

Oracle® Hospitality Hotel Property Interface IFC8 FIAS Specification



Release 2.20.25

May 2022

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Hospitality Hotel Property Interface IFC8 FIAS Specification Release 2.20.25

Copyright ©, 2022, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

1 Contents

Preface	7
<hr/>	
2 History	10
<hr/>	
3 Understanding FIAS	16
<hr/>	
Overview	17
Data Bytes Format	17
Data Types	18
Other Notes	18
<hr/>	
4 Field types	20
<hr/>	
5 Time outs waiting for Responses	22
<hr/>	
6 Record-ID types	23
<hr/>	
Communications and Link Control	23
LS - Link Start, LA - Link Alive, LE - Link End	23
LD - Link Description, LR - Link Record	24
Examples	25
LC - Link Configuration	26
Connection Authentication	27
FIAS IFC AuthKey exchange for vendor system connection authentication	27
FIAS Link Description:	28
Parameters for Authentication value encryption:	29
Sample communication:	30
FIAS Implementation Notes & Exceptions	32
Database Synchronization	32
DR - Database Resync request	33
DS - Database Resync start, DE - Database Resync end	34
Examples	34
Night Audit	35
NS - Night Audit Start, NE - Night Audit End	35
Example	35
Guest Data	36
Guest In notification	36

Guest Out notification	38
Guest Data Change notification	38
Examples	40
Virtual Numbers	43
VN Guest in Notification	44
VN Guest Data Change Notification	44
VN Guest Out notification	45
Virtual Number response notification	45
Examples for a Room-based DID handling.	46
Permanent Virtual number Assignment:	47
Extended Guest Data	47
Text Messages:	47
On-Line messages:	48
Message Delete	49
Message Text Online	49
Message Request	49
Message Text	50
Message Delete	50
Guest Bill Request and Bill Item display:	51
Guest Bill Request:	51
Guest Bill Item	51
Guest Bill Balance	52
Remote Checkout Request:	52
Examples	53
Locators	56
Locator On	57
Locator Off	57
Locator Retrieve	58
Examples	58
Room Data	58
DND & Make-Up Room	59
Examples	60
Message Notification	60
Do Not Disturb	60
Class Of Service	60
Voice Mail Notification	61
Room Maid Status	61
Room Status Guest Service Status	61
(Make up Room & Do Not Disturb):	62
Clear Text Info	62
Minibar Right	62
TV Right	62

Building Management Systems	63
Outgoing Room Status	63
Wakeup	64
Wakeup Request	65
Wakeup Clear	65
Wakeup Answer	65
Examples	66
Key Services	66
Overview	66
Key Option / Access Rights	67
Key Request	68
Key Delete	69
Key Answer	69
Key Data Change	69
KeyRead	71
Examples	72
Track2 Data	75
SPA & Other Charge Systems	76
Posting Simple	79
Posting Inquiry/Posting Request	80
Posting List	81
Posting Answer	82
Examples Posting Simple	83
Examples Posting Inquiry / Posting List	85
Examples Posting Request	86
Total Amount calculation	87
Inquiries Using The Magnetic Stripe (Track2) of a Guest's Key-Card	88
POS Cash Rounding Difference Amount Handling	89
7 FIPS – Fidelio Interface Protocol Specification	90
Physical Transmission Layer	90
Serial connections (RS232)	90
Cabling	90
DB-9 Connector	91
DB-25 Connector	91
General Transmission Layer Considerations	92
Data Bytes Format	92
Link Control Bytes Format	92
Other Notes	92
LRC Calculation Example	93
Full-Duplex Protocol	94

Overview	94
Basic Full-Duplex Description	94
Full Duplex Link Control Bytes	95
TCP/IP	97
Overview	97
Framing	97
Client/Server	98
Connection Mode	98
TCP/IP SSL Stream	99
Overview	99
8 Usage of LS LD LA LE Records	99
Overview	99
General Considerations	100
Alive-Check	100
TCP-IP Implementations	101
RS232 communication	103
9 Appendix A - FAQ	108
Frequently Asked Questions:	108
10 Appendix B – Code Tables	110
IF - Interface Types	110
AS - Answer Statuses	110
GL - Guest Languages	111
KT - Key Types	115
PT - Posting Types	115
CS - Class of Service (COS)	115
MR, VR, TV - Guest Rights	116
RS - Room Maid Statuses	116
RT - Request Types	117
RT in LD (Request Values In LC Record)	117
RT in KR	117
RT in XM	118
ES/ET - Equipment Status (DID)	118
11 Appendix C - Field ID	119
Field ID list	119

Preface

Purpose

This guide explains a standard for application record formats and data flows for data communications between an Oracle Hospitality Property Management System (PMS) and a Hotel Property vendor system. It gives a general description of record formats, and data flow requirements and covers specifics for Record Types, Field Types, and Field usage.

For information regarding Oracle's low-level protocol specification and recommendations, please refer to the Oracle Property Interface Protocol Specification - Section of this document.

NOTE:

FIAS is supported by the following PMS systems from Oracle Hospitality:

Oracle Hospitality Suite 8 (any version) & Oracle Hospitality OPERA PMS (>= Ver. 5.x)

Audience

This document is intended for vendor systems integrating with ORACLE Hospitality Property Management system, OPERA, or Suite8 using FIAS Protocol.

