Accurate tracing at the parts level can play a key role in easing and managing maintenance operations within the railway industry, the same way it streamlines freight operations on more traditional deployments. Transf-ID is an RFID-based tracking system for use in the rail freight industry that will identify cargo and rail vehicles but also mechanical parts that are frequently serviced. The system includes middleware as well as hardware components, combined in an end-to-end solution that translates tag scans into meaningful business events. All tagged items are scanned on the move at checkpoints, leading to valuable information regarding their traveled routes.

Raywood freight operations deal with numerous moving assets, including freight, rail vehicles, and other transportation material. Much of this “rolling stock” comprises distinct mechanical parts that workers must regularly exchange during routine business operations. Safety regulations and business processes require carriers to track, trace, and maintain all these assets and parts. Maintenance itself is time-consuming and requires checking many individual — often inaccessible — mechanical parts, but what if vehicle components themselves could assist in the checking procedures?

RFID technologies have become the cornerstone for enabling this functionality (see the “Related Work in RFID Deployment” sidebar). Tags identify the parts they’re mounted on. Antennas broadcast radio frequency (RF) signals and receive tag responses from areas they point toward, then forward the raw RF signals to the readers controlling them. Readers, in turn, decode the signals from the different antennas into byte sequences and, eventually, into meaningful software messages. Finally, middleware aggregates data from scans that it then feeds into a processing module, which estimates service parameters.

Here, we present the Transf-ID prototype, which tracks parts such as rail vehicles, chassis, axles, and swap bodies (http://en.wikipedia.org/wiki/Swap_body). Axles are the most relevant me-
chanical parts because they require periodical maintenance, and workers must exchange them at many international borders because prevailing gauge standards differ. We’ve also performed initial tests with other important parts, such as bogies (http://en.wikipedia.org/wiki/Bogie) and couplings. We studied adequate tag placement of these elements over the broad range of wagon types we had available. Our middleware aggregates data from scans, identifying all the different train components and their relationships with one another. The system tracks and traces assets and infers the traveled routes, accurately assessing age, mileage, or wear — particularly relevant for parts that are frequently exchanged among different freight cars. This improves the efficiency of maintenance tasks performed on moving parts. The system automatically keeps an inventory and plans maintenance schedules based on accurate mileage for each part, rather than just age.

**System Requirements**

Transf-ID must comply with several functional, physical, and operational requirements.

**Functional Requirements**

Our fundamental aim with Transf-ID was to build a system that could track all mobile assets found in the rail freight carrier domain (see Figure 1). The infrastructure needed to work with different assets, such as freight cars, containers, boxcars, chassis and bogies, removable axles, couplings, swap bodies, cargo, locomotives, and maintenance road-rail vehicles. Because we can tag all these assets on their sides, we can thus use the same set of antennas to read all tags.

The system identifies assets when they pass in front of readers, entering or exiting control points. They don’t pass in isolation but are in some way interrelated (an engine moves a car, which in turn runs on axles and carries a container). Our middleware receives this identification data and aggregates it to infer the train’s composition. The middleware can also locally preprocess identification data at checkpoints and send it through an IP network to the corporate data processing center, where the railway operator can apply business logic rules.

All the business events we’re after respond to a common pattern: assets on a convoy vary between two successive scans — that is, they enter and exit a business-significant location. Changes in a convoy’s line-up indicate either railroad-specific events (axle swapping or replacements at overhauls, for example) or freight-specific events (loading and unloading operations or shipment breakdowns and consolidations). Correspondingly, we can establish RFID checkpoints at loading/unloading points, transfer or intermodal stations, border stations, maintenance sites, and simple midroute checkpoints.

Identification and classification, together with appropriate reader deployment, let operators control assets’ locations at business-critical checkpoints in near real time. This is invaluable information for them and for customers who hire operators to send freight they wish to check on. Mileage-based maintenance operations become possible and can be triggered automatically. Because more reliable information is available, the operator can employ more lenient safety margins for maintenance scheduling while still observing regulations, thus maintaining or improving overall safety. We must, however, differentiate between true business events and normal operational maneuvers. Trains must frequently enter, retreat, and re-enter the same station or check-
RFID meets the Internet

point to allow workers to attach extra wagons to the convoy. All these maneuvers result in elements passing by the checkpoints multiple times and producing a confusing pattern of RFID reads. Instead of installing direction transducers below the tracks, our system must rely only on the data the RFID readers provide.

