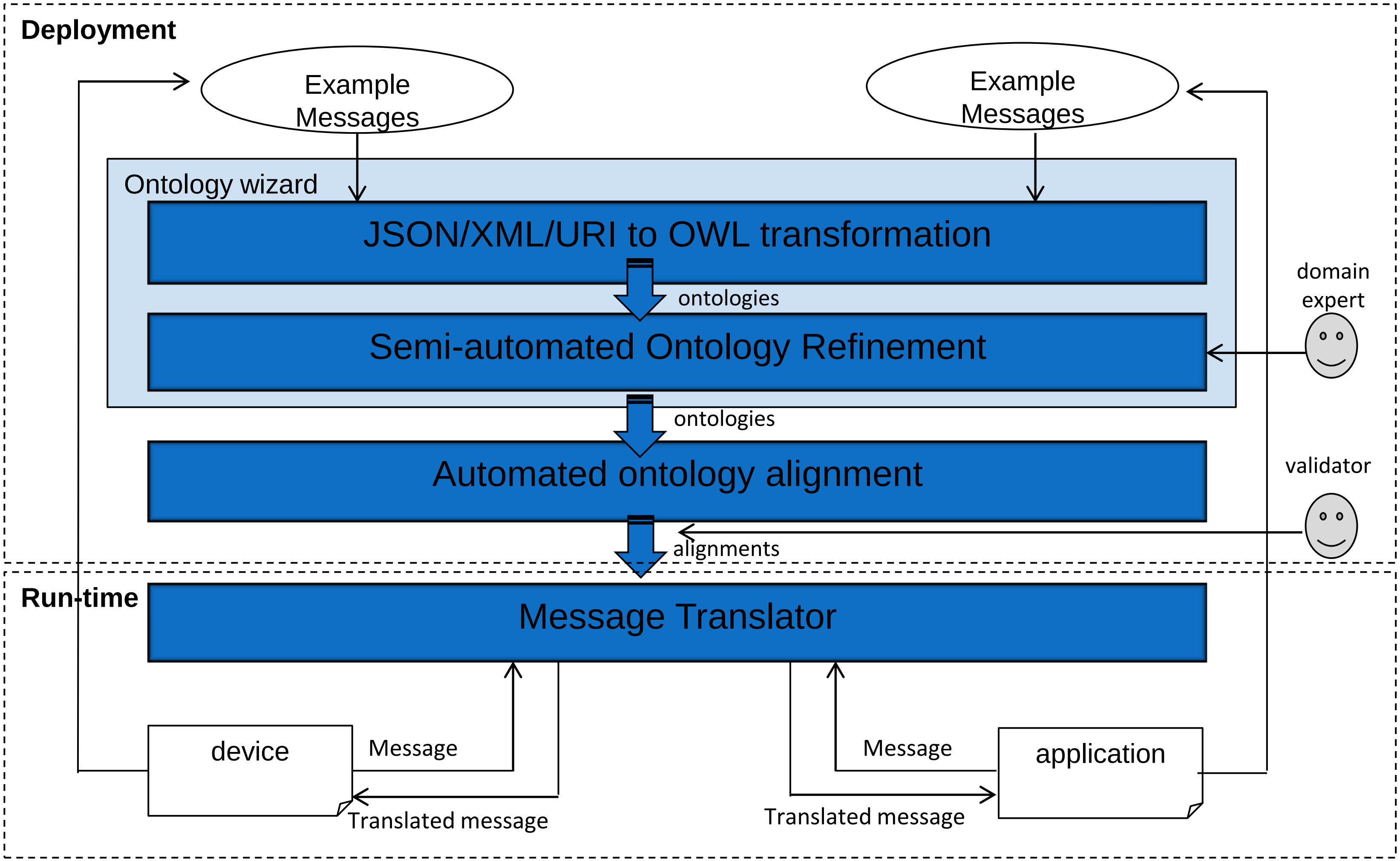
Is WoT enough to for this vision?

