INTERNATIONAL STANDARD

ISO/IEC 24730-1

Second edition
2014-02-15

Information technology — Real-time locating systems (RTLS) —

Part 1: Application programming interface (API)

Technologies de l'information — Systèmes de localisation en temps réel (RTLS) —
Partie 1: Interface de programmation d'application (API)

Reference number
ISO/IEC 24730-1:2014(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 24730-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture techniques*.

This second edition cancels and replaces the first edition (ISO/IEC 24730-1:2006), which has been technically revised.

ISO/IEC 24730 consists of the following parts, under the general title *Information technology — Real time locating systems (RTLS)*:

— *Part 1: Application programming interface (API)*

— *Part 2: Direct Sequence Spread Spectrum (DSSS) 2,4 GHz air interface protocol*

— *Part 21: Direct Sequence Spread Spectrum (DSSS) 2,4 GHz air interface protocol: Transmitters operating with a single spread code and employing a DBPSK data encoding and BPSK spreading scheme*

— *Part 22: Direct Sequence Spread Spectrum (DSSS) 2,4 GHz air interface protocol: Transmitters operating with multiple spread codes and employing a QPSK data encoding and Walsh offset QPSK (WOQPSK) spreading scheme*

— *Part 5: Chirp spread spectrum (CSS) at 2,4 GHz air interface*

— *Part 61: Low rate pulse repetition frequency Ultra Wide Band (UWB) air interface*

— *Part 62: High rate pulse repetition frequency Ultra Wide Band (UWB) air interface*

Introduction

ISO/IEC 24730 defines several air interface protocols and a single Application Programming Interface (API) for Real Time Locating Systems (RTLS) for use in asset management and is intended to allow for compatibility and to encourage interoperability of products for the growing RTLS market.

This part of ISO/IEC 24730, the RTLS Application Programming Interface, establishes a technical standard for Real Time Locating Systems. To be fully compliant with ISO/IEC 24730, Real Time Locating Systems (RTLS) shall comply with this part of ISO/IEC 24730 and at least one air interface protocol defined in ISO/IEC 24730.

Real Time Locating Systems are wireless systems with the ability to locate the position of an item anywhere in a defined space (local/campus, wide area/regional, global) at a point in time that is, or is close to, real time. Position is derived by measurements of the physical properties of the radio link.

Conceptually there are four classifications of RTLS:

— Locating an asset via satellite - requires line-of-sight - accuracy to 10 meters

— Locating an asset in a controlled area, e.g., warehouse, campus, airport - area of interest is instrumented - accuracy to 3 meters

— Locating an asset in a more confined area - area of interest is instrumented - accuracy to tens of centimetres

— Locating an asset over a terrestrial area using terrestrial mounted receivers over a wide area, cell phone towers for example – accuracy 200 meters

With a further two methods of locating an object which are really RFID rather than RTLS:

— Locating an asset by virtue of the fact that the asset has passed point A at a certain time and has not passed point B

— Locating an asset by virtue of providing a homing signal whereby a person with a handheld can find an asset

Method of location is through identification and location, generally through multi-lateration, of various types

— Time of Flight Ranging Systems

— Amplitude Triangulation

— Time Difference of Arrival (TDOA)

— Cellular Triangulation

— Satellite Multi-lateration

— Angle of Arrival

The location information of an asset may further be enhanced with information on its spatial orientation.

This part of ISO/IEC 24730 defines an application programming interface (API) needed for utilizing an RTLS system.

An API is a boundary across which application software uses facilities of programming languages to invoke services. These facilities may include procedures or operations, shared data objects and resolution of identifiers. A wide range of services may be required at an API to support applications. Different methods may be appropriate for documenting API specifications for different types of services.

The information flow across the API boundary is defined by the syntax and semantics of a particular programming language, such that the user of that language may access the services provided by the application platform on the other side of the boundary. This implies the specification of a mapping of the functions being made available by the application platform into the syntax and semantics of the programming language. An API specification documents a service and/or service access method that is available at an interface between the application and an application platform.

This API describes the RTLS service and its access methods, to enable client applications to interface with the RTLS system. This RTLS service is the minimum service that shall be provided by a RTLS system to be API compatible with this standard.

This part of ISO/IEC 24730 uses a “full stop” as the decimal point separator since an API file is being created with an output in a .csv file format which uses the comma to separate values.

