RFID
Radio Frequency IDentification: Concepts, Application Domains and Implementation

Dominique Guinard, Patrik Fuhrer and Olivier Liechti
University of Fribourg, Switzerland
Submission ID: 863
Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
Introduction

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
RFID Tags

> Radio Frequency Identification (RFID) is a method for remotely storing and retrieving data using devices called tags.

> RFID tags (often presented as the new generation of barcodes) are composed of:
  - an antenna.
  - a microchip containing a small amount of data.

> RFID technology enables:
  - Contact-less identification.
  - The reading of identifiers that are in motion.
  - The detection of objects that are not in line of sight.
RFID Readers

The RFID readers (aka sensors) emit an electromagnetic field. The tag converts the field into a source of power. As the tag is powered, the sensor can start reading/writing the tag's content. Such a reading is called an RFID event and is transmitted to a computer in charge of processing it.
The EPCglobal Architecture Framework

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
Towards the Internet of Things

> World-wide standards are required for global interoperability.

> **Auto-ID Labs** and **EPCGlobal** are two non-profit organizations tackling this goal.

> Both have a simple motto: 

  developing an open standard architecture for creating a seamless global network of physical objects called the EPC Network, aka the “Internet of Things”.

> The **EPCGlobal Architecture Framework** regroups various standards:
  
  – Electronic Product Code (EPC) – March 8, 2006 (v1.3)
  – EPC Information Services (EPCIS) – April 12, 2007 (v1.0)
  – Object Naming Service (ONS) – October 4, 2005 (v1.0)
  – Application Level Events (ALE) – September 15, 2005 (v1.0)
Electronic Product Code (EPC)

35 . 0000AF8 . 00FA55 . 000015AF

- First standard supporting the “Internet of Things”: the EPC (Electronic Product Code) Tag Data Specification.
- A world-wide unique number identifying a particular instance of a physical object.
- Often represented in an URI form (Uniform Resource Identifier), e.g.:
  
  urn:epc:id:gid:2808.64085.88828
  
  - gid: header
  - 2808: manufacturer
  - 64085: product or object class
  - 88828: serial number
EPC Information Services (EPCIS)

The EPCIS are the primary vehicle for data exchange between trading partners.

The EPCIS Standard specifies two interfaces and one data model.

- **EPCIS Event Query Interface** defines how business events can be requested from repositories and other sources of EPCIS data.
- **EPCIS Event Capture Interface** specifies a standard way to communicate data to applications that wish to consume it.
- **EPCIS Data Specification** defines the format and the meaning of the exchanged information. This information takes the form of “events” describing the what, when, where and why for physical object movements.
Object Naming Service (ONS)

> A simple idea: having the EPC of an object, where can I retrieve (authoritative) data about it?

> Designed on top of DNS (Domain Name Service).
Filtering Collection Interface

> Reducing the volume of data that comes directly from EPC data sources typically involves:
  - receiving EPCs from one or more data source such as readers;
  - accumulating data over intervals of time, filtering to eliminate duplicate EPCs and EPCs that are not of interest, and counting and grouping EPCs.

> The Filtering Collection (Application Level Events or ALE) Interface provides a standardized format for reporting such collected and filtered EPC data.
> Put all together, the EPC standards converge towards a global network of the physical objects surrounding us: the EPC Network.

> The EPC Network is still young but its potential might well boost the number of adopters. Examples: Wall Mart, Metro Group, Gillette,...
Agenda

- Introduction
- The EPCglobal Architecture Framework
- Application Domains
- RFIDLocator: a Localization Framework
- Demonstration Video
- Extension Mechanism of the RFIDLocator
- Using the Framework: the Smart Badge project
- Conclusion
Matching a computer-readable standardized number to any object surrounding us has an incredible number of applications in various domains:

- **Assets Identification and Tracking**
  - Document tracking in attorney/lawyers offices

- **Supply Chain Management**
  - Wall Mart and its “Top 100 Suppliers” challenge
  - Metro Group and its “Future Store” initiative
Matching a computer-readable standardized number to any object surrounding us has an incredible number of applications in various domains:

- **Anti-Counterfeiting**
  - Pfizer to fight fake Viagra
- **E-health**
  - blood tracking
  - patient identification
  - smart operation theatres
RFIDLocator: a Localization Framework

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
RFIDLocator: a Localization Framework

> The RFIDLocator is a localization framework.
> It was developed in the Software Engineering Group of the University of Fribourg in collaboration with Sun Microsystems Switzerland.
> It supports the development of spatially aware applications within a predefined area (e.g. a building)
> Example applications:
  - Tracking assets within a building
  - Inventory systems
  - Tracking patients within a hospital
The RFID Stack

Service Layer
- Presentation
  - Added Behaviour
- Web frontend, Rich clients, Midlets,...
  - SmartBadge

Environment Solving Layer
- Solver Plugins
  - Localization Framework
- Solver1, Solver2
  - RFIDLocator
- Solver3, Solver4

Sensing Layer
- RFID Middleware
  - RFID Hardware
- event managers, information servers
  - readers, antennae, tags

JMS (application messages)
JMS (EPCIS Data Specification)
The RFIDLocator is a distributed Java Enterprise application, developed using the Enterprise JavaBeans specification.

The application is deployed on the Sun Java System Application Server.