- Applying Web architecture to Internet of Things is a great facilitator of interoperability.
- WoT, however, is mostly about the protocols and formats.
- WoT as such will not enable realization of :
  - 2 temperature sensors both delivering measurements over HTTP GET as JSON, but of different structure and with different object/property names
  - 2 heater devices accepting commands over HTTP PUT as JSON, but of different structure and with different object/property names
  - 1 motion-detector&light-switch control software receiving measurements' data by connected devices (motion detector and switch, via a gateway) and sending commands back to them using heterogeneous vocabularies (e.g. app:Motion and dev:Movement) and syntax (XML vs JSON)
- For true interoperability, we need also the semantic interoperability, the ability of the devices to unambiguously convey the meaning of data they communicate over Web protocols

Device/application communication in WoT

- Entities (VOs/CVOs in iCore) are mainly
  - devices (e.g. sensors interconnected via ThereGate gateway)
  - applications (software running on e.g. a smart phone) that utilize devices' data e.g. temperature or movement readings
- Real smart environments in IoT
  - Different kind of (domain) devices or gateways may exist
  - More than one application that requires data from more than one device to run
- The task of 'assigning' devices to applications automatically can be performed by computing the similarity of their semantics (bridging semantic heterogeneity)

“Smart Proxy” architecture



Smart Proxies in iCore

Smart proxy: to automatically disambiguate data, messages (and their semantics) that are exchanged between digital objects and VOs, and also VOs and CVOs in order *to facilitate*:

- The **self-deployment** of CVOs in unknown environments, where VOs are semantically described with heterogeneous semantics (concerns the semantic matchmaking between CVO and VO descriptions)
- The **self-configuration** of VOs into clusters based on the discovery of similarities between registered VOs, utilizing the computations of alignments between VO semantics (concerns the semantic matchmaking between VO descriptions)
- The **self-adoption** of digital objects into iCore, by having SP translate the semantics of domain information related to the digital object into the semantics of iCore (and vise-versa). The digital object itself needs not to have iCore software installed, nor to know the syntax or the semantics used in iCore domain

(1a) Semantic and data models to be used in iCore

- Reuse existing ontologies
  - DUL (upper ontology for generic concepts)
  - SSN (W3C WG for Sensor Description)
  - IoT-A (IoT-A project IoT ontology)
  - QUDT (ontology for quantities, units, quantity values)
  - domain models e.g. SWEET-NASA
- Specify new concepts/properties for the representation of VO/CVO (profile, services, …)
  - Additional layer for integrating existing ontologies
  - High-level description, registration, discovery and access/invocation
  - Main concepts: PhysicalObject, DigitalObject, SoftAgent, Service, VirtualObject

(1b) Applications to use and/or need for semantic modeling practices (type, scale, domain)

- VO ontology – pre-designed description of VO : can be Protégé
- VO templates - pre-designed VO descriptions used to install and create VOs: can be XML
- VO registry – pre-designed description for the advertisement/discovery of VOs: Can be a semantic registry (WSDL services annotated with ontology, S-OWL, SAWSDL, …)
- VO repository: can be Sesame (graph stores for triples and data stores for big data)

(1c) Languages (Formal and non-formal), Technologies (toolkits, software tools) and protocols enabling semantic interoperability in iCore

