



Towards Unified Tag Data Translation for the Internet of Things

Loïc Schmidt, Nathalie Mitton, David Simplot-Ryl

► **To cite this version:**

Loïc Schmidt, Nathalie Mitton, David Simplot-Ryl. Towards Unified Tag Data Translation for the Internet of Things. Wireless VITAE'09, May 2009, Aalborg, Denmark. 2009. <inria-00388022>

HAL Id: inria-00388022

<https://hal.inria.fr/inria-00388022>

Submitted on 26 May 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Unified Tag Data Translation for the Internet of Things

Loïc Schmidt, Nathalie Mitton, David Simplot-Ryl
INRIA Lille-Nord Europe, CNRS UMR 8022, University of Lille 1
Villeneuve d'Ascq
France
Email: {loic.schmidt,nathalie.mitton,david.simplot-ryl}@inria.fr

Abstract—Following the “Internet of Things” concept, each object will be associated with a unique identifier which will allow to retrieve information about it in large databases. In the process of retrieving information, this identifier (ID) may have to be translated into different formats (e.g. domain name style format for object name service query, binary, legacy,...). The Tag Data Translation (TDT) is responsible for the translation of IDs into these different formats.

We propose a general TDT system which extends the standards of EPCglobal which only targets Electronic Product Code (EPC). We integrate other RFID and smart cards standards (such as ISO 14443 and 15693) and GS1 standards which are more general as they also deal with bar code (EAN/UPC).

I. INTRODUCTION

The “Internet of Things” aims at creating a large wireless network in which all objects would have a unique identifier. This concept is attributed to the MIT Auto-ID Center, founded in 1999 [11].

This concept goes along with Radio Frequency IDentification technology (RFID). A RFID tag can be placed on all objects, offering a way to question them and know their identity. Using this ID, an efficient object name service (ONS) and shared databases, we retrieve information at anytime. The Auto-ID Center defines, with partners, several standards for the Internet of things. These standards can be found under the name of EPCglobal Network [1]. The Auto-ID Center is now known under the name of Auto-ID Labs.

In such a network architecture, we have to ask different components to retrieve information on an ID, from the tag to the application and vice versa. For this, the ID of an object must be translated into different representations, like defined in the EPC tag data translation [4][5]. In the EPC global architecture, the tag data translation process (TDT) converts one representation of an Electronic Product Code (EPC) into another representation, which can be used for ONS or EPCIS query, *etc.*

EPCglobal and Auto-ID Labs have defined the TDT for EPC RFID tags only, leaving aside other standards in RFID or smart cards [8][9] and other technologies such as GS1 bar-code standards (EAN/UPC) which are currently the predominant technology for identifying items [10]. In the current economic context where most of items are identified by a GS1 bar code and where RFID appears to be the new standard in

such applications, the needs of an architecture which deals with both RFID and bar code technologies seems necessary. Furthermore, identify and authentify people with smart cards is also widely spreaded (credit cards, discount cards, SIM cards, *etc*) and encompass such a technology with RFID and bar codes seems important in order to define a general unified architecture.

In this paper, we propose a way of integrating all these technologies in a more general TDT.

In Section II, we briefly present the EPCglobal Network architecture, and the goal of the EPC TDT. Section III gives an overview of other standards (ISO 14443, ISO 15693 or GS1 system). The way these new standards are added in our TDT is shown in Section IV and evaluated in Section V. Finally, we conclude with future works.

II. CONTEXT

This section describes the EPCglobal architecture [2] and the EPC Tag Data Translation (TDT).

A. RFID systems : EPC global example

The MIT Auto-ID Center was created to develop a global unified item identification system based on RFID to replace the UPC bar code. This center has been replaced by two organizations, the Auto-ID Labs and EPCglobal. The first one is responsible for the development of EPC technology while the second one manages of the EPC Network.