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

<https://support.oracle.com>

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (including business impact)
- Detailed step-by-step instructions to re-create
- The exact error message received
- Screenshots of each step you take

Documentation

Oracle Hospitality product documentation is available on the Oracle Help Center at

<http://docs.oracle.com/en/industries/hospitality/>

Table 1 Revision History

Date	Description
November, 2017	Version 2.20.21 <ul style="list-style-type: none">• Conversion from former CHM file• Removed EFT Functionality
July, 2018	Version 2.20.22 <ul style="list-style-type: none">• Corrected wrong field name in Virtual number handling description• Added clarification for mandatory use of RT field in LD record for connection authentication
November, 2018	Version 2.20.23 <ul style="list-style-type: none">• Added Alive Check usage with LS and LA record• Corrected RT value in Connection Authentication example
April, 2020	Version 2.20.24 <ul style="list-style-type: none">• Adjusted Footnote 6/7 for RE record (RoomData) Removed Footnote 7 and replace it with 6• Added Examples for EMS Scheduler records sent (RoomData)• Added FIPS TCP/IP SSL stream connection using certificates• Added +0 field in PostRequest message to send POS Cash rounding difference amount (i.e., when occurring in Postings with rounded Total Amount containing discounts)• Added more details on the handling of Posting Sequence Number (P#) in Posting messages

Date	Description
April 2022	<p data-bbox="911 260 1430 302">Version 2.20.25</p> <ul data-bbox="911 302 1430 798" style="list-style-type: none"><li data-bbox="911 302 1430 365">• Adjusted field value descriptions and length limits<li data-bbox="911 365 1430 449">• Modified use of GN (GuestName) in KeyRequest and KeyDataChange when using the CT (ClearText) field<li data-bbox="911 449 1430 533">• Added note that updating the Guest Name (GN) in PMS will not trigger a KeyDataChange message<li data-bbox="911 533 1430 617">• Adjusted GS values in GuestdataChange example for room move of sharer reservation<li data-bbox="911 617 1430 680">• Added missing GI, GC, GO Record ID to WS Field in Field ID List<li data-bbox="911 680 1430 798">• Corrected missing mandatory field CT in instances for KeyAnswer

2 History

Date	The version number and change description
15 Sep 1994	Version 1.00 – first draft, overview, Record, and Field Types
31 Oct 1994	Version 1.01 – the start of field explanations and tables, new fields for guest rights
8 Nov 1994	Version 1.02 – varying corrections and additions to tables; all revisions between the last and current versions are now marked
20 Dec 1994	Version 1.03a – rough draft, revisions marked
4 Jul 1995	Version 1.04 – clarification of Link Start sequence, add fields for Voice Mail, new language codes, clean up examples (all changes since V1.02 marked)
2 Jan 1995	Version 1.05 – repaginate
29 Mar 1996	Version 1.06 – import into Word 7.0
30 Apr 1996	Version 1.07 – change Key Options functions (currently unused), enhanced EFT records, and field types. All changes from V1.06 are marked.
1 Aug 1996	Version 1.08 – added Virtual/Phantom Extension examples, changes for terminology, changes from V1.07 marked
26 Mar 1997	Version 1.09 – added Locator records, cleaned up examples, all changes since V1.07 still marked
1 Oct 1998	Version 1.10 – significant clean-up/reformatting, added more examples, more tables, started FAQ, only significant changes from 1.08/1.09 marked
9 Jan 1998	Version 1.11 – add PP for messages to be sent to printers, A0-A9 for assignable fields,
Jan 2001	Version 1.50 – added tables (available fields), reformatted document, made further corrections, and added more examples.
Nov 2001	Version 1.51 - reformatted document, made further corrections, and added more examples.
July 2003	Version 1.60 - imported to HTML-Helpfile, reformatted document, made further corrections, and added more examples. Also, removed support for the XO record.
June 2004	BETA-Version 2.00 - reformatted document, made further corrections, and added more examples. Added new fields and records for Chip&PIN handling in EFT Renamed description of 'G#' from Guest Number to Reservation Number to avoid confusion Enhanced length
February 2005	Release of Version 2.00 Added more fields and examples for EFT-Chip&Pin handling
June 2005	Version 2.01 Removed KeyOptions (KO) from Guest Data records (not supported by PMSs anymore) further corrections and added more examples

Date	The version number and change description
	Added DA & TI to EFT settlement records
September 2005	Version 2.01 b Added support for DU and DD to PR
May 2006	Version 2.01 i Made PM in PR always mandatory
March 2007	Version 2.10 Further corrections and explanations to examples Added new record \$V for EFT-Settlements without separate authorization Removed support for FS (Field Separator) in LD Added new Interface Type "MS" (Miscellaneous) Removed support for SM (Seminar Channels) in all related records Added support for RT (RequestType) in LD and support for LC (LinkPmsConfiguration) record Added support for G+ (Profile-ID) in PL Removed support for RA record Added new record VA and redesigned DID handling (added the logic to GI/GC/GO records now) Added support for PU in incoming RE Added CVV Number (\$M) to all EFT records - functionality is not supported by the Fidelio system yet FDX-Protocol: minor corrections to the time-out descriptions removed support for XOF/XON Removed \$2 from PS (can only be supported in PR) Added \$Y - response to \$Q (Cancel Transaction) Removed KO from GI and GC Clarified correct handling of GI/GC/GO WS is now mandatory for all posting related records (PR/PL/PA)
May 2007	Version 2.10a minor corrections in examples for Chip&PIN records
May 2007	Version 2.10b minor corrections in the description of data-types
June 2007	Version 2.11 Added G# to KA Added \$3 to KA Added RT to KA and KD Smaller corrections to key-examples Corrected documentation on RT (RequestType)
November 2007	Version 2.11c - Last Release for IFC7!! minor corrections in field description of posting records
November 2009	Version 2.20 - Release for IFC8 only !! enhancements to examples and descriptions in all areas enlarged P# from 4 to 8 characters Added AS to XB Added examples for On-Line Key systems Added \$1 to KR and KM Added ID to KR, KD, KM Added GN to KR, KM Added IM to \$S, \$V, \$O and \$P Added new EFT-RequestTypes Added new AnswerStatuses Added PP to \$G, \$O CT is now mandatory in KA records Added \$2 to KA

