Aerospace ID

AUTO-ID LABS

Guidelines for Lifecycle ID & Data Management

Mark Harrison,

Auto-ID Lab, University of Cambridge, UK

Report Abstract: This paper summarizes the research activity on Lifecycle ID & Data Management during the course of the Aerospace ID Technologies programme. An update on recent activities regarding messaging for maintenance events is included, together with an outlook envisaging how existing messaging systems within the air transport industry can be merged with cross-sector information retrieval approaches, such as the EPC Network.

Introduction

For several years, the aerospace sector has already been using a number of automatic identification technologies for marking of aircraft parts with a unique identifier for each part. Previously, human-readable nameplates, linear barcodes and also two-dimensional Data Matrix barcodes have been used. More recently, many organizations are considering how to migrate to Radio-Frequency Identification (RFID) for tracking aircraft parts.

In comparison with name plates, linear and two-dimensional barcodes, RFID offers a number of significant advantages:

- The ability to read a unique ID without line-of-sight to the tag (useful if the part is obscured by a panel)
- Long read ranges (of the order of 5 metres, although this may be less, depending on environmental conditions)
- The ability to read a tag without first finding its exact location on a part.
- The ability to quickly read multiple objects, without needing to scan each one manually (useful for checking the presence of safety equipment and counting items)
- The possibility to store significant amounts of data – not just a unique ID
- The possibility to write data updates back to the tag
- The fact that the physical size of the tag does not need to increase proportionally to the desired read range nor with the amount of data to be stored.
- The ability to read tags in dirty environments, where optical marks such as barcodes and DataMatrix symbols might be obscured

These advantages have led a number of aerospace companies to not only consider RFID as a technology to supplement nameplates and barcodes – but has also triggered a renewed interest in electronic collection and exchange of a larger data set for each part, including the following:

- Unique Identifier for part
- 'Birth Record' data – data fields known at the time of manufacture
- Maintenance event data – information about significant removals, installations and exchanges of parts – and the reasons for doing so.
- Information about 'No Fault Found' occurrences
- Mechanics' comments about a part

Aerospace ID

AUTO–ID LABS

Unique Identifiers for aircraft parts

In the simplest case, it is sufficient to store a unique identifier on the RFID tag and to access additional information via the network, making use of caching and pre-positioning of data where appropriate, to improve performance. Indeed, for aircraft parts that are traditionally non-serialized (e.g. lifejackets and oxygen masks), where the main objective is to check that the required number of objects is present on board an aircraft and in the correct locations, it may even be possible to use the low-cost low-memory EPC RFID tags that are currently being used by the retail & consumer goods sectors. The technical proposal for the 'UID Construct 3' identifier explains how these identifiers may be encoded onto such tags. Note that for RFID purposes, it would be necessary to allocate a unique serial number within the CAGE code to each part – although this could simply be an all-numeric serial number, which requires fewer bits – so the whole unique ID could then fit within an existing 96-bit tag that is already being used by other industry sectors.

The Aerospace ID technologies programme has worked closely with the Air Transport Association (ATA) and its members to identify user requirements and develop a technical proposal for how existing identifiers used within the aerospace and defence industries can be represented as Electronic Product Codes (EPC)[1] and encoded in a compact binary format for use in RFID tags. This work is detailed in a previous white paper[2] – but in summary, three unique identifier (UID) constructs are proposed, as shown in Figure 1.