The EPC Network defines standards going from tag data to Application Level Event (ALE) in its architecture framework. This framework is depicted on Figure 1. This ALE is used by clients to obtain EPC data from sources.

At the lowest level is the Tag Data Standard (TDS) defined by EPCglobal. This data is retrieved thanks to a low-level reader protocol (LLRP [3]) which communicates with tags using the Tag Protocol. A standard for reader management is also defined. All these standards aim to unify the way of identifying uniquely items and manage compatible readers.

EPCglobal defines an interface for EPC Information Service (EPCIS [6]) in order to share EPC-related data in and between enterprises. This standard includes EPCIS Data Specification providing definitions for all types of EPCIS data, and EPCIS

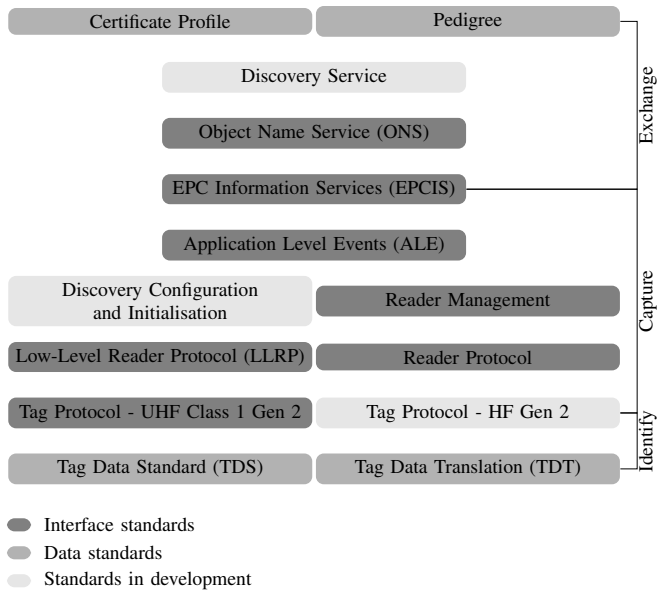


Fig. 1. EPCglobal Architecture Framework

Query Interfaces defining the way for querying and delivering data from EPCIS.

Another standard is used for retrieving EPC-related data. The object name service (ONS [7]) provides a way of finding an EPCIS which contains the needed data.

Each of these standards uses its own format. Indeed, read data from tags can be used with ALE, EPCIS, legacy applications, ONS or for writing informations in tags. Nevertheless, the product code must be translated into different formats to be used in each of these scenarii. The tag data translation performs this task.

B. EPC Tag Data Translation (TDT)

In the EPCglobal Network, an EPC can have various representation formats to be used with other EPCglobal standards. The TDT is a tool which aims to convert EPC into several representations. In the specification of the EPCglobal TDT, five different formats are defined. An example for each of these formats is shown in Figure 2.

Representation	Value
TAG-ENCODING URI	urn:epc:tag:sgtin-64:1.0037000.030241.1041970
PURE-IDENTITY URI	urn:epc:id:sgtin:0037000.030241.1041970
ONS HOSTNAME	030241.0037000.sgtin.onsepc.com
LEGACY	gtin=00037000302414;serial=1041970
BINARY	10001000000000000001000001110110 001000010000011111110011000110010

Fig. 2. EPCglobal Tag Data representation

The tag-encoding URI representation of an EPC is used with the Application Level Event. The pure-identity URI is defined for EPCIS. URI stands for Uniform Resource Identifier. In EPC TDS, they are represented as Uniform Resource Names (URNs) with “epc” as namespace. The EPC must be in its ONS hostname representation to perform an ONS query. The

legacy application prefers to use the legacy representation of an EPC. At last, the binary format can be used for communicating with the reader (for the writing action). The translation between all these formats can be performed at any level of the architecture, so the TDT is a very important tool in the EPCglobal Network.

III. OTHER INPUT STANDARDS

In this section we present the other standards we want to integrate into our TDT.