Date	The version number and change description
	Added documentation for KZ removed PD from PR (The same functionality is given through the DD field) Added ID to RE Added RT to XM Added G+ (Profile ID) to GI, GC, PR Corrected size of DD field Added logic to create messages in the Fidelio system through XL Removed KO from KM
July 2010	Version 2.20a added \$W (AVS verification data); future PMS versions will support this feature corrected descriptions of receipt printing for EFT-transactions
July 2010	Version 2.20b corrected time-formatting in KeyRecords
August 2010	Version 2.20c added CS to GI/GC added case studies for complex key-handling
October 2010	Version 2.20d corrected EFT examples added documentation for AVS (\$M) and CVV data (\$W)
January 2011	Version 2.20e corrections to the 'purpose' chapter.
January 2011	Version 2.20f clarification to description of PostingRecords
January 2011	Version 2.20g added G+ to GC corrected description of the NoPost (NP) field in GI/GC
April 2011	Version 2.20h corrected description of data-type "M"
May 2011	Version 2.20i changed length of Roomstatus (RS) to N,2
June 2011	Version 2.20j corrected definition of \$D in Appendix C
June 2011	Version 2.20k added example for a refund with a \$P-record
September 2011	Version 2.20l comments to the length of Room number (RN)
September 2011	Version 2.20m added CT to KR and KM
September 2011	Version 2.20n Done minor corrections and clarifications in descriptions.
October 2011	Version 2.20o corrected examples of GO (removed the unsupported GN from all examples) Added PH (Hotel-ID) to LC
October 2011	Version 2.20p changed length of language (GL)

Date	The version number and change description
December 2011	Version 2.20q updated FAQ section
December 2011	Version 2.20r updated PS/PR examples
February 2012	Version 2.20s updated PS/PR examples
March 2012	Version 2.20t clarification that PR/PL may never be used together with GI/GO
April 2012	Version 2.20u clarification for message handling
May 2012	Version 2.20v clarification for synchronization records
July 2012	Version 2.20w clarification for P# handling in posting records
September 2012	Version 2.20x \$R - changed maximum length from C,10 to C,20
September 2012	Version 2.20y clarification for Key Services
June 2013	Version 2.20z Corrected length description of DD
July 2013	Version 2.20 a1 removed reminding references to FS (Field Separator) in LD - this functionality was removed from the interface program in 2007
July 2013	Version 2.20 a2 Clarification for the usage of DA and TI in NS/NE records Clarification for the use of CO in PR
July 2013	Version 2.20 a3 Removed \$2 from \$K/\$G/\$P records Clarified \$2 in \$G
August 2013	Version 2.20 a4 Corrected length definition of P# in Appendix C
August 2013	Version 2.20 a5 Added documentation about outgoing room status records
September 2013	Version 2.20 a6 Clarification for EFT-handling: either Transaction-Number (\$J) or Audit-Trail-Number (\$F) must be supported.
December 2013	Version 2.20 a7 Corrected description of the time-field (LT) in locator records
March 2014	Version 2.20.1 Further descriptions to posting records with \$2
April 2014	Version 2.20.2 Added IM to \$A
April 2014	Version 2.20.3 changed length of First name (GF) to 80 characters (in all applicable records) changed length of Last name (GN) to 200 characters (in all applicable records)

Date	The version number and change description
August 2014	Version 2.20.4 Corrections to examples and description of CT in RE
January 2015	Version 2.20.5 Enhancements to the description of Bill View functionality
February 2015	Version 2.20.6 Corrected description of DN field in RE record
March 2015	Version 2.20.7 Corrected length definition of BD and DC fields in XI record
July 2015	Version 2.20.8 Adjusted description of Link Alive handling in Correct example of LS/LD/LA/LE section
January 2016	Version 2.20.9 Adjusted description of Link initialization handling in Implementation Notes and Exceptions section minor corrections in the Communications and Link Control section added explanation of PMS handling for Guest Bill View & Remote Check out and its record flow (Extended Guest Data - XR, XI, XB, XC records) Adjusted Answer Status code SV for wake-up usage. This is used only internally by IFC&PMS – and is not to be sent by external systems adjusted Examples for Message delete record XD in the Extended Guest Data section added section Time Out waiting for Response records in Fields and Records section to explain recommended handling on an external system for expecting response records by PMS
March 2016	Version 2.20.10 added description of DN field usage in Room Data section (RE record) added description of Guest Service Status (DND & MUR door sign) in the Room Data section Adjusted Disclaimer Added support for Encryption of sensitive data for EFT functionality. An additional separate Protocol Specification document is required! Details in Section Record-Id types --> EFT ff.
August 2016	Version 2.20.11 adjusted description of Supplemental fields (CT or X1) used in Posting Records (PS or PR)
October 2016	Version 2.20.12 added documentation of EFT CardId field (CI) for EFT token handling, added documentation of RequestType (RT) 4096 for token request in CcUsage record (\$U)
November 2016	Version 2.20.13 adjusted Department Code (DC) field size description for Guest bill item (XI) Message
December 2016	Version 2.20.14 Adjusted list of fields that will be encrypted within EFT Data records.
January 2017	Version 2.20.15 adjusted usage of LS/LD/LA/LE records section
March 2017	Version 2.20.16 Added note that DR DS DE records do not require LinkRecord (LR) at link initialization.
May 2017	Version 2.20.17 Adjusted Data Type section code description.