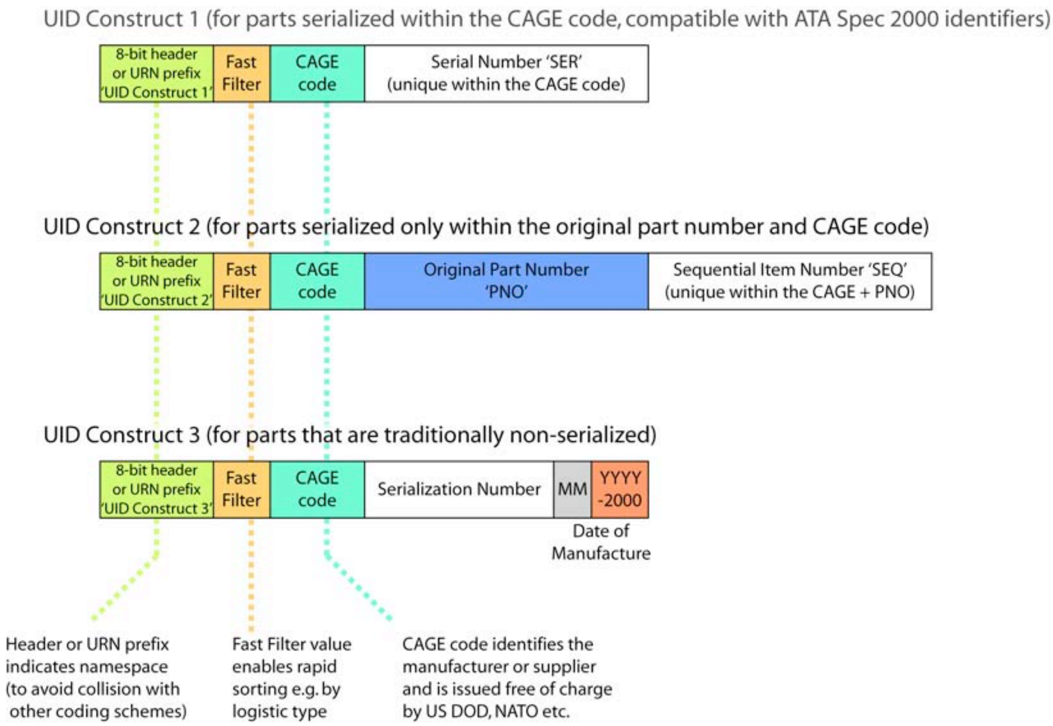

Figure 1. Three unique identifier (UID) constructs proposed for the aerospace sector, for use with EPC-compatible RFID tags.

The technical proposal describes the encoding and decoding rules and formats, aligned with how these are described in EPCglobal Tag Data Standards[3].

More recently, we have also developed Tag Data Translation[4] 'definition files', which are a machine-readable representation of these encoding and decoding rules. Additionally, Tag Data Translation software we have developed has been contributed to the Accada open source project[5] – and user-friendly translation software will be released soon and contributed to the Accada project. If the technical proposal is supported by EPCglobal with only minor modifications, this software will allow the aerospace community to easily encode and decode in a consistent, reliable manner the unique identifier stored on RFID tags for aircraft parts.

The current status is that the technical proposal has been submitted to EPCglobal's Aerospace & Defence Business Action Group – and awaiting their consideration. Following that step, the technical work group on Tag Data Translation & Standards will consider it and if appropriate, allocate three 8-bit header codes, one for each of the constructs. When these are known, the translation software can be used by all members of the aerospace sector. (At present, the header values in the software are unspecified and indicated as 'xxxxxxxx' rather than presuming any specific binary values).

It should be noted that the Tag Data Translation software described above is only intended to handle translation of the unique identifier. A new white paper by Suzuki will describe a complementary approach for handling the other data fields that might be stored on a tag and translating between an XML format for messaging and an efficient compact binary format on the tag.

Choice of RFID Tags – Capacity and Complexity

High-capacity RFID tags offer not only read/write capabilities but also enough storage to hold a moderate amount of data about each part, although for practical purposes, this is still constrained by the following factors:

- Cost of additional memory on the tag's integrated circuit
- Increased power consumption of the additional memory
- Data transfer times limited by maximum allowed data transfer rates (currently 640 kbps maximum for the UHF Class 1 Gen 2 air protocol [6]).

To some extent, high-capacity RFID tags are considered as portable miniature databases that travel with the part, enabling data about the part to be read anywhere in the world, so long as a suitable RFID reader device is available; for data that is read directly from the part's tag, it is not always necessary to have a live connection to a computer network to retrieve the data. However, we should not be under any illusions that an RFID tag is a USB memory stick with a WiFi communication interface – the memory capacity is much lower (8 kbytes compared with typically 1-4 Gbytes), the data rates are much slower (640 kbits/s compared with 11Mbits/s or 54Mbits/s for IEEE 802.11b/g respectively), the cost per Mbit is much higher (~

$1000/Mbit vs $0.001/Mbit for USB flash memory) – and most RFID tags typically do not support a sophisticated file system – addressing of memory is still handled at a very low level of programming individual bits.

Furthermore, if one were to rely solely on writing data to an RFID tag, that data would only be available within a read zone of a few metres surrounding the part – and would not necessarily be readily available to other interested parties who may benefit from access to this data.