The EPCglobal TDT standard defines a way of converting EPC only. We can also add the bar codes IDs in the process, as they are still widely used for identifying class of items. Indeed, GS1 bar codes standards are the most used identifier system in supply chains. In fact, EPC are identity types derived from GS1 system as illustrated in Figure 3. There is a lot of codes in GS1 system and EPC TDS used only a few of them.

EPC Type	Related GS1 Type (Application Identifier)
GID	
SGTIN	GTIN with added serial number (01 + 21)
SSCC	SSCC (00)
SGLN	GLN with added serial number (254 + 21)
GRAI	GRAI (8003)
GIAI	GIAI (8004)
DoD	

Fig. 3. Relation between GS1 types and EPC types

Application Identifiers (AI) are used to identify the role of the code, like in EPC TDS. GTIN are used to identify items classes, *etc.* There are approximatively 100 AIs in the GS1 General Specifications [10]. Even if EPCglobal uses only seven of them, it seems useful to implement all the GS1 standards. Figure 3 shows that with some GS1 AIs, we can have a complete EPC. Nevertheless, not all GS1 bar code types can carry all GS1 AIs. The EAN/UPC or ITF-14 bar code types can only carry GTINs (GTIN-8, GTIN-13 or GTIN-14). GS1-128, GS1 DataBar and GS1 DataMatrix can carry all AIs.

There are several other standards used in RFID and smart cards such as ISO 14443, 15693, *etc.* In ISO 14443, the unique identifier (UID) consists of 4, 7 or 10 bytes for respectively single size UID (Figure 4) or double and triple size UID (Figure 5).

uid0	Description
'08'	uid1 to uid3 is a random number dynamically generated
'x0' - 'x7'	proprietary fixed number
'x9' - 'xE'	
'18' - 'F8'	reserved for futur use
'xF'	

uidn : n^{th} byte of the UID

Fig. 4. Single size UIDs in ISO 14443

The UID in ISO 15693 is 64 bits long (see Figure 6).

These other standards have to be taken into account in order to offer a complete solution. In the next section, we will show how we include them into the TDT process.

uid0	Description
Manufacturer ID according to ISO/IEC 7816-6/AMI	Each manufacturer is responsible for the uniqueness of the value of the other bytes of the unique number.

uid_n : n^{th} byte of the UID

Fig. 5. Double or triple size UIDs in ISO 14443

MSB		LSB	
64	57	56	48
‘E0’		Manufacturer ID according to ISO/IEC 7816-6/AMI	Unique serial number assigned by the IC manufacturer

MSB : most significant bit

LSB : less significant bit

Fig. 6. UID format in ISO 15693

IV. ADVANCED TAG DATA TRANSLATION

In this section we propose an Advanced Tag Data Translation. The aim is to add some of the standards presented in Section III in the TDT process to offer the possibility to use them in RFID system. We choose to add GS1 codes used in bar codes and some ISO standards. Our TDT needs a parameter “dataType” to know if the input is either GS1/EPC or ISO.

A. GS1 code

The first step of the TDT process for GS1 bar codes is to know what kind of bar code is used in input. The GS1 symbology identifiers (SI) offer a way of retrieving this information as shown in Figure 7. Each reader has the ability to know the SI of a readed bar code. Even if all readers can not send this SI with the data message, we assume that the application which is using the TDT knows the type of the readed bar code so it can send it to the TDT with the code.