Date	The version number and change description
	Adjusted Data Format columns in Record ID Types and Appendix C - Field ID and Code Table
August 2017	Version 2.20.18 added notes to Appendix B - Tables GL - GuestLanguages section
September 2017	Version 2.20.19 added new section for vendor connection authentication via Ifc AuthKey handling (to ensure only authorized system connects to IFC TCP port): Go to Record-ID types / Connection Authentication
October 2017	Version 2.20.20 corrected typo in Interface Protocol specification - TCP/IP section

3

Understanding FIAS

Oracle's FIAS (Fidelio Interface Application Specification) is a universal protocol specification used by different kinds of third-party property systems to exchange data.

It allows a system to define its specific data records using the list of available record types and fields to create desired functionality supported within an interface.

Upon startup of the communication, the vendor system must provide record types and fields of the desired records to the interface.

The record type LR is used to do this. The interface application will create related logic and pass it on to the PMS System based on these definitions.

The PMS system will then be able to provide the related information for each action to the vendor. You can refer to this in the section [Communications and Link Control](#) in this document.

To describe the records, use the FIAS Link records, which you can also use in the later communications. Compare this to the XML schema definitions.

This specification lists the allowed field per record type. The definition must be done for all records - both records to the PMS and from the PMS.

Some records used are bi-directional; You must combine the necessary field into one link record.

An example:

RoomEquipment (**RE**) should be used to signal Room Status changes (**RS**) to the PMS and to change class-of-service (**CS**) to the PBX. Both records need the Room Number (**RN**):

Example records:

Set COS (**CS**) to '3' for Room (**RN**) 2781:

```
RE|RN2781|CS3|
```

Maid status notification (**RS**) (clean/vacant) for Room (**RN**) 2781 (default maid statuses are listed in the [Room Maid Statuses Table](#) in Appendix B):

```
RE|RN2781|RS3|
```

So to define a link record for RoomEquipment (RE), send the following:

```
LR|RIRE|FLRNCRS|
```


Overview

This specification is designed to allow for future expansion, either of new records or new fields, using records that are not of fixed length or content. Neither is the fixed position fields (except for the Record ID field). As more information becomes available or no longer necessary, the interface can add or omit fields by configuration.

In most cases, fields are not mandatory; when required, they are noted: tables listing available Field IDs will have mandatory fields in **bold red** typeface.

Defining the mandatory fields is a must in the Link Record for that Record ID.

The PMS works by parsing incoming records according to the Record ID field. If fields contain data that the PMS does not require or use for that Record Type, the data will be parsed over and ignored.

Records should always contain all the data necessary to perform a function. However, defaults for unspecified statuses should be used for many functions, such as Check-in. For example, a Check-in record sent to a PBX should contain the room number, any necessary guest information, and default to opening the phone line. It is not required to specify that the line has to be opened, nor is it necessary to send separate records to support guest information at Check-in. A RoomEquipment (**RE**) record can be sent after Check-In to signal an extension should be blocked; for example, if the guest has no charging privileges.

Data Bytes Format

Records are composed of data bytes and link control bytes. The data portion of a record should not contain any bytes usually reserved for link control (Hex 00 through Hex 1F and Hex 7F). The control characters from Hex 1C through Hex 1F (FS, US, RS) are used by some systems as field separators. For systems using formatted text (guest messages or folios), it is also acceptable to embed such characters as Hex 0A and Hex 0D (LF, CR).

When this is the case, these characters are considered as part of the standard data stream and do not require a preceding escape character (DLE - Hex 10); they are not then available for use as link control characters. The rest of the standard ASCII character set is sufficient (Hex 20 through Hex 7E); however, extended ASCII (Hex 80 - Hex FF) is used to support multiple alphabets. Data passes in an unpacked format; it should not be packed in 'nibbled,' BCD, or other formats. This helps to simplify installation and support.

This specification uses as a field separator the bar character ('|' - Hex 7C). Using a field separator is not necessary to pad fields to their maximum size. The PMS sends all fields without padding, and when fields get transmitted from the other system reference data configured in the PMS (i.e., room numbers, guest numbers, etc.), they should be sent without padding. If padding is done, numeric fields should be right justified, with leading zeroes ('0') except in the case of negative amounts when the leading character is the minus sign ('-').

Data Types

In general, fields are either numeric (decimal digits '0' - '9'), monetary (this includes the decimal numeric characters, plus '-' and "." as necessary), or alpha (all alphabetic letters). Some fields require some combination of these types.

AN - Alphanumeric characters. Correspond to the character set of ASCII Code Page with codes 32-127. No Control Character.

ANS - Complete character set of used Code Page (all printable characters), No Control Character.

N - Numeric characters, includes '0' - '9', the minus sign ('-') as leading character, and where necessary 'A' - 'F' and 'a' - 'f' as hex characters. These fields always reflect integer values (no decimal positions). No Alpha character, No Control character

M - Monetary characters include all numeric characters and period ('.') as decimal indicators where necessary when coming FROM the PMS record TO the PMS must be without decimal indicator. Depending on the regional/local setting and currency, the PMS can handle monetary fields without an implied decimal point depending on the regional/local setting and currency.