To gain the maximum benefit from collection and sharing of some information, it is therefore not sufficient to simply write records to an RFID tag and read them upon receipt of a tagged part. Whenever data is collected about an activity concerning a particular part, it is perhaps wiser to ensure that the data is recorded in an electronic format to a temporary memory cache (such as a flash memory card within a handheld reader device), so that the data can be promptly synchronized to the network. This has been discussed previously and is illustrated in Figure 2.

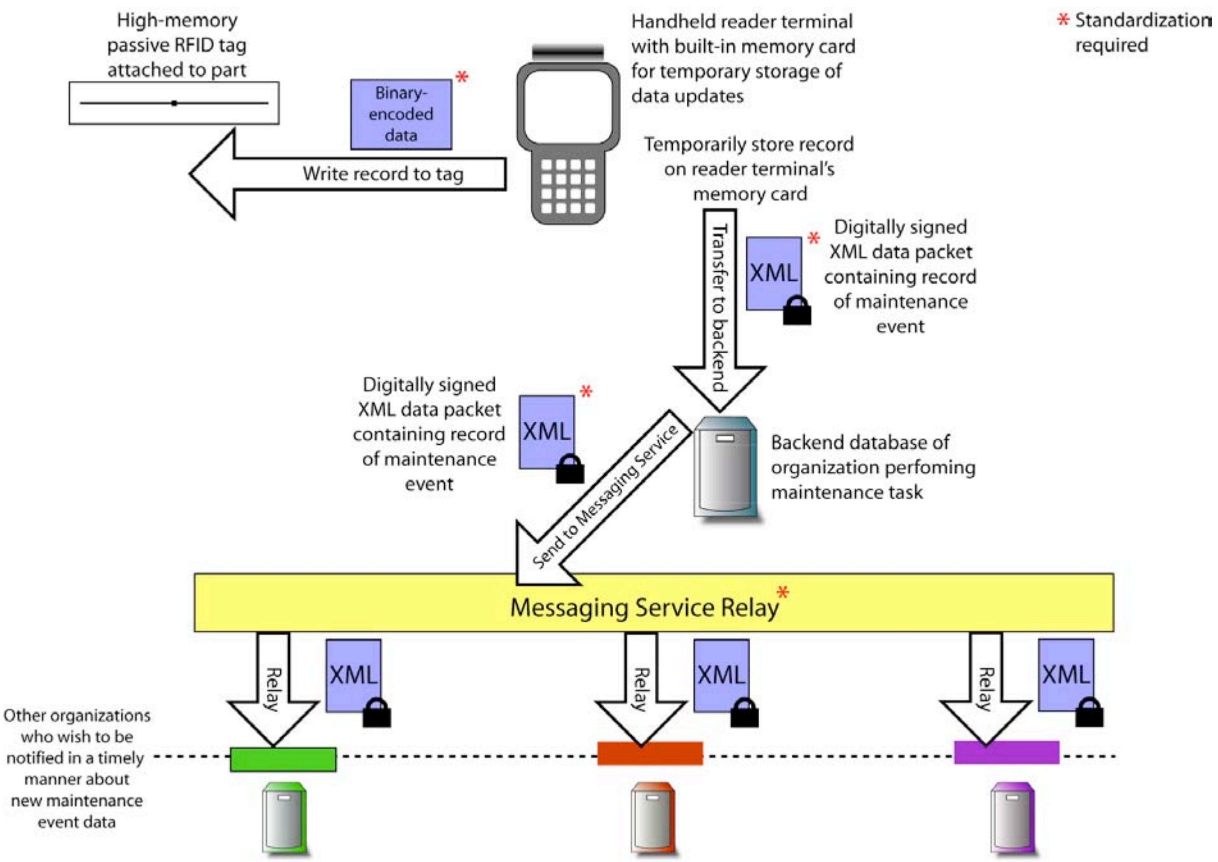

Fig. 2 – Communication of maintenance events across the network

This approach of recording data also onto the network, perhaps using a handheld device as an intermediate temporary cache also gives organizations in the aerospace sector the flexibility to choose whether to use high capacity tags for parts or to use cheaper lower capacity tags that store perhaps only a unique identifier and possibly some 'birth record' information. We can expect a range of tags to become available for the aerospace sector, with memory capacities ranging from as little as 96 bits to 512 bits, 2kbits, 8 kbits, 64kbits and perhaps higher capacities in future.