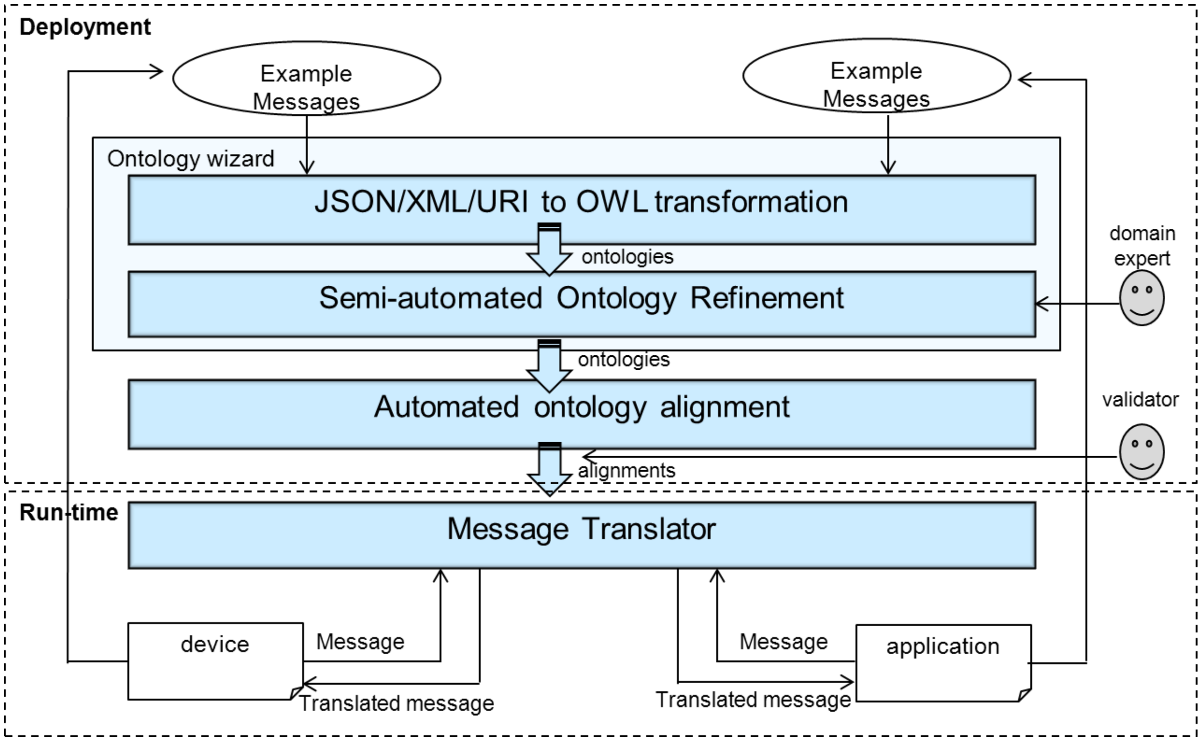
- RDF, OWL, SESAME repository
- Deployment-time:
  - Ontology learning tool (for iCore entities with no semantics attached in design-time)
  - ontology alignment tool (for iCore entities with heterogeneous semantics)
- Run-time: translation tool utilizing alignments for message communication

(1d) Possible contributions/inputs to AC4

- An elaborated/integrated IoT ontology
- Tool for ontology alignment
- A framework for semantic integration of virtual objects

(1c) + (1d) preliminary work that can be contributed to AC4

- ‘Smart proxy’ proof-of-concept for the (semi-)automated alignment of smart/control entities (VOs), meeting both syntactic and semantic interoperability of exchanged data



Smart Proxies for Self-deployment of Applications in IoT

Questions

- Thank you for your attention

Part of this work was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. This Programme is supported by the Marie Curie Co-funding of Regional, National and International Programmes (COFUND) of the European Commission.

VO definition reusing vocabularies

- VO is an entity in iCore that has part: a) 0 or more ICT objects, b) 0 or more non-ICT objects, and c) at least 1 software agent (draft definition)
- A VO **is_a** dul:Object AND

  has_part exactly 1 (ssn:SensingDevice or
  ActuatingDevice or EmbeddedDevice
  associated_with 0 or more non-Device)

