Geo-coding in Smart Environments: Smart-M3 and Geo2Tag Integration Principles

Kirill Krinkin, FRUCT LLC, ETU
Kirill Yudenok, FRUCT OSSL, ETU

ruSMART 13, Saint-Petersburg, 28-30 August 2013
Agenda

- Geo-coding (LBS) and smart spaces;
- Context and it’s location data;
- Smart-M3 and Geo2Tag platforms;
- Problems, tasks and related works;
- Geo-coded Smart Space (GCSS);
- It’s system requirements and high-level design;
- GCSS high-level requirements;
- GCSS layered architecture;
- Location based engine.
Geo-coding (LBS) and smart spaces

- **Geo-coding** allows to markup any kind of data by geographical coordinates and time (media, events, documents, etc.):
  - semantic information search;
  - M2M interactions;
- **Smart Spaces** as the basis for seamless distributed communication field for software services provides semantic level for data processing:
  - continuous distributed semantic data;
  - communication field for software services;
Context and it’s location data

"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.” [Abowd et al’99]

- One of the important part of context is location based data, that is being used for:
  - clarifying semantic meaning of queries;
  - limitation of space search;
Smart-M3 platform

**Smart-M3** is an open source software platform that aims to provide Semantic Web information sharing infrastructure between software entities and various types of devices.
Geo2Tag platform 1/2

*Geo2Tag* platform is the centralized high performance geo-tagging database with dedicated server, which is provided REST API for access to geo-tags.

- user management: registration, login, log-off, sessions;
- data retrieval about users and matching personal geographical spaces to the personal smart spaces;
- channel management: subscription/unsubscription
- sending geographical data from smart-space to the geo-tagging system;
- getting data from geo-tagging system;
- spatial filtration;
**Geo2Tag data model:**

**User** – data that represent human of Geo2Tag;

**Channel** – object that contain name, description and set of tags;

**Tag** – object that contain URL of multimedia object, name, description, time of creation and coordinates (L, B, H);
Problems and Tasks

Problems:
- investigation of the smart spaces and geo-coding platforms integration;

Tasks:
- design and development of system requirements, high-level design, architecture, data model (ontology) of the integration agent;
- integration smart system development by combining geo-coding and smart space platforms;
- testing and analysis of the system characteristics.
Related works

- *Pervasive Computing Research Group* (indoor Location Based Services and coding real world objects);

- *[K. Kolomvatsos et al’07]*: spacial ontology, ontology driven map annotation, GIS-based ontology population and navigation algorithms;

- *[A. Dearle et al’03]*: tree-based region distribution of semantic information in global space (like geographical fractal structure of smart space data);
Geo-coded Smart Space (GCSS)

**GCSS** is a smart system that looks like Smart-M3 agent (KP) by combining the work of two platforms *Smart-M3* and *Geo2Tag*.

- **Main difference** from previous works – common platform for knowledge processing of Smart-M3 space;
- **New property** – location in space and time, that give the ability to search in a space (room, house).
GCSS layered architecture  1/2

- Semantic FE
- Geo-coding FE

Integration

Domain engines

- Smart Space engine (SIB)
- Geotag Engine

Data cloud backend (optional)
Interfaces – smart-spaces and geo-coding front-ends (FE) and their functionality;

Integration – components for translating geo-tags from Geo2Tag format to Smart-Space format and vice versa;

Domain engines – particular implementations of smart-space geo-coding middleware (Smart-M3 and Geo2tag);

Data cloud backend – optional component, which providing services: offline data (pre-)processing, storage for BLOB objects, indexing, caching etc.
System requirements and high-level design

*Use-cases of the GCSS system:*
- geographical markup of smart space data;
- search set reduction;
- search context rectification.

**The main task of the agent** – the union of the Smart-M3 and Geo2Tag platforms.
- Agent is not an extension of the Smart-M3 platform, it is expanding the space with new data – geo-data;
GCSS high-level requirements

- providing interfaces for semantic data and access;
- distributed storage for semantic information;
- interfaces for association semantic objects with geo-tags;
- spatial and temporal filtration.

Non-functional requirements:

- Performance (cloud based massive offline processing and local context indexing/caching);
- Compatibility (legacy interfaces: SSAP, REST);
Location based engine

**GCSS Geo2Tag ontology:**

![Diagram](image-url)
The root element is the class **User** (all geographical data from personal geo-space);

- The **Channel** could contain any amount of geo-tags;
- The main object of this ontology is a **geo-tag** (class Tag);
  - The size of a one geo-tag nearing \( \sim 1K \);
- The property **data** plays significant role in integration mechanism, which allow to specify objects or relations in smart space.
Geo2Tag agent architecture

- Geo2Tag agent is a KP in terms of Smart-M3 platform;
Geo2Tag agent architecture details

- Main components: *Geo2Tag service handler* and *Smart-M3 handler*;

- *Geotags <-> Triples conversion handler* responsible for converting geo-data (JSON format) to the space RDF-triplets;

- Geo-tag consists of a tuple \(<t, L, B, H, data>\), which can be easily converted into a space geo-data triplet of \(<time, coordinates, data>\) type;

- Agent is used *object-oriented model* for ontological data processing (all RDF triples are transforming into objects);
Future steps

- improvement of Geo2Tag agent architecture and it’s ontology;
- Geo2Tag agent development process;

Still open questions:

- overall system performance;
- effective object monitoring;
- temporal and spatial filtration;
- integration with media objects.
Q&A

Kirill Krinkin {kirill.krinkin@fruct.org}
Kirill Yudenok {kirill.yudenok@gmail.com}
FRUCT & ETU OSLL {http://osll.fruct.org/}
Links


- K. Kolomvatsos, V. Papataxiarhis, V. Tsetsos.: “Semantic Location Based Services for Smart Spaces“, 2nd International Conference on Metadata and Semantics Research (MTSR), Corfu, Greece, 2007