Physical Requirements
Using RFID tags for labeling rail assets involves some particularly demanding conditions that don’t arise in other, widespread scenarios. The European Telematic Applications for Freight, Technical Specifications for Interoperability (TAF-TSI) already include many requirements¹ (section 2 of these regulations explains the quantitative details). These regulations plan for

• harsh outdoor environments, with exposure to dust, moisture, chemicals, wear, vibrations, and extreme temperatures;
• unobstructed clearance areas surrounding rail cars, requiring us to place readers within state-of-the-art reading ranges;
• strong impacts during handling by cranes and the need to prevent tags from becoming protruding parts, meaning we can place tags only in protected hollows;
• heat from internal friction, further restricting possible locations (leaving as little as 12.5 square centimeters for tag placement); and
• tag reading that occurs at 30 km per hour (tags pass through in small groups, however, so tag density isn’t an issue).

Physical requirements also relate to

• selecting tags’ optimum placement to ensure good reading ranges,
• protecting against vandalism, and
• dealing with cumbersome reading on metal mounting and metal environments.

All these requirements imposed tight restrictions on the placement, size, toughness, and endurance of the tags we could use.

Operational and Business Requirements
We wanted to keep human intervention during Transf-ID operation to a minimum. Consequently, we needed to pay attention to its physical durability and logical stability. Tags’ service lifespan should mirror that of the (often long-lasting) identified parts, thus avoiding costly manual replacement procedures. Current battery lifetimes don’t satisfy this requirement, so we needed to discard active and semiactive tag types in favor of passive tags, which would be maintenance-free. Together with the portrayed physical and functional requirements, this tightly limits the tag types we can choose from.

To maintain robustness, the system also needed to deal with several abnormal conditions autonomously. In our system, redundant information provides failure tolerance, so malfunctioning or lost tags won’t prevent the system from identifying all components and signaling faults. We also planned for reads missed owing to interference or poor reading range. Lastly, the system needed to adapt to anomalous circumstances that make convoy traffic, routes, and even processes vary significantly from the norm.

The software also needed to take on massive, automatable operations. Without the burden of these tasks, onsite personnel can increase their productivity because they’ll rarely need to intervene. We gave such personnel PDA-based handheld readers equipped with an issue-resolution wizard application. We emphasized the application’s usability so that untrained personnel could operate it in a disturbing and hazardous environment where trains pass by. In case the problem escalates and requires staff intervention, the system needed to let them operate from remote premises.

Transf-ID also had to fully integrate with existing corporate software. This meant developing some components on existing software and interfacing through available ports, but it was also relevant to vendor selection and required us to follow corporate security policies.

Finally, we tried to honor cost restriction with our design, given that we want to prove profitability for such systems when they’re widely deployed.

Distributed Middleware Architecture
The system architecture we chose employs commercial middleware, which provides a solid foundation for our system at a moderate cost as well as a mature and modular design. It supports the readers we considered, as well as Electronic Product Code (EPC) standards — such as Application Level Events (ALE)² and Electronic Product Code Information Services (EPCIS)³ — and lets us add custom-made modules. The use
of commercial middleware will also make future growth easy for our prototype by, for instance, simplifying the remote reader operation. Additionally, most RFID middleware vendors provide solutions that interoperate with existing software, passing messages on to user applications via Java Message Service (JMS), Web services, or other technologies. These solutions can also guarantee message delivery should network failures occur.

A service-oriented architecture (SOA) lets us easily replace our chosen middleware by maintaining the system’s core functionality. In fact, we developed two distinct versions of our software that can run on middleware from two vendors, with minor cosmetic changes. Figure 2 shows the elements in one of our two possible configurations.

Our architecture supports multiple locations from which readers can provide information using JMS messaging. To remain vendor agnostic, we used only some core functionality from the Premises Server module (an RFID-specific middleware from IBM). Primarily, we use it to route all JMS messages from readers to the corporate enterprise resource planner (ERP) and database, and from corporate systems back to the readers and other onsite hardware (light stack).

The core logic lies on custom-made modules, which communicate through Web services. Inferring a convoy’s lineup isn’t a straightforward process because we can’t make any assumptions regarding its arrangement. The core service module gathers as much information as it can from scanned tags and passes the inferred lineup on to the corporate systems. Inconclusive scan data or incomplete wagons trigger error delivery. The ERP can also route errors down to the station, alerting operators when, for instance, a container is present that should have been boarded on a different train.