There is certainly not yet a clear consensus within the aerospace sector that all players will use or expect high capacity tags for each type of part – and the capacity of tag for a particular part will depend on a number of factors including the following:

- How much data about the part needs to be accessed in a location where access to the internet or other networks is not available?
- The volume of data to be stored with the object, considering the data rates available for reading that data (and whether this represents a significant time waiting to read or write data)
- The difficulty and cost of replacing a tag during a part's lifecycle if the memory becomes full (due to permanent locking of blocks of memory)
- The need to use digital signatures to assure the authenticity and authorship of the data. (Each digital signature may need between a few hundred bits and a few thousand bits of storage, depending on the bit-efficiency of the encryption algorithm)

Other factors affecting the complexity of the tag include the following:

- Whether the tag is required to be self-indexing or support a directory structure to enable direct access to specific data fields
- Whether a tag is required to support differentiated access control policies, to limit read/write permissions for specific data segments to specific users or groups of users.
- Granularity with which data blocks can be independently secured, e.g. by temporary or permanent locking.

Data Exchange among multiple organizations

The aim of using RFID with aircraft parts is not merely to collect more data – but rather, to collect it more efficiently, more accurately and using technology that allows for timely sharing of information with others who can improve their processes by using it. For example, if information about faults with parts is shared with the OEM vendors of the parts in a timely manner, it may be possible for them to analyse the fault reports to identify systematic performance issues on parts of a particular type or parts being used under particular conditions, which can in turn lead to opportunities to re-examine and improve their design and manufacturing processes to improve the reliability of parts. This is particularly important as a number of parts suppliers are moving away from a traditional sales model and towards leasing parts under guaranteed service contracts. It should be clear that there can be significant savings in time and cost by ensuring that the data captured in maintenance processes is promptly communicated over the network to other interested parties, regardless of how much or how little of the data is also written to the RFID tag.

There is also an aim towards replacing paper records with electronic records. A good example of this trend is the work by members of the ATA and others to develop an electronic version of the airworthiness certificate (FAA Form 8130 / EASA Form One)[7], which is not merely a digital bitmap image of a paper form – but rather a machine-readable dataset that contains all of the information contained in the paper form, but which can then be rendered in the format of the existing paper form – but can also be digitally signed, to ensure the authenticity and integrity of the data content.

Standard message formats are clearly an essential pre-requisite to information sharing among multiple organizations. The ATA RFID on Parts project team has already identified a number of data fields for the birth record data. More recently, the team has been working to identify which data needs to be collected for various types of maintenance events. Chapter 9.6 of ATA Spec 2000[8] already specifies data formats for traceability records – and it should be quite straightforward to prepare corresponding XML schema for these messages. The Aerospace ID research team has contributed some ideas and proposals to this work at the ATA.

If we are considering exchange of messages among multiple parties across the network, it is also essential that each party can gather the messages into the correct chronological order, to re-assemble the correct history of the part. Architecture paradigms for information sharing will be discussed further in the following section. However, it is conceivable that two maintenance events may happen to the same part on the same calendar day – and if we are receiving this information as messages over the network, we need to ensure that they are interpreted in the correct sequence.

It is essential that the message format is largely independent of the data carrier technology, especially as this technology will change over time – and vary in capability – so we need the message format to be largely independent of the data carrier technology (e.g. RFID, barcode, memory button etc.) used to store the data locally on the part. When we consider using digital signatures, it is important to be able to specify to which data the signature corresponds. It is therefore preferable to be able to refer to a particular maintenance event by it own unique ID rather than referring to a specific block of memory on the tag, since this may change, especially if a tag needs to be replaced when its memory is full and some of the most recent maintenance events from the old tag are written to the new tag.

One of the suggestions we have made to the ATA RFID on Parts project team is that it may be beneficial to give each maintenance event message a unique message ID. This can simply be constructed by combining the unique ID of the part with a precise timestamp – a resolution of 1 second is probably sufficiently granular.

In fact, computer systems typically store dates and timestamps internally as a long integer value, which is converted to and from a human-readable representation, as necessary. For example, the UNIX timestamp is based on the number of seconds elapsed since midnight on the 1st January 1970, UTC, not counting leap seconds. Most programming languages provide mechanisms[9] to convert between dates expressed as long integers and dates in standard human-readable formats.

Using 32 bits we can encode an Action Date in the format YYYYMMDD as an 8-character human-readable string with a granularity of 1 day. Alternatively, with 36 bits we can represent a timestamp with 1 second granularity until the year 4147 AD! We propose that this be considered as an element of a unique message ID for events, as illustrated in Figure 3.