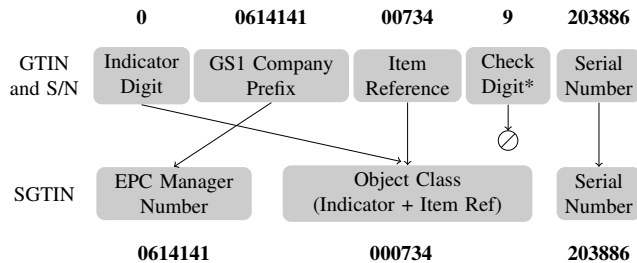
SI	bar code type
JE0	EAN-13, UPC-A or UPC-E
JE1	Two-digit Add-On Symbol
JE2	Five-digit Add-On Symbol
JE3	EAN-13, UPC-A or UPC-E with Add-On Symbol
JE4	EAN-8
JI1	ITF-14
JC1	GS1-128
je0	GS1 DataBar
jd2	DataMatrix

Fig. 7. GS1 system Symbology Identifiers

Concerning the GS1 bar codes, we have added a new format which is called “GS1_AI_IDENTIFIER”. This format starts with the three-characters data string of the SI, and is followed by the code. There are two types of code: (i) without AIs, so the code is a GTIN; (ii) with AIs in parenthesis (e.g. *JC1(01)00012345678905(21)12345678* represents a GTIN and a serial number in a GS1-128 bar code). All GS1 bar codes can be converted in this new format and in the “LEGACY” outbound format.

In the previous section, we have seen that some GS1 bar code types (GS1-128, GS1 DataBar and GS1 DataMatrix) can carry a complete EPC. The figure 8 shows how to map a GTIN with a serial number (AI 01 and 21) into a SGTIN. We call

these bar codes “EPC compliant” as they can be converted in all EPC TDS representation. Our TDT can also convert all EPC in the new “GS1_AI_IDENTIFIER” (with AIs).



*Bar code’s check digit is not stored in EPC tag

Fig. 8. From GTIN plus a serial number to SGTIN

All SIs and AIs are stored in XML files, so adding a new one is very easy.

B. ISO standards

Our TDT can translate ISO 14443 and 15693 ID into various representations. These standards are used in RFID, smart cards and NFC. Representations should have the same structure as EPC TDS ones in order to be used in an unique and unified “EPCglobal Network”-like architecture. This way IDs can be translated in BINARY, LEGACY, PURE_IDENTITY, TAG_ENCODING and ONS_HOSTNAME. These formats are shown in Figure 9.

Representation	Value
TAG-ENCODING URI	urn:iso:tag:15693-64:98.104197
PURE-IDENTITY URI	urn:iso:id:15693:98.104197
ONS_HOSTNAME	104197.98.15693.onsiso.com
LEGACY	iso15693;mfgcode=98;serial=104197
BINARY	111100000110001000000000000000 0000000000000000000011001011100000101

Fig. 9. ISO 15693 Tag Data representation

C. TDT Engine

The TDT takes the ID as an input, encoded in any format, and a variable number of parameters depending on each other: (i) desired output format; (ii) input data type (GS1 or ISO); (iii) GS1 SI and code length (for output format GS1_AI_ENCODING); (iv) tag length, company prefix length and filter (for some input format). Figure 10 shows the TDT behavior. We use the Fosstrak implementation of the EPC TDT as core of EPC translation.

If input data type is ISO, the ISO engine is chosen to perform the translation. It is more complicated when the input data type is GS1, because of the need of cross-engine for “EPC compliant” bar-codes. If the TDT has to translate an EPC in the GS1_AI_ENCODING, the Fosstrak TDT engine must translate it to LEGACY before relaying it to GS1 engine (the process is inverted in the case of GS1_AI_ENCODING input data type that have to be translated to an EPC TDS output format).

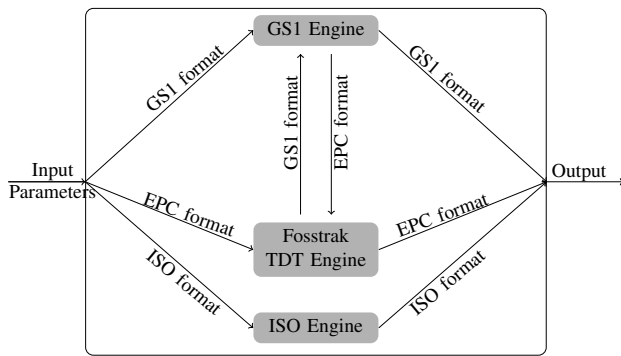


Fig. 10. TDT Engine

V. RESULTS

Starting from the Fosstrak implementation of the TDT, which has for contributors some people from Auto-ID Lab and ETH Zurich, we have added the previous standards. We can now compare initialisation and execution time without and with our “add-on”.