D - Date, numeric characters, formatted as YYMMDD

T - Time, numeric characters, formatted as HHMMSS

NOTE:

As the PMS sends and expects to receive fields without padding, leading zeroes or spaces in all alphanumeric fields are considered significant data (i.e., if a room number contains a leading zero or a leading space, this digit is regarded as part of the room number).

Monetary characters:

Posting details are generally expected without decimal indicators. However, EFT-transaction will always contain a based on the configuration of the respective local currency separator.

Other Notes

Low-level ACK/NAK responses are required (only applies to asynchronous serial connections); application-level responses are only necessary where appropriate. The receiving system doesn't need to send an application-level response that a particular action has been performed; in the PMS's case, this response is sent only when the other system requires them. Receiving them carries out meaningful processing only when they need further action.

There shall be an application-level response in most cases where records are rejected at the application level. For example, a posting record that is received correctly but contains bad/invalid data (e.g., unknown room number, the application response would have...**[ASNG|CTINVALID ROOM|]**).

Using a NAK causes immediate retransmission of the same record with only low-level logging of communication errors.

4

Field types

Field Types are two-character IDs (ANS) included at the beginning of each field. This allows the field to be easily identified. Fields have maximum sizes, but transmitting the total field size is unnecessary. All fields have a separation character ('|' - 0x7C; this is the default; see the Communications and Link Control section below). Even if there is no data for a field (i.e., the Record Type field), if the field ID is included, it must have a separation character to indicate the presence of a blank field.

 **NOTE:**

All examples are shown without low-level protocol framing or response characters.

Fields listed in these examples are defined in Record ID Types below. Please note that these are only examples; where fields are not mandatory, they are included to indicate how this specification works, not to restrict the functionality of your system. Field Types in the examples are in **bold typeface** to help identify them. The left arrow symbol '←' indicates this record is sent **from** the PMS, and the right arrow symbol '→' that the record is sent **to** the PMS.

Example

GI|RN103|GNMr. Rogers|

GI - Check-in

RN - Room number: 103

GN - Guest Name: Mr. Rogers

There are only a few strict requirements for which fields must be included or allowed in any given record. Even though a field is requested, if it does not have a logical use within the context of that Record Type, it might not appear in the actual records sent, or it may go with no data (i.e., immediately followed by a field separator).

Note that the content of many Field Types is configurable within the PMS (e.g., **GN**, **GV**, etc.) and, as a result, may vary from site to site.

It is beyond the scope of this document to describe all the possible usages of the fields listed below. Please contact ORACLE Hospitality if you have questions about specific fields.

 **NOTE:**

The order of the fields is usually linked to the order in the received link record but can be changed at any time. The receiving system needs to be prepared to receive records with different field orders.

5

Time outs waiting for Responses

Incoming FIAS commands from the external system which require a response by PMS:

- **LA** - Link Alive after initialization
- **DR** - Database sync request
- **XM** - Guest message request
- **XR** - Guest bill request
- **XC** - Remote Check-out request
- **LP** - Locator Retrieve
- **PS** - Posting (simple)
- **PR** - Posting Request as Inquiry or Posting

The PMS responses to these requests vary and might take longer due to the complexity of the request or with more significant sites with high data traffic. The external system should implement time-outs while waiting for the response messages for the above commands.

If an external system does not receive the expected response message by PMS, it should time out at this stage and stop waiting for a response.

The minimum default time-out should be 30 seconds and 60 seconds for PR commands. It is recommended to make the time-out value configurable to fit every case.

6

Record-ID types

The first field in all records is the **Record ID**. There is no data for this field; the Record ID is followed immediately by the field separator character, Field Types, and relevant data.

The IDs for the Record Types currently supported, grouped in logical or functional families, are listed below.

Communications and Link Control

LS - Link Start, LA - Link Alive, LE - Link End

These Record Types are used to control the status of the link. The PMS only opens or closes the link when starting or stopping its software. This means that if a Link Start (**LS**) is received from the PMS, the Link Description (**LD**) and Link Records (**LR**) must be re-transmitted (see Implementation Notes & Exceptions below).

The Link Alive (**LA**) record is provided to verify the link is still functioning. The PMS only uses this Record ID to respond to a Link Start (**LS**) or a Link Alive (**LA**) when the link is or was previously active (see Implementation Notes & Exceptions below and refer to "Usage of **LS/LD/LA/LE** Records").

However, suppose the other system sends a LA record as a test of the link. In that case, the PMS will send a low-level acknowledgment (only applies to asynchronous serial connections, see the ORACLE Interface Protocol specification for further details). The other system should recognize this as a signal that the PMS interface software is running; an application-level response is not sent.

If the PMS sends a Link End (**LE**) record, the other system should buffer all non-discardable records (i.e., charges) until it receives the subsequent communication. At that point, the link should be reactivated even if the Link Start (**LS**) record is missed.

Consider possible delay in response from PMS for LA record at initialization sequence. See related note - [Time outs waiting for Responses](#)

Record ID	Field ID	Description	Format	Direction
LS, LA, LE	DA	Date	D	Both
LS, LA, LE	TI	Time	T	Both

LD - Link Description, LR - Link Record

The other system must send these records immediately after it receives the Link Start (**LS**) record from the PMS upon startup or initialization. Please note that it is possible to re-configure the link at any time.

The link description (**LD**) record indicates the Link Records (LRs) start and general link information. The other system sends link Records (LRs) to describe each record it will be sending and expects to receive; this is a Record ID Type, followed by a list of fields that should be included (for that particular Record ID), one Record ID per Link Record (**LR**).