This has the benefit of ensuring a unique ID for each maintenance event, which can then be used as a cross-reference to data written on the tag – or as a reference to a particular block of data that is digitally signed, while also enabling sorting of maintenance events into correct time order. This also addresses one aspect of data synchronization, particularly in the case of dealing correctly with late-arriving messages.

Which part?			When?	By whom?	What was done?	Details specific to type of activity	Additional information		
CAGE Code	Unique Serial No	Current Part No	Action Date	Action Company	Action Code	Removed From:	Condition Code	Your Part No	Other Data etc.
61G49	1234567	P7DTR26	20020420	81979	RMV	SPL 73T11/UCN 872615	UNS	123-4567	

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Traceability xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="Traceability.xsd" Version="1.0">
  <HDR_Segment>
      <Timestamp>1019304247</Timestamp>
      <MFR>61G49</MFR>
      <SER>1234567</SER>
      <PNR>P7DTR26</PNR>
      <ACD>2002-04-20</ACD>
      <ACO>81979</ACO>
      <ACT>RMV</ACT>
  </HDR_Segment>
  <ACT_Details>
      <RMV_Segment>
          <Removed_From>
              <SPL>73T11</SPL>
              <UCN>872615</UCN>
          </Removed_From>
      <RMV_Segment>
  </ACT_Details>
  <CND>UNS</CND>
  <Your_Part_Num>123-4567</Your_Part_Num>
</Traceability>
```

These three elements can form a unique ID for this maintenance event

Details specific to a removal event

Fig.3. Example of how a maintenance record might be represented in XML

Architectures for information sharing

There are two main modes of operation for distributed information sharing across multiple organizations – messaging and information retrieval.

Historically, much of the electronic data interchange (EDI) between organizations has used the messaging paradigm, in which one organization sends a message in a standard format to many recipients, possibly assisted by an intermediary message relay or broker. The sender of the message can 'send and forget' and is not necessarily required to retain the information for long-term retrieval; it is the responsibility of the recipients to store the messages they receive if they require this.

After checking the format, authorship and authenticity of the messages received, they can then use data-binding technologies to extract data and map this into their existing databases and information systems.

Examples of such messaging within the air transport industry are the IATA Type A and Type B message standards and the emerging Type X messaging from ARINC and SITA.

In the information retrieval paradigm, each organization collects information about objects while they are in their custody and stores this within their own data repository. In practice, this may be a stand-alone replica system that is regularly synchronized with selected data from the main operational databases used within a company. There is no obligation to send a message to multiple recipients for each piece of data collected – but information sharing can be facilitated through the use of standard query interfaces to the data repositories, together with serial-level registries or lookup services which enable an authorized party to find multiple providers of information about a particular part or object.

An example of a distributed information retrieval architecture is the EPC Network[10], in which EPC Information Services (EPCIS)[11] are distributed data repositories with standard query interfaces – and Discovery Services provide the serial-level registries or lookup services across the lifecycle of a part. SITA have partnered with VI Agents and Afilias to offer such a community-based information retrieval service[12] for the air transport sector. Furthermore, there is already a freely available open standard from EPCglobal on EPC Information Services[11], which could be readily applied to the air transport sector. The EPCIS v1.0 standard already includes extension mechanisms to allow for industry sectors to use their own data dictionaries and structured data formats within EPCIS events. This means that an XML message format for a maintenance event could be embedded within an EPCIS event and retrieved using the standard EPCIS query interface if the identity of the part or other criteria (e.g. time range, location) about the event are known. The EPCIS v1.0 standard does not currently provide a mechanism to do a granular XML query within the extended data, for example using an XPath[13] expression. This means that it may not be trivial to directly query an EPCIS repository for all maintenance events where a part was overheating or leaking, unless this information were additionally recorded in one of the standard EPCIS event data fields, such as the 'disposition' field. However, some implementations of EPCIS or future versions of the EPCIS standard may provide this query functionality within the data extensions. Table 1 compares the features of these two approaches to information sharing.

Messaging	**Information Retrieval** (e.g. via EPCIS and Discovery Services)
Send and Forget (less expectation to retain long-term)	Store and be prepared to provide
Message sent to (multiple) individual specified recipients or to a 'topic' (e.g. mailing list, where subscribers may not be known to whoever posts a message)	Message stored locally or via hosted service with local management.