At initialisation, the Fosstrak TDT is ready to translate EPC in 1454ms (average from 100 measures) while our TDT takes 1751ms. This is due to the loading of the two XML files needed for the GS1 bar codes translation.

But at run time, performance in translating EPC are very similar (translation of 10000 EPC) : 564ms for Fosstrak TDT against 624ms for our TDT (this average has been calculated from 100 times 10000 conversions)

Even if we cannot compare it with Fosstrak, it is interesting to know translating time for GS1 and ISO added codes. The GS1 bar code take 10ms to be translated from “GS1_AI_ENCODING” to “LEGACY” 10000 times. Our TDT translate 10000 EPC in 1168ms from “BINARY” to “GS1_AI_ENCODING”. ISO 14443 and 15693 have similar performance in translating (161ms for 10000 translations).

VI. CONCLUSION

In the scope of the Internet of Things, where all objects is carrying an unique identifier, standards have to be defined in order to retrieve informations about objects all over the world. Auto-ID Labs and EPCglobal Inc. defines standards for such kind of infrastructure. They define a way to encode ID in RFID, to query database for informations or to query object name service (ONS) to retrieve useful database, *etc* by translating ID into differents representation matching the needs of each component of the architecture. This is the EPC tag data translation (TDT). The problem here is that these standards use only EPC tag data standard as the unique ID definition, but there are a lot of other standards. The GS1 bar-code system (with well-known EAN/UPC) is actually the most used standard in items identification applications. There are also ISO standards which define a way to encode an unique ID.

We have shown how we increase TDT process to encompass EPCs, GS1 bar-codes and some ISO standards in order to

create an unified TDT. The performance of this unified TDT is lightly under the Fosstrak implementation of the EPC TDT. Future work would be to add more existing standards (ISO 15963, other bar-code, *etc*). This TDT would offer a way to an unified architecture of a RFID middleware encompassing all useful standards.

ACKNOWLEDGMENT

This work is partly supported by FP7 Aspire European Project [12] and the french project ICOM from the PICOM [13] (Pôle de compétitivité des Industries du COMmerce).

The unified Tag Data Translation is under LGPL 2.1 license and the source code can be downloaded at “svn checkout –username anonymous <https://forge.lifl.fr/schmidt/svn/tdt>” (password : anonymous).

REFERENCES

- [1] EPCGlobal Inc., <http://www.epcglobal.org>
- [2] EPCGlobal Inc., *EPCglobal Architecture Framework (1.2)*
- [3] EPCGlobal Inc., *Low Level Reader Protocol (1.0.1)*
- [4] EPCGlobal Inc., *EPCglobal Tag Data Standards Version 1.4*
- [5] EPCGlobal Inc., *EPCglobal Tag Data Translation (1.0)*
- [6] EPCGlobal Inc., *EPC Information Services (1.0.1)*
- [7] EPCGlobal Inc., *EPCglobal Object Name Service (1.0.1)*
- [8] ISO/IEC, *ISO/IEC FCD 14443-3:1999*
- [9] ISO/IEC, *ISO/IEC FCD 15693:2000*
- [10] GS1, *GS1 General Specifications v0.8i2*
- [11] Auto-ID Labs, <http://www.autoidlabs.org>
- [12] Aspire European Project, <http://www.fp7-aspire.eu/>
- [13] Pôle de compétitivité des Industries du COMmerce, <http://www.picom.fr/>