Additionally, the field **RT** (RequestType) can be used in the LinkDescription to retrieve specific information from the ORACLE interface. For example, the configured timeouts - are sent in the Link Configuration (**LC**) record or can be used to force connection authentication in combination with the Cryptogram (**CG**) field.

Note that in the examples below, the fields requested may not match the order they are sent in the record; field order is not considered significant. After the last Link Record (**LR**), the other system should send a Link Alive (**LA**) to indicate that the link is now considered active.

Link Description

Record ID	Field ID	Description	Format	Direction
LD	DA	Date	D	To PMS
	IF ¹	Interface Family	AN, 2 chars (See Interface Type Table)	To PMS
	TI	Time	T	To PMS
	V#	Vendor System's Version #	AN, max 10	To PMS
	RT ²	RequestType Combination of any of the below values: 1 - Request the configured Room payment methods in RP 2 = <i>not used</i> 4 = Request PMS version in A1 8 = <i>not used</i> 16 = Request IFC version in A2 32 = Request IFC Driver Version in A3 64 = <i>not-used</i> 128 = Request DLS-timeout in A0 Mandatory for Connection Authentication use. Must be 1 or greater in combination with the CG field to force connection authentication verification with the IFC application	N, max 4 The Fidelio interface reads this value as Bit-driven, so any combination of the values can be used by adding them together. PH (HotelID) is returned if RT is set in LD	To PMS

Record ID	Field ID	Description	Format	Direction
	RL	The maximum record length for message records (Do not confuse with general maximum record length. This is C, 2000)	N, variable, max. 2000	To PMS
	CG	Cryptogram for secure connection with IfcAuthKey - see section Connection Authentication for EFT data encryption functionality - see additional FIAS Encryption specification	AN	To PMS

1 determines the display of the PMS Interface system and enables corresponding functionality in the PMS.

2 required when IFC connection authentication functionality is used - in correspondence with **CG** field.

required when EFT functionality is used – in correspondence with **CG** field – for encryption key exchange for encrypting sensitive data

Link Record

Record ID	Field ID	Description	Format	Direction
LR	RI	Record Indicator	ANS, 2 chars	To PMS
	FL	Field List	ANS, variable	To PMS

NOTE:

The order of fields is fixed and may not be altered. A Linkrecord MUST look like: **LR|RIxx|FLxxxx|**

LR records for **Database Resync (DR, DS, DE)** are not required to send during the initialization sequence.

The record definition is hardcoded within the IFC application and does not explicitly need to be defined by the LR record.

Examples

The following is an example of both systems starting at the same time. The data flow should be followed exactly, except for the Link Records (LRs) format. These are sent as required by the functionality of the other system.

The PMS sends a Link Start (**LS**) record with the date (**DA**, 15 October 2000) and time (**TI**, 12:30:45 PM) fields. This indicates that the PMS software has been restarted, and

the other system must send any configuration records (**LD/LR/LA**) before sending any buffered data records:

LS|DA001015|TI123045|

The other system responds with a Link Description (**LD**) with vendor version # (**V#**) 1.01 and interface type (**IF**) PBX:

LD|DA001015|TI123046|V#1.01|IFPB|

It sends a Link Record (**LR**) with a Guest Check-in field list (**RIGI**). Requested fields are Room Number (**RN**), Guest Number (**G#**), Guest Name (**GN**), Guest Language (**GL**), Guest VIP status (**GV**), and Guest Group number (**GG**), with support for multiple guests (Guest Share, **GS**), include Swap Flag (**SF**) in database resync records:

LR|RIGI|FLRNG#GNGLGVGGSSF|

Link Record (**LR**) with Guest Change (**GC**) field list – requested field list is the same as Guest Check-in (**RIGI** above) except for the **SF** field (**GC** records are not sent as part of a database resync and don't use the Swap Flag) and the **RO** field (used in Room Move records):

LR|RIGC|FLRNG#GNGLGVGGSSRO|

Link Record (**LR**) with Guest Check-out field list (**RIGO**) – requested fields are Room Number (**RN**), Guest Number (**G#**), Guest Share (**GS**), and Swap Flag (**SF**):

LR|RIGO|FLRNG#GSSF|

Note: Guest Check-out records (**GOs**) sent during database resync will not contain any fields other than Room Number (**RN**) and the Swap Flag (**SF**), as there is no valid data for other fields (see database swap example below).

After the last Link Record (**LR**), the other system should send a Link Alive (**LA**) record. This indicates that the other system has sent descriptions of the link and all Record Types it wants to receive or send. The link is now active, and the PMS will immediately start sending any real-time or buffered data:

LA|DA001015|TI112349|

The PMS responds with a Link Alive (**LA**) as the link was inactive before:

LA|DA001015|TI112350|

LC - Link Configuration

Record ID	Field ID	Description	Format	Direction
LC	A0	EFT-Timeout and DLS-Timeout values from the PMS	N, separated by a semicolon	From PMS
	A1	PMS Version number	AN, variable	From PMS
	A2	IFC Version number	AN, variable	From PMS
	A3	IFC driver version	AN, variable	From PMS

	PH	Hotel-ID	ANS, max 20	From PMS
	RP	RoomPaymentmethods as defined in the PMS	AN, variable, values are comma-separated	From PMS
	CG	Cryptogram (Only for EFT functionality) see additional FIAS Encryption specification	ANS	From PMS

The information in the LinkConfiguration record is requested through the **RT** field in the **LD** record:

●LD|DA001015|TI123046|V#1.01|IFPB|RT183|♥

☺LC|RP16,ROOM|A060;10|A16.2031|A27.43.23|A37.02|PH1|♥

In this example, the following information is transferred:

The configured RoomPayment methods are: "16" and "ROOM"

The EFT Timeout is 60 seconds

The DLS Timeout is 10 seconds

The Fidelio version is 6.20.31

The interface version is 7.43.23

The driver's version for FIAS is 7.02

The Hotel-ID is 1

Connection Authentication

FIAS IFC AuthKey exchange for vendor system connection authentication

General Information

FIAS specification offers a vendor connection on TCP/IP with secure authentication via exchanging AuthKey to ensure the correct system will connect to the IFC application.