Information technology — Real-time locating systems (RTLS) —

Part 1: Application programming interface (API)

1 Scope

This part of ISO/IEC 24730 enables software applications to utilize an RTLS infrastructure to locate assets with RTLS transmitters attached to them. It defines a boundary across which application software uses facilities of programming languages to collect information contained in RTLS tag blinks received by the RTLS infrastructure.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15963, *Information technology — Radio frequency identification for item management — Unique identification for RF tags*

ISO/IEC 19762 (all parts), *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary*

IEEE Guidelines for use of a 48-bit Extended Unique Identifier (EUI-48™)

IEEE Guidelines for 64-bit Global Identifier (EUI-64™) Registration Authority

Extensible Markup Language (XML) 1.0, (Third Edition), W3C Recommendation, World Wide Web Consortium (W3C), 4 February 2004[1)]

SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), W3C Recommendation, World Wide Web Consortium (W3C), 27 April 2007[2)]

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19762 (all parts), and the following apply

3.1
field
element of a data record in which information is stored, which may contain one or more properties of a tag blink

3.2
XML tag
marker that qualifies content in a XML document

1) http://www.w3.org/TR/REC-xml/

2) http://www.w3.org/TR/2007/REC-soap12-part1-20070427/

3.3
persistent connection
network connection between a server and a client that is kept open for several application level message exchanges, or request call, even after sending application level error responses

3.4
tag status
mandatory fields within a Locate message not including the <source> and <format> fields

4 Symbols and abbreviated terms

For the purposes of this document, the symbols and abbreviated terms given in ISO/IEC 19762 (all parts) and the following apply.

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CR	ASCII Carriage Return
EUI	Extended Unique Identifier
JMS	Java Messaging Service
HTTP	HyperText Transfer Protocol
HTTPS	HTTP Secure Protocol
LF	ASCII Line Feed
OUI	Organizationally Unique Identifier
REST	Representational State Transfer
RTLS	Real Time Locating System
S-HTTP	Secure HTTP Protocol
SLMF	Simple Location Message Format
SLMP	Simple Location Message Protocol
SOAP	Simple Object Access Protocol
SSL	Secure Sockets Layer
TDOA	Time Difference Of Arrival
TCP/IP	Transmission Control Protocol / Internet Protocol
XML	eXtensible Markup Language
XSD	XML Schema Definition

5 The service

5.1 Purpose

The purpose of the RTLS software API is to provide a simple minimal interface that facilitates adoption and can be implemented easily on both the RTLS system and the application. The purpose is also to

allow fast transfer of messages to any client that connects to the RTLS system. In addition, the message flow should be human-readable and easy to interpret.

The API is to be supported by a device of minimal functionality, which is a 'collection and forwarding device', with no persistency or database required.

The device can provide rate filtering and location smoothing capabilities, but these functions are not required by the proposed API because the API can operate before or after this type of pre-processing.

5.2 Specification summary

The RTLS software service shall be a 'Text over Socket' connection, such that a TCP/IP client can connect and receive a real-time stream of RTLS messages.

The messages are separated by <CR><LF> to facilitate console display. The field format is comma-separated.

The 'Text over Socket' protocol is the minimal mandatory compliance. If additional transport protocols are implemented such as HTTP and JMS, then either Text or XML[3] formats shall be used. If REST or SOAP[4] is implemented, then an XML format is used. The Text format includes comma-separated fields, while the XML format includes 'abbreviated' XML tags. Both formats are described within this part of ISO/IEC 24730.

REST, SOAP and higher functionality services are not mandatory because they tend to limit the message rate.

A goal of this part of ISO/IEC 24730 is to easily support 3,000 messages per second or more. Although in many applications a rate of 3,000 messages per second may not be needed, the RTLS minimal API shall support it because an actual tag deployment with 3,000 tags blinking at 1 Hz may easily produce that message rate. When rate filtering is involved, and/or the IT systems and software are tuned to handle high rates, REST, SOAP and other transports may provide additional functionality. As indicated, the comma-separated Text format shall be considered when supporting high data rates because the format is non-verbose and allows transports to operate at medium to high rates.

This API or protocol is referred to as 'Simple Location Message Protocol', or 'SLMP'. The mandatory comma-separated format is referred to as 'Simple Location Message Format', or 'SLMF'. For specific transports, 'SLMF-Sockets' is the TCP/IP compatible mandatory interface/API of SLMP, while, for example, SLMF-HTTP is an optional interface/API when supporting HTTP as a simple RTLS REST service. As indicated above, an SLMP implementation may optionally include an XML format.

A client application connects to the RTLS using a TCP/IP connection. The RTLS system responds with a stream of messages that stops only when the client connection is closed.

The RTLS system shall send keep-alive massages if the line is silent for long periods of time. This part of ISO/IEC 24730 does not prescribe a particular keep-alive interval, but rather leaves this decision to the RTLS system vendor (see clause 5.7.3). The Client App shall attempt a reconnect periodically if the Socket connection to the RTLS system is lost.

The RTLS system provides the API via a minimal device that collects message from readers, calculates locations, and forwards messages. This device is not required to have persistency and does not need to keep historical data or state of last tag during the active session. Thus, this API does not provide tag status, but rather only tag events. Tag status is left to the application to handle in the context of a business aware database, or for a higher level API not in this scope.

The API herein defined shall support multiple concurrent client connections.