It may be difficult to specify different access controls for messages sent to a distribution list for a topic	Decisions regarding access control can be made on a per-query basis - but preferably in an automated way
An archive account may subscribe to collect all messages posted by a particular organization - or posted for a particular topic	A data repository (e.g. EPCIS) provides archival of events
Newcomers can access historical messages via the archive, if an archive is available	Newcomers can query the data repository for historical events. No major disadvantage to 'joining late'.
Subscription interface for list manages distribution to multiple recipients per topic	Publish & Subscribe interface manages distribution of new events to multiple subscribers
Messaging alerts interested parties of new events from known information providers (and from new information providers) and carries payload	EPCIS standing queries notify subscribers about new events from known information providers. Standing queries on Discovery Services notify subscribers about new providers of information - and could also carry a data payload attachment
Discovery Services could also provide links to multiple message archives, when queried by topic, e.g. National Stock Number (NSN) or Part Number.	Discovery Services enable gathering of information from multiple providers (e.g. multiple EPCIS instances)
Well suited for short-lived operational messages	Intended for both pub/sub notification of current events + long-term retention & access

Table 1 – comparison of Messaging with Information Retrieval

Outlook – towards a hybrid approach to information sharing

Some companies are already willing to migrate from the traditional messaging approach to an information retrieval approach – and in some application scenarios, such as interline e-ticketing, this change from messaging to database lookup has already happened. Other companies are more cautious about adopting new technologies such as web services and about providing standard query interfaces to their information systems. In the near term, it will be important to consider providing hybrid approaches, so that companies can initially use familiar messaging services, then migrate at their own pace to an information retrieval mode of operation.

This hybrid solution can be achieved as follows:

Additional 'archival' users can be attached to existing messaging services. These archival users store a copy of each message in a repository for the organization that sent the message. By transforming each message into an 'event' and providing a standard interface for querying each archive, each organization can also provide an information retrieval service (e.g. EPCIS) to trading partners who prefer to retrieve information rather than receive messages, with their archival repository accumulating data with each message that they send.

Information retrieval services such as EPCIS can support not only one-off queries but also long-running standing queries, in which new events added to the data repository are communicated to one or more recipients who have already subscribed via a standing query to indicate their interest in receiving this information in future. Subscribing to a standing query is very much like subscribing to a mailing list or message queue on a specific topic, except that each recipient can specify their own topic criteria in a very granular way. The messages sent in response to standing queries can of course use existing messaging services, such as Type X, which have been developed for guaranteed end-to-end delivery of XML messages.

Conclusion

This paper summarizes the activity within the Lifecycle ID & Data Management research theme, the involvement in the ATA RFID on Parts project team and also some recent discussions with SITA regarding information sharing architectures. At all times, we have attempted to encourage all players to think beyond the data that might be written to an RFID tag and consider how they might share information with other organizations in a timely and efficient manner, in order to achieve shared benefits and improved reliability and passenger safety.

References

1. *EPC - Electronic Product Code*. http://www.epcglobalinc.org/standards/tds
2. Harrison, M., *EPC Identifiers for Aerospace*, in *Aerospace ID Technologies Programme: White Papers*. 2006. http://www.aero-id.org
3. *EPC Tag Data Standards*. http://www.epcglobalinc.org/standards/tds
4. *EPC Tag Data Translation standard*. http://www.epcglobalinc.org/standards/tdt
5. *Accada.org - open source software for the EPC Network*. http://www.accada.org
6. *EPCglobal UHF Class 1 Generation 2 air protocol*, EPCglobal Inc. http://www.epcglobalinc.org/standards/uhfc1g2
7. *FAA Form 8130-3, Airworthiness Approval Tag*, Federal Aviation Administration. http://forms.faa.gov/forms/faa8130-3.pdf
8. ATA, *Air Transport Association Spec2000 standard*. http://www.spec2000.com
9. *For example, the java class java.util.Date provides such conversion methods*. http://java.sun.com/j2se/1.4.2/docs/api/java/util/Date.html
10. *EPCglobal Architecture Framework Version 1.0*. http://www.epcglobalinc.org/standards/architecture
11. *EPC Information Services (EPCIS) v1.0 standard*, EPCglobal Inc. http://www.epcglobalinc.org/standards/epcis
12. *SITA AutoID pilot community service*. http://sita.autoid.aero
13. *XPath*, W3C - World Wide Web Consortium. http://www.w3.org/TR/xpath