This will enhance the security in communication with the vendor system to avoid data breaches by letting a system connect, which would not be expected.

Alternatively, FIAS also offers a secure TCP/IP communication using [TCP SSL Stream](#). In this case, the whole communication will be secured by exchanging SSL certificates between the Interface application and the vendor system. A specific Hotel Property Interface version is required.

General flow

- Vendor creates AuthKey
- AuthKey will be handed over to the person configuring the Oracle IFC and entered into the IFC application configuration.
- The vendor sends RequestType value plus encrypted string (String value = the Date and Time stamp of last received Link Record – sample: LS|DA170719|TI132450|) by lfcAuthKey encryption in FIAS LD Record CG field.
- IFC8 verifies the string using the entered lfcAuthKey.
- When verification is successful, initialization/connection continues.
- When not ok, IFC8 will stop initialization and connection. IFC8 must be restarted manually, and it will not allow new connections to avoid loops.

The lfcAuthKey

1. For authentication of the system, a static AES key is used, which is used to crypt the public RSA key before it is passed over the LAN. This key is called “lfcAuthKey.”

The vendor must create this key.

It must be handed over to the person configuring the related Oracle Interface so he can insert this key into the Interface application.

This “lfcAuthKey” must be stored encrypted and secure on both “sides” of the communication systems.

On Windows-based systems, it is recommended to use the DPAPI encryption method.

A sample string format of an lfcAuthKey to hand over:

lfcAuthKey:

AES: FidCrypt0S|GVDpVnl6qYITQXQJZxXdbw==

format will be = [Crypt Header: FidCrypt0S=simple Base64] [Separator:] [AES AuthKey]

2. The crypt header for the lfcAuthKey is: “FidCrypt<XY>,” where <XY> is a two-character value as described here:

XY= Crypting method:

0S = indicates simple Base64 encoding method (index0)

“FidCrypt0S;...” for simple Base64-encoding

FIAS Link Description:

Following is a list of available attributes in the FIAS LinkDescription record.

Please refer to the ORACLE FIAS Protocol section for full record and field details.

LinkDescription / LD:

FIAS Field code	Possible values
DA	System Date
TI	System Time
IF	Interface Type
V#	Version of the vendor system
RT	Request Type (value = not empty) see Request Type table
CG The mandatory field for lfc connection authentication	encrypted Date and Time stamp of last received Link record with prefix Crypt header AES/Base64 (FidCryptAB ...)

 NOTE:

If there is no CG value sent within the LD record, the IFC application will not verify if the connection with the vendor is authorized (Less secure). In case the IFC application has lfcAuthKey entered in its configuration. Still, no RT field is sent within the LD record (even CG value is sent) IFC application will terminate as the RT field is missing for proper authentication verification.

Parameters for Authentication value encryption:

AES:

KeyBitLen = 128

PaddingMode = PKCS7

BlockSize = 128

CipherMode = CBC with a random IV equal in length to the block size

The IV will be re-issued for each (!) encrypted field and transmitted as Prefix (see underlined in sample):

“FidCryptAB;3k57hsoHm04fGEyaA3+UVw==oNTebC0J36LY8GV9azyhzw==”

The above values are minimum and can be changed due to Oracle Approved Security Standards.

Example if implemented via .Net Framework:

```
Aes AesCrypt = new AesCryptoServiceProvider { Mode =
CipherMode.CBC, BlockSize = 128, Padding = PaddingMode.PKCS7,
KeySize = 128 };
```

Sample communication:

Sample with AES Key:

Ifc AuthKey provided to the Oracle consultant:

FidCrypt0S|GVDpVnl6qYITQXQJZxXdbw==

Connection Initialization with valid AuthKey exchanged.

<- sent by IFC application to the vendor

-> received by IFC application from vendor

[08.31/10:22:01]#1/0 <- _LS|DA170831|TI102201|_

[08.31/10:22:01]#0/0 ->

LD|DA170831|TI102201|IFPB|V#1.13|RT4|CGFidCryptAB;3k57hsoHm04fGEyaA3+UVw==oNTebC0J36LY8GV9azyhzw==|

[08.31/10:22:01]#0/0 -> _LR|RIRE|FLRNDNMLCSVMRTRSID|_

[08.31/10:22:01]#0/1 -> _LR|RIPS|FLRNRTTADUDDPTM#MAIDX1SOPXMPDATI|_

[08.31/10:22:01]#0/2 -> _LR|RIWR|FLRNDATIRT|_

[08.31/10:22:01]#0/3 -> _LR|RIWC|FLRNDATIRT|_

[08.31/10:22:01]#0/4 -> _LR|RIWA|FLRNRTASCTDATI|_

[08.31/10:22:01]#0/6 -> _LR|RIGI|FLRNG#GNGLGSENEPESSF|_

[08.31/10:22:01]#0/7 -> _LR|RIGC|FLRNG#GNGLGSROEOEIETRTENEPES|_

[08.31/10:22:01]#0/8 -> _LR|RIGO|FLRNG#GSENEPESSF|_

[08.31/10:22:01]#0/9 -> _LR|RIPA|FLRNASCT|_

[08.31/10:22:01]#0/0 -> _LA|DA170831|TI102201|_

[08.31/10:22:02]#2/0 <- _LC|DA170831|TI102202|RP116|A28.10.2.20|A31.13|A9IFC8 PMS Simulation|_