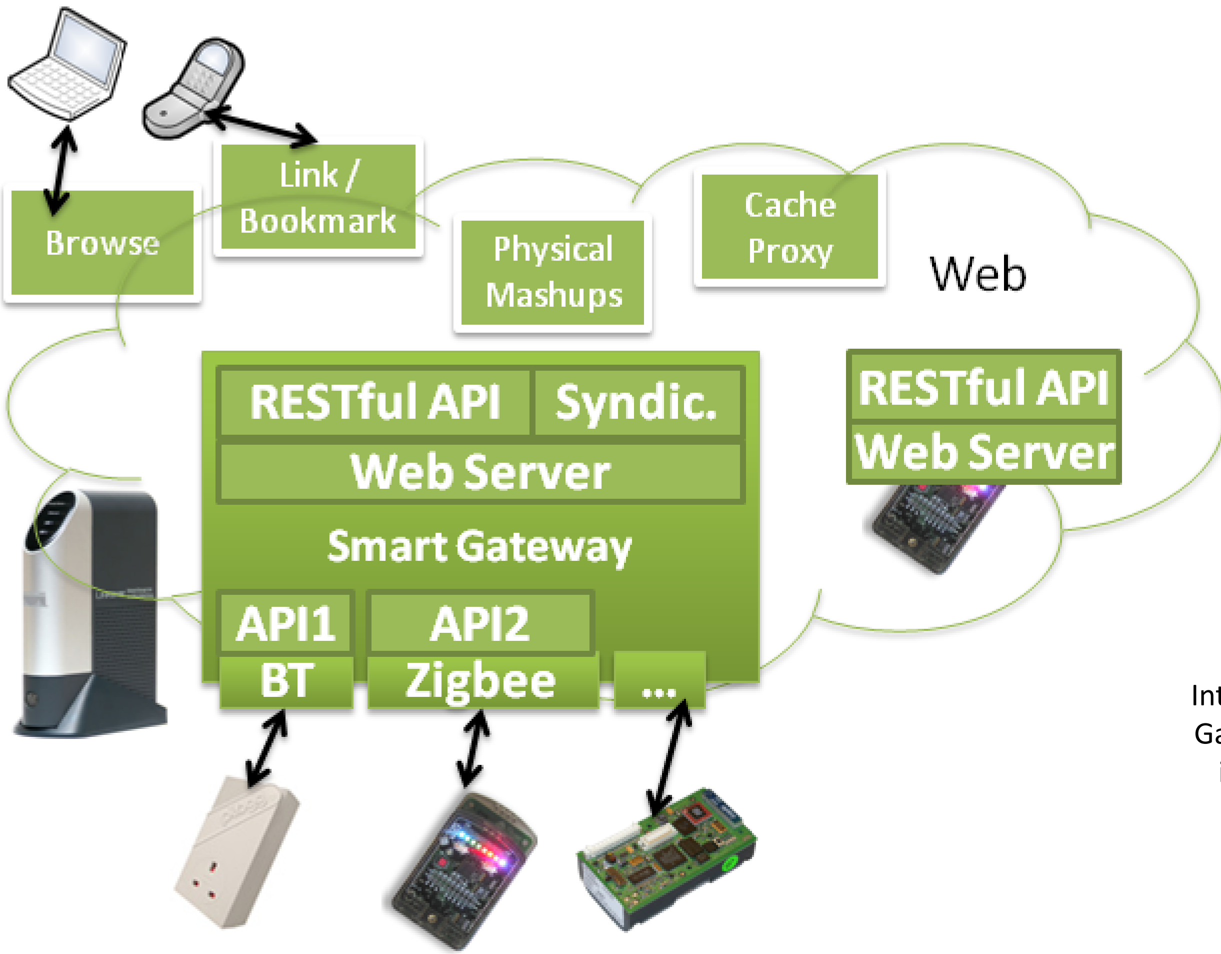
  has_part exactly 1 SoftAgent

More use cases…

- A package with an RFID or a UCODE label attached to it sent by a post office in origin-A can be automatically managed and forwarded by the intermediate destination-B to the next destination-C, without the current requirement that all post-offices (origin and destinations) share the same database or even the same semantic repository.

Integrating ‘things’ into the Web:

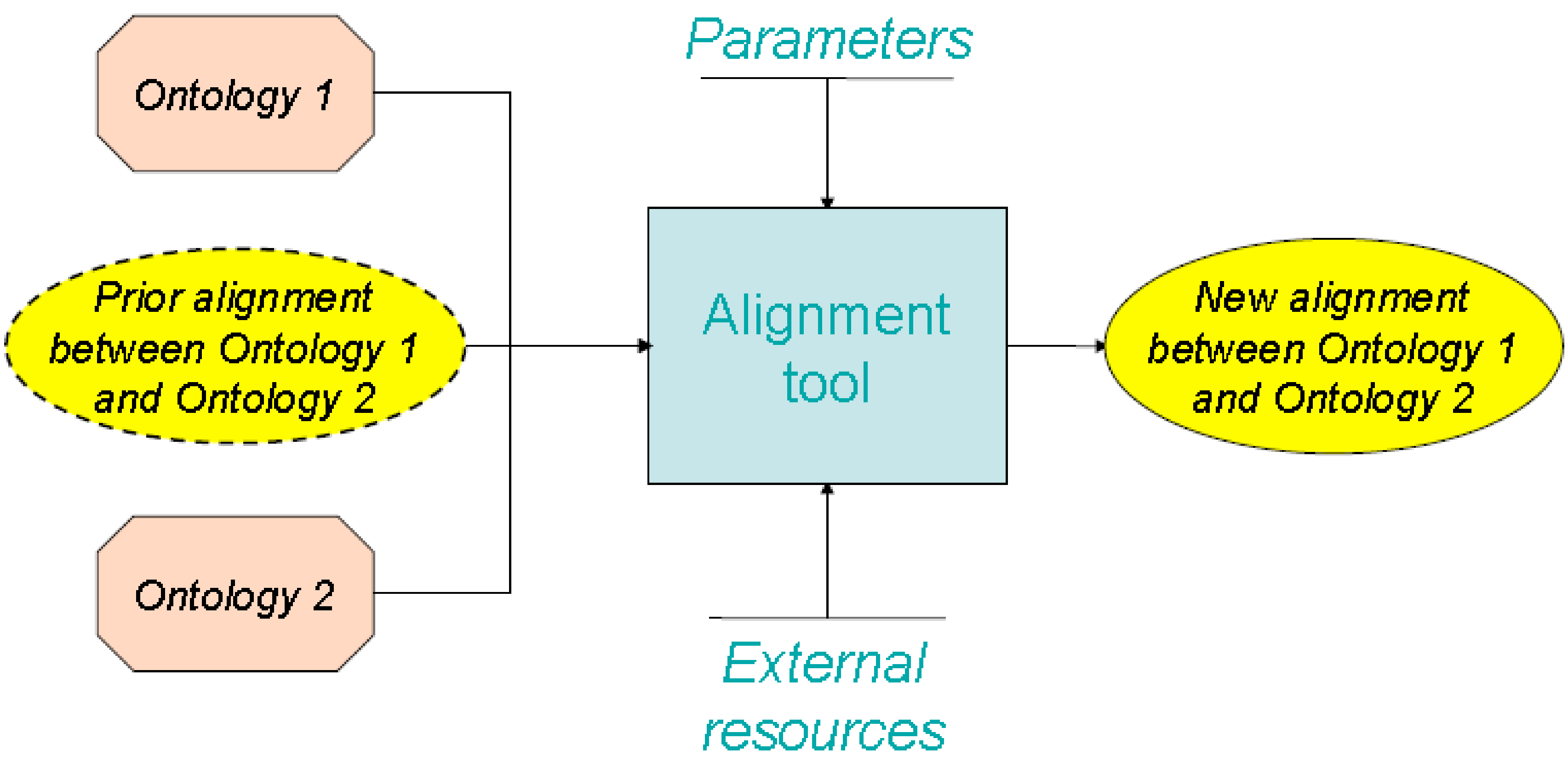
the RESTful approach



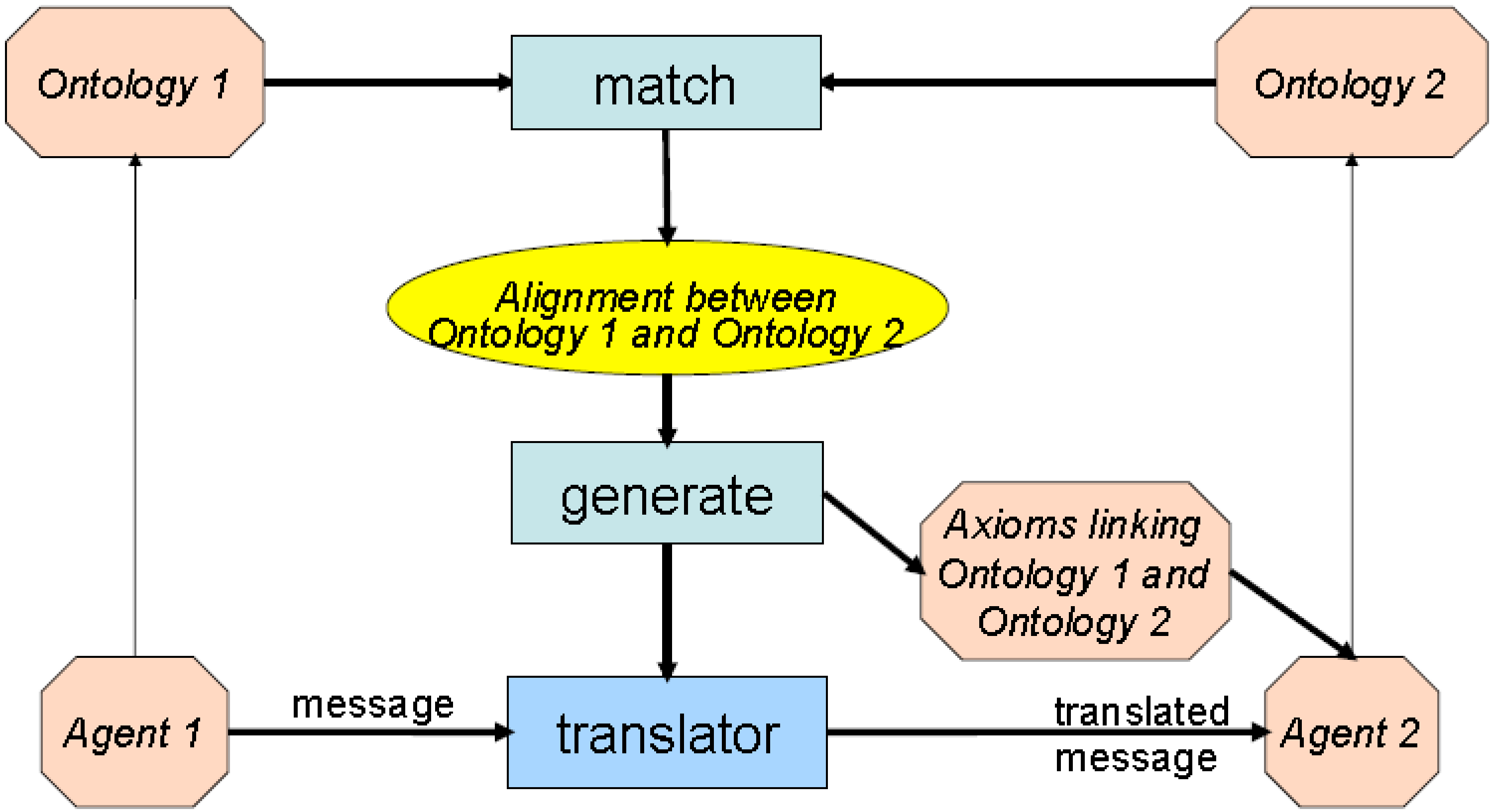
Web and Internet Integration with Smart Gateways (left), direct integration (right).

Paper of D. Guinard et al, “Architecting a Mashable Open WWW of Things”, Technical Report No. 663, Department of Computer Science, ETH Zurich, February 2010