Corporate management systems turn events into profitable information. They make logical decisions involving, for example, orders issued to maintenance teams to inspect service parts that have experienced excessive mileage. Or,
they can react to errors detected during the reading process and request manual intervention. The corporate information system with which we integrated Transf-ID carries most of its logic as Procedural Language/Structured Query Language (PL/SQL) procedures and action triggers. These procedures perform calls issued on feedback Web services to request actions.

A local operator can use the mentioned PDA application to manually recheck conflicting elements with the PDA’s RFID reader. We also developed a tag recorder application — not shown in Figure 2 — to ease the process of assigning new EPC numbers to each tag and mounting them on their respective items.

**Hardware and Embedded Components**

In addition to the middleware, our solution comprises various hardware elements, some of which host embedded software components.

**Tags**

Exploring the market for suitable tags, we found that none fit our entire list of requirements (size, range, form factor, and suitability for metal mounting and harsh environments). Pharmaceutical industry inlays offered a good starting point in terms of size and range, so we tried to provide more appropriate encapsulations for them. We finally chose a combination of inlay and encapsulation that a local company later batch-produced for us.

As mentioned, in the railway industry, tags must be metal mounted — a demanding situation for an RFID system. When an RFID antenna emits a signal, eddy currents appear along metal surfaces that create their own electromagnetic fields, which interfere with the signal our RFID tags try to return. Another significant problem is detuning due to parasitic capacitance — an undesirable and inevitable capacitance that appears between closely spaced conductors, such as the inlays’ circuit and the metal surface where a tag is placed. The simplest solution is to allow some separation between the inlay and metallic surfaces. Our tag carries a thick encapsulation and a special insulation layer that also minimizes magnetic interference.

**Antennas**

Every checkpoint has an antenna on each side of the track to read tags on either side of a train. However, the distance between both antennas presented a challenge because signals are weakened too much over longer wires, especially for small tags. So, state-of-the-art performance was a must (our required reading range was just shy of two meters), and our choice of readers and antennas was critical.

We selected an antenna model that uses separate loops for transmission and reception, which results in better sensitivity. It also supports configuration for linear polarization, which required that we place all tags horizontally for correct alignment with the antennas and that we choose tags with inlays that were manufactured to be optimized for this polarization.

**Readers**

The reader we chose supports deploying embedded middleware, which we can use to filter reads within the reader (thus reducing network traffic) and to interface with external signaling equipment. Achieving vendor independence made reader compatibility with most commercial RFID middleware vendors a major concern for us. Likewise, all hardware and software our system employed had to conform to EPCglobal (866 MHz) standards, which are the prevailing standards in the RFID industry.

**Implementation Details**

To test a prototype of our system with complex traffic patterns, we deployed it in 2008 at a station on the border between Spain and France. We wanted to prove technical viability for this kind of system and, just as important, to assess whether wide-scale deployment would prove cost effective in day-to-day operation. We can ascertain the latter only over a longer period of operation. We’re currently evaluating the results to determine the extent and schedule of further deployments.

**Adapting to EPC Coding Schemes**

We designed an internal transcoding scheme for EPC codes that reuses most original information in existing serial-number formats for identifying rail assets and freight (for instance, a vehicle identification number [VIN] in the case of automobiles). Manual operators find this more intuitive than completely new numbering schemes.

**Layered Event Processing**

The middleware must process events that come from the raw flows of tag readings in order to
turn them into significant, business-profitable information. The system achieves this by using a layered processing scheme that yields more abstract events as processing occurs.

**First layer.** We place an antenna at each side of the track, so a data flow with raw reads comes from each rail. We tag each asset at least twice for redundancy: shorter assets (such as axles) have a tag on each side (both treated equally), whereas longer assets (such as train cars) have tags at both ends on opposite corners, always in the same diagonal. The first layer performs tag filtering, smoothing out reads and consolidating a unique, ordered flow that filters out duplicate and vanishing reads.