---

3) *Extensible Markup Language (XML) 1.0, (Third Edition)*, W3C Recommendation, World Wide Web Consortium (W3C), 4 February 2004. (http://www.w3.org/TR/REC-xml/)

4) *SOAP Version 1.2 Part0: Primer,* W3C Recommendation, World Wide Web Consortium (W3C), 24 June 2003, (http://www.w3.org/TR/2003/REC-soap12-part0-20030624/)

5.3 Message stream configuration

This part of ISO/IEC 24730 does not prescribe specific methods for filtering the output message stream produced by the RTLS system; however, an RTLS system vendor may implement filtering on a data collection and forwarding device if desired while still complying with this part of ISO/IEC 24730. For example, the vendor could implement a client-side subscription method that passes filter definitions to the RTLS system. The RTLS system could then use the filter definitions to limit message stream output to clients. Alternatively, a vendor could implement server-side configuration of filter definitions that apply to all client connections.

5.4 Security

Security protocols regarding RTLS message exchange are intentionally excluded from this part of ISO/IEC 24730 because security can be addressed using existing security standards and technologies at the communication layer based on preferences and policies of individual customers. For example, in the case of a TCP/IP connection SSH is a security protocol that is easy to implement and is considered compliant with this standard. Likewise, security protocols such as HTTPS and S-HTTP are security protocols that may be implemented on top of this part of ISO/IEC 24730.

5.5 Purpose

The API herein defined provides a standard mechanism for client application to access location-enriched tag blinks from an RTLS system.

5.6 Language independence

The API herein defined specifies a software language-independent interface to the RTLS Service. It does so by using an industry standard protocol, Text over Socket (TCP/IP), to communicate to the RTLS service.

5.7 Architecture

Figure 1 describes the API message exchanges between a client application and the RTLS System. The 24730-1 API shall allow multiple client connections, thus it keeps TCP/IP connection state per client.

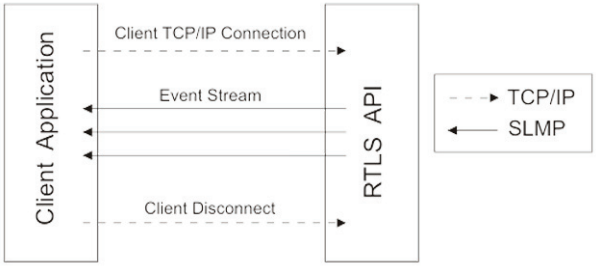

Figure 1 — Architecture of SLMP

5.8 SLMP messages

This clause describes messages that are used within this part of ISO/IEC 24730. Each message type includes a set of fields, which the RTLS vendor shall implement in accordance with the definitions provided herein. All field values and XML tag names that are explicitly defined in this part of ISO/IEC 24730 are case sensitive.

5.8.1 Data types

Data types described in this clause pertain to fields that are associated with messages defined within this part of ISO/IEC 24730. For the Locate message, an RTLS vendor may optionally include extension fields that are not described within this part of ISO/IEC 24730. For such fields, the vendor may choose data types of their choice.

DateTime

This data type represents a date time format as defined in ISO 8601: YYYY-MM-DDThh:mm:ss-hh:mm

Year in the form **YYYY**-MM-DD
Month in the form YYYY-**MM**-DD
Day in the form YYYY-MM-**DD**

T indicates "**T**ime will follow"

Hours in the form **hh**:mm:ss
Minutes hh:**mm**:ss
Seconds hh:mm:**ss**

Plus or minus UTC offset in hours and minutes **(-hh**:**mm or +hh:mm)**

Example: 2010-11-24T09:07:04-08:00 //for PST time zone

Note that a fraction with up to one-tenth millisecond (.0001 seconds) accuracy may be added to the lowest order time element in the representation. For example, to denote "14 hours, 30, minutes, and 12.359 seconds", represent it as "14:30:12.359".

Double

This data type represents a floating-point format that includes an encoded optional decimal point, and may be expressed with or without the exponent and mantissa. Examples include: 2345.334, -98.7, 1.0, 4, 0.0, 0.5, 9.87+E8.

The range for a field of type 'Double' is 1.7E–308 to 1.7E+308, and the maximum string length is 256 characters.

HexBinary

This data type represents structured or unstructured data that can be expressed in hexadecimal format, where each byte is a binary octet. The high order nibble is expressed as the first (leftmost) nibble within an octet, and each HexBinary string shall contain an even number of nibbles.

The maximum field length for a field of type 'HexBinary' is 256 bytes.

Integer

This data type represents numbers that can be written without a fractional or decimal component, and fall within the set {..., –2, –1, 0, 1, 2, ...}.

The range for a field of type 'Integer' is – 2,147,483,648 to 2,147,483,647.

String

This data type represents a set of ASCII characters, limited to the following characters:

A, B, C, D, E, F, G, H, I, J, K, L, M, N ,O, P, Q, R, S, T, U, V, W, X, Y, Z, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, space, !, (,), [,], *, #, $, %, &, +, -, _, ., /, ?, =

The maximum field length for a field of type 'String' is 256 characters.