The RFID middleware is the Sun Java System RFID Software.

Besides the software choices, standard RFID hardware was chosen.
Object Model: The Actors

- **Reader**
  Models an RFID reader (aka Sensor)

- **PhysicalAntenna**
  A hardware component able to capture RFID events.

- **LogicalAntenna**
  Groups 1..n PhysicalAntennae.

- **TraceableObject**
  Models assets equipped with an RFID tag and traced by the application.

- **LocatorObservation** and **BufferedObservation**
  Results of an RFID event.

- **Action**
  Action assigned to the RFID events: either IN or OUT of the Location.

- **User**
  Models the users and administrators of the system.
Services and Manager: The Core Classes

- The **SensorListenerMDB** is the integration point between the EPCIS and the RFIDLocator.

- The **Managers** are used to create, delete and manage the object model (actors).

- The **Solvers** are the algorithms of the framework. They decide whether an RFID event should be persisted as a business event (**LocatorObservation**).
Sequence Diagram of the Solving Process
Demonstration Video

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
Extension Mechanism of the RFIDLocator

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
Adapting the Framework to a new Standard

> Using a tool based on JAXB and Ant, the RFIDLocator can accept any kind of XML-formalized RFID event.

> The XML schema of the formalism is used to generate explicit content objects that can be accessed in a “Object Oriented fashion”.

> The RFIDLocator only needs to be able to extract three information from these content objects:
  - *What* was observed (tag(s));
  - *When* it was observed (time);
  - *By whom* it was observed (antenna(e)).
Extending the Solving Logic

To change the behavior of the core solving logic one only needs:

- To program a new **Solver** using the provided interfaces and tools.
- To have a **LogicalAntenna** referencing it.

Using this simple but yet powerful scheme we can create **Solvers** to:

- Aggregate RFID events (~business-level ALE).
- Direct business events to different outputs (JMS queues, DB, log files, method calls, etc.)
Using the Framework: the SmartBadge project

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
The Smart Badge project is a multi-tier infrastructure offering contextual services to members of an institution:

- Each actor receives an RFID enabled card.
- RFID readers are installed at points of special interest and strategic places.
- The system delivers adapted content to users when they enter or pass by monitored places within the environment.
Use Cases

> Smart Badge on a University Campus:
  - RFID tags in students/staff ID cards.
  - Services:
    ➪ Adapted meal propositions sent per text.
    ➪ Lecture slides per email.
    ➪ Jobs/student union information.
    ➪ Monitoring attendees.

> Smart Badge at a Conference:
  - RFID tags on conference badges.
  - Services:
    ➪ Next talks matching profile.
    ➪ Updates, news, social events.
    ➪ Your conference log/blog.
    ➪ Monitoring attendees.
Using the Framework: the Smart Badge Project

Architecture

- Sensing Layer
  - Raw data / Antenna
  - RFID Hardware
  - EPCIS
  - RIED

- Environment-solving Layer
  - Logical Antenna / Location
  - RFID Locator
  - PB
  - JMS

- Service Layer
  - Service Points / Transmitters
  - Smart Server
  - MySQL
  - Service Delivery layer | Service Management layer

- Services and users
  - Smart Web Server
Contextual Service Providing: Architecture
Managing Contextual Services

> Smart Web Server
> Web front-end to manage services, service points and users.
> Users can access it to manage their profiles and preferences (e.g. diets, preferences for mobile device, affiliation, etc.)
> PHP & MySQL application
Managing Contextual Services
Conclusion

Agenda

> Introduction
> The EPCglobal Architecture Framework
> Application Domains
> RFIDLocator: a Localization Framework
> Demonstration Video
> Extension Mechanism of the RFIDLocator
> Using the Framework: the Smart Badge project
> Conclusion
Enhancements

> The RFIDLocator is a working prototype, not a fully-featured system.
> Making it available for commercial purposes would require:
  – A thinner granularity for the user management (roles, restricted access to the TraceableObjects, etc.).
  – A more complete administrative GUI (modifying the reader's configuration, etc...).
> Update to new standards and middleware.
Achievements

> Integration of various standards of the RFID field.
> Built a realistic distributed application involving many different hardware devices.
> Performing distributed application enabled by the use of Java Enterprise Framework and its EJB components:
  – scalable, robust, reliable, transaction control.
> The RFIDLocator framework has a clean, flexible and well documented software architecture:
  – Validated by the implementation of the SmartBadge.
  – Allowing further extends and uses of this open-source framework.
Conclusion

Future of RFID

> EPC Network standards presents a huge potential in term of cross organizations interoperability.

> Price/performance ratio is improving encouraging global adoption.

> Application domains do not cease to widen as technology evolves:
  – Cellular phones (Nokia, etc.) to include RFID readers for NFCs applications.
  – Underskin RFID implants for humans allow new use-cases.

> Privacy and security concerns should not be underestimated. They have to be solved at both technological and legal levels (e.g. EU e-privacy directive).
Resources

> RFIDLocator's official website: http://diuf.unifr.ch/rfid
  – Java API
  – Application source code and binaries (under GPL)
  – Documentation (under FDL)
  – Related publications

> Other URLs:
  – Department of Informatics of University of Fribourg
    http://diuf.unifr.ch
  – Software Engineering Group
    http://diuf.unifr.ch/softeng