**Second layer.** A checkpoint isn’t an isolated portal but a “black box” with reading gates at every entry (one or more). Checkpoints take incoming elements and produce outgoing ones—we didn’t take for granted any information about operations inside black boxes. This layer first maps EPC codes to asset classes and codes, then infers the black boxes’ content at every moment using a stateful automaton whose work is governed by a set of metarules that define black-box behavior. On the basis of those metarules, each time a reader detects an asset, the automaton tries to match the current black box’s state and the recent read history to one of a few dozen patterns that mirror the different situations that could be occurring inside the box. We derive specific patterns from a simplified set of metarules:

- Black boxes can contain an initial inventory of stored assets. The system assumes that read assets already present in the initial inventory are exiting.
- Assets previously recognized as entering the black box and then reappearing are exiting.
- Assets that must be mounted on other assets (such as an axle on a wagon) have the same direction as their supporting asset.
- Assets that can carry other assets always have tags on both ends (under normal operation). Owing to central symmetry, at each gate, the left rail (as seen from outside the station) always receives the first tag of an asset entering the black box and the last tag of an asset leaving it, and vice versa for the right rail. This lets the system infer direction.
- The black box preserves asset lineup between reads (a convoy might reverse its path, but neither it nor any of its elements flip inside a black box).
- If two nonconsecutive assets exit through a gate, all those previously detected between them are no longer in the convoy. Either a swappable part or cargo was replaced, or there was an error that the system needed to signal.
- Black boxes are reset after a convoy leaves (everything that enters a station should eventually exit or be placed in storage).

Let’s look at a simple case. Consider a car with two tags ($C_{1L}$ and $C_{1R}$) entering a station:

1. An antenna scans $C_{1L}$, and the beginning of the car goes into the black box.
2. $C_{1L}$ disappears, meaning either that the rest of the car keeps entering the black box or that it retreats as part of a maneuver. This is a metastable state.
3a. If $C_{1L}$ is read again, regardless of what happened before, we know that the beginning of the car is again passing the gate.
3b. Otherwise, if $C_{1R}$ is read instead, we know the whole car has entered the black box. In either step 3a or 3b, the state has collapsed to a decidable black-box configuration.

The second layer provides an ordered list of assets and their situation with respect to the gates, allowing higher-level layers to group them.

**Third layer.** The next layer follows a similar design as the second one (an automaton that tries to match event history with patterns) but with a different mission. Each asset (cargo or parts) is grouped with the wagon that carries it, thus creating packages for sets of assets. Patterns here are more complicated because they also need to deal with faulty situations (at least signaling, if not recovering from them) such as missed reads, missing tags, or incompatible asset configurations. Either way, they always abide by the metarules we explained earlier.

Following the example, when the reader detects $C_{1L}$ and $C_{1R}$, this layer creates a car package for $C_{1}$ and the rest of the read assets.

**Fourth layer.** This layer compares convoy line-ups at each passage through a gate, thus dis-
RFID Meets the Internet

Distinguishing business events from maneuvers. For instance, if C1 exits carrying a different load or axles, it triggers an event that the system reports to the operator’s corporate management systems.

Finally, corporate systems match convoy line-ups with shipment schedules to track parts. They know at any moment which wagon is carrying which parts and, by aggregating route data, can calculate for how long and how far. This triggers maintenance operations when suitable.

Testing and Deployment

Initial tests for tag selection occurred in a controlled, open-air field environment. We adapted a setup suggested elsewhere to work with our reader in dense mode for EPC Gen2 at 866.9 MHz. Each reader’s RF output emits 33 dBm (decibels above 1 milliwatt) to a 6-dBi-gain (decibels with respect to an isotropic radiator), linearly polarized, double-loop, patch array antenna with no attenuator in between. Consequently, we placed the tested tags at four meters from the antenna (roughly 6.9 times the original distance on a nonreflective surface).

We deployed our prototype in a real station, pictured in Figure 3. This particular test station has two through-tracks with complex traffic patterns. The station also performs many maintenance tasks related to wagon axles and thus presented a perfect test scenario. We placed the antennas at their definitive locations: at the edge of the restricted clearance, roughly two meters from the passing tags. Test runs with several fully tagged convoys showed virtually 100 percent correctly read tags when trains passed by. The middleware also operated as expected during these simple test runs, correctly identifying axle replacement events and train lineups. We had to immediately replace a few defective tags because they were unreadable even in close proximity, but range wasn’t an issue with the rest. We also performed simulations for more random train maneuvers that we might expect at this station.

We began our first production deployment by tagging 500 rail vehicles and their parts (with several thousand tags), which was completed by mid 2008. To better assess the likely cost savings, the system will soon operate on real traffic. We’ll evaluate scan-failure rates over the next year and tag-failure rates over the next few years.

Our system is a fully functional prototype of both hardware and software that presents several breakthroughs. The prototype demonstrates success — for the first time in the railway industry — at using such small, passive tags to tag metallic parts and read them on the move. From our prototype, we expect to continue working toward a full-scale production deployment, introducing the Transf-ID system in other places (throughout the Spanish rail network and other neighboring networks) and broadening the range of tagged assets. Also, as EPCglobal architecture and other logistic information networks continue evolving, the system could implement controlled interfaces for third parties (other carriers, operators, customers, or rail administrators) that might have legitimate interest in the system’s information.