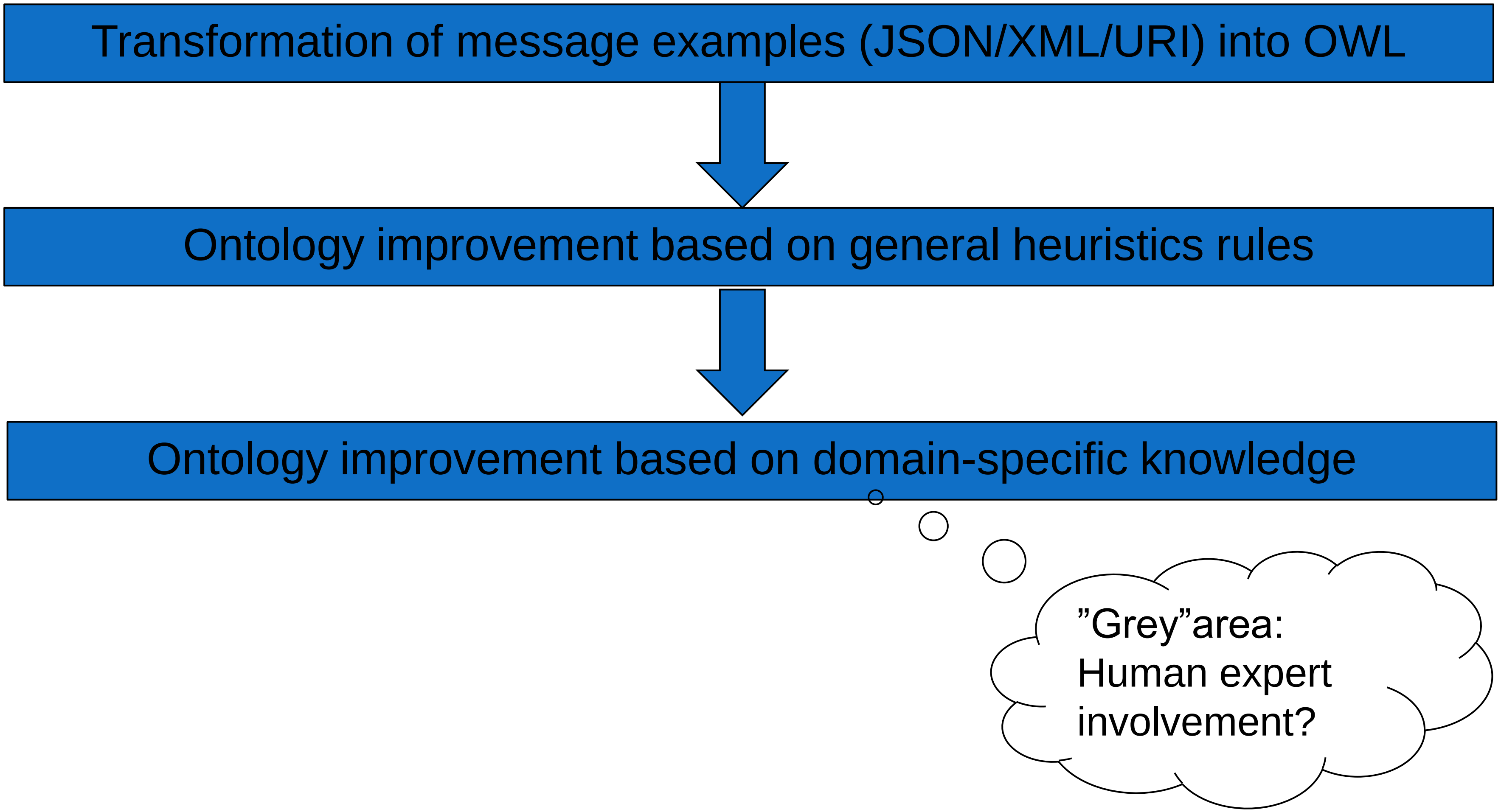
Ontology alignment at a glance



Application: agent communication

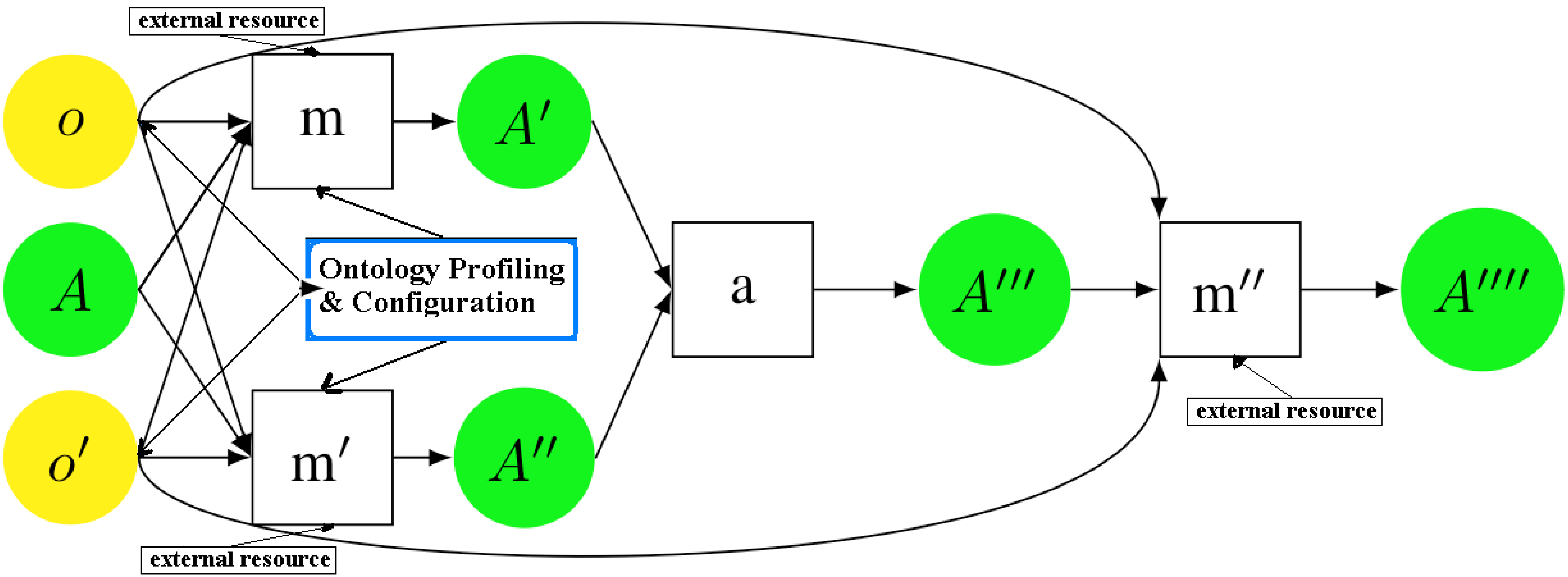


Transforming data messages to ontologies



Aligning the semantics of entities

- Aligning ontology elements (classes/properties) of 2 entities



Alignment/Matching identity

- Ontologies are lightweight flat ontology definitions, a couple of classes and a few data type properties
- **For Ontology Similarity**
  - Vector Space model (terms frequency) and cosine similarity
- **For Class/Property Similarity**
  - Use of labels (names/comments could be also used via an ontology profiling approach)
  - Identify 1:1 alignments and equivalent relation between elements (other semantic relations used in WordNet lexicon can be identified)
  - Matching cases:
    - Synonyms (movement/Motion), a WordNet-based method
    - Compound terms (e.g. SecurityToken/token, timed/time_stamp)
    - String-distances (e.g. Switch/switching/SwitchState)
    - Multilingual (non-English labels are translated to English and matched)

References for semantic interoperability

- Uschold, M. Where are the Semantics on the Semantic Web? AI Magazine, 24(3), pp 25-36. Fall 2003
- Noy, N., Doan, A., & Halevy, Y. (2005). Semantic Integration. AI Magazine. 26(1), pp. 7-10.
- Alon Y. Halevy, Why Your Data Don't Mix , ACM Queue, 3(8), 2005.
- J. Euzenat, T. Le Bach, R. Dieng, et al, Knowledge web 2.2: Heterogeneity in the semantic web Technical report, NoE Knowledge Web project, 2007.
- Shvaiko, P., Euzenat, J.: Ontology matching: state of the art and future challenges, IEEE Transactions on Knowledge and Data Engineering, 08 Dec. 2011. IEEE computer Society Digital Library. IEEE Computer Society, <http://doi.ieeecomputersociety.org/10.1109/TKDE.2011.253>