Our system delivers savings through management and maintenance optimization. It enables tracing all rail assets and, as a result, it globally improves rail freight safety because mechanical parts can go through more realistic maintenance schedules that adapt to actual use. This experience demonstrates that low-cost,
RFID technologies in business areas such as retail are gaining widespread deployment. However, this isn’t the case for many other fields, which has prompted the European Commission to regard the field as one of the most promising future research areas.\(^1\)

Rail applications have seen a quick evolution, spreading from the US to other countries since the 1980s.\(^2\) Some RFID rail applications have long been standardized in both the US (Automatic Equipment Identification\(^3\)) and Europe (Telematic Applications for Freight Technical Specifications for Interoperability\(^4\)). Current applications include hardware and controller designs tailored for rail scenarios,\(^5–7\) as well as more comprehensive systems. Among those, most US carriers (including all Class I railroads) as well as others in Europe (FlyToGet, Schweizerische Bundesbahnene [SBB], and Nederlandse Spoorwegen [NS]) and Asia (JR Freight) have adopted some sort of solution.

However, the Trans-ID system we present in the main text has some clear differences because it extends this kind of system to encompass freight cars, cargo, and mechanical parts in rail vehicles. A single, unified system will track all three types of components. Previous applications were practically limited to locomotives, wagons, and cargo containers, but extending the system to smaller railway items or mechanical parts is a breakthrough, requiring a remarkable reduction in tag size. Typical RFID tags for metal environments are around 15 cm long, whereas our prototype employs tags sized at 5 × 2 cm.

Finally, tagging railway mechanical parts presents specific challenges that others have addressed primarily by using active tags (www.logica.com/file/4594). We have successfully introduced cheaper, maintenance-free, passive tags in our system, much smaller than those typical of today’s railway industry and suitable for many of a rail vehicle’s mechanical parts. We recently encountered some work in progress that attempted to tag axles with passive tags, from Portuguese operator Comboios de Portugal (CP).\(^8\) Its system is designed for stationary reads at maintenance premises, whereas ours can track tags on the move. Another recent proposal\(^9\) has a sophisticated tag design but doesn’t provide a comprehensive system and uses far bulkier tags, inappropriate for smaller items.

Acknowledgments
Transf-ID was partially funded through Profit (a Spanish public R&D program) from the Ministerio de Industria, Turismo y Comercio, under project CIT-370400-2005-7.

References
RFID Meets the Internet

Jorge González Fernández is the IT business development director at DHL Exel Supply Chain, Spain, where he’s responsible for IT function alignment, improvement, and business development across all business divisions. His research interests include the Internet of Things, networked enterprise, and IT in supply chains. González has an MSc in telecommunications engineering and a master’s in IT management from the Instituto de Empresa, and a PDD in management from IESE. He has been part of several European R&D projects concerning IT logistics. Contact him at jorge.gonzalez.fernandez@dhl.com.

Juan C. Yelmo García is an associate professor in the Department of Telematics Engineering at Universidad Politécnica de Madrid (DIT-UPM), and the director for doctorate and postgraduate studies in UPM’s Higher Technical School of Telecommunications Engineering. His research interests include services engineering, the Internet of Things, digital identity, and distributed applications and middleware. Yelmo has a PhD in telecommunications engineering from UPM. Contact him at jcyelmo@dit.upm.es.

Yod-Samuel Martín García is a researcher in the Department of Telematics Engineering at Universidad Politécnica de Madrid (DIT-UPM). His main interests are distributed service architectures and human–computer interaction. Martín has an MSc in telecommunications engineering and an MPhil in telematics engineering, both from UPM. Contact him at samuelm@dit.upm.es.

Jorge de Gracia Santos is a researcher and PhD student in the Department of Telematics Engineering at Universidad Politécnica de Madrid (DIT-UPM). His research interests include the Internet of Things and middleware applications. De Gracia has an MSc in telecommunications engineering from UPM. Contact him at jorgegs@dit.upm.es.


From the analytical engine to the supercomputer, from Pascal to von Neumann, from punched cards to CD-ROMs—the IEEE Annals of the History of Computing covers the breadth of computer history. The quarterly publication is an active center for the collection and dissemination of information on historical projects and organizations, oral history activities, and international conferences.

www.computer.org/annals