[08.31/10:22:02]#1/0 <- _LA|DA170831|TI102202|_

The Cryptogram value sent to IFC in LD record CG field:

FidCryptAB;3k57hsoHm04fGEyaA3+UVw==oNTebC0J36LY8GV9azyhzw==

Data	Datatype	Notes
FidCryptAB	Crypt Header	Indicates AES Base64 encoding method
;	Field separator	
3k57hsoHm04fGEyaA3+UVw==	Vector data	

oNTebC0J36LY8GV9azyhzw==	Key data	containing encrypted Date/Time stamp from last received Link Record “_LS DA170831 TI102201 _” Value = “170831102201”
--------------------------	----------	--

Using a test tool to verify the correct encryption/decryption-

Sample behavior when an Invalid Authkey string is received from the vendor (not expected by IFC):

<- sent by IFC application to the vendor

-> received by IFC application from the vendor

[07.19/13:40:41]#1/0 <- _LS|DA170719|TI134041|_

[07.19/13:40:42]#0/0 ->

_LD|DA170719|TI134041|IFPB|V#1.13|RT4|CGFidCryptAB;wrs43icYcDOz0+7U+hsoug=
=8fLwUEOfzhDv9yg4z0lhw==|_

[07.19/13:40:42]#0/0 -> _LR|RIRE|FLRNDNMLCSVMRTRSID|_

[07.19/13:40:42]#0/1 -> _LR|RIPS|FLRNRTTADUDDPTM#MAIDX1SOPXMPDATI|_

[07.19/13:40:42]#0/2 -> _LR|RIWR|FLRNDATIRT|_

[07.19/13:40:42]#0/3 -> _LR|RIWC|FLRNDATIRT|_

[07.19/13:40:42]#0/4 -> _LR|RIWA|FLRNRTASCTDATI|_

[07.19/13:40:42]#0/6 -> _LR|RIGI|FLRNG#GNGLGSENEPESSF|_

[07.19/13:40:42]#0/7 -> _LR|RIGC|FLRNG#GNGLGSROEOEIETRTENEPES|_

[07.19/13:40:42]#0/8 -> _LR|RIGO|FLRNG#GSENEPESSF|_

[07.19/13:40:42]#0/9 -> _LR|RIPA|FLRNASCT|_

[07.19/13:40:42]#0/13 -> _LA|DA170719|TI134042|_

As invalid or no Authkey detected IFC application will show Error Message:

[07.19/13:40:42] <MessLvI3> Not authentic Cryptogram received

[07.19/13:40:42] <MessLvI3> ChangeCommState:Off

[07.19/13:40:42] <Error>

-Code: 110 (Parse_IfcAuthKey)

-Source: ParseClass/ParseFiasClass/Send

-Description: IfcAuthKey is invalid

IFC application will get stuck, and IfcAuthKey must be manually re-entered in IFC Configuration by a user.

The connection will be stopped, and connecting to a defined port is impossible.

Previously entered already encrypted AuthKey will be available to choose from in the pick box.

After confirming the change in IFC application configuration, connection to the defined Port is possible again.

FIAS Implementation Notes & Exceptions

The PMS will send a Link Start (**LS**) as its first message when initializing its software once the communication port has been opened. The other system should respond with a Link Description/Link Record(s)/Link Alive (**LD/LR/LA**) sequence.

If the PMS does not receive a response to the Link Start (**LS**), especially the Link Description (**LD**) and Link Records (**LR**), it will retransmit a Link Start (**LS**) upon receiving the first record from the other system. The other system must respond with the above sequence (**LD/LR/LA**) whenever it receives a Link Start (**LS**) from the PMS. (Note, this can only happen on an RS232 connection without a handshake. The PMS interface could detect the disconnect from the other system on the communication layer in different situations).

The other system should always open the communication port upon startup and listen for a message from the PMS for at least 3 seconds. The other system might send an LS record if no message was received. If a message from the PMS was received (usually a Link Start (**LS**) record), the other system should react to that message accordingly, typically sending **LD** and **LRs/LA**.

If the PMS receives a Link Start record (**LS**), it responds with a Link Start (**LS**) if the link has never been started. The other system should then transmit the **LD/LR/LA** sequence.

The functionality of the PMS if it sends a Link Start (**LS**) and receives a response other than a Link Start/Link Description/Link Records/Link Alive (**LD/LR/LA**) sequence is undefined.

Subsequently, the PMS would send no data record, and incoming data records would be ignored. Once the proper init sequence has been received and LinkAlive status has been reached, the PMS will start processing data records.

The system dropping the link should transmit a Link End (**LE**) for a normal shutdown. However, in exceptional situations (hardware or software failure or user error), the PMS will consider the link inactive if there are consecutive low-level time-outs (no response from the other system) exceeding a configurable count. The PMS will buffer what it considers critical data. The other system can request a database synchronization (**DR**) once the communication has been re-established for recovery.

If the PMS considers the link inactive (i.e., Link End (**LE**) from the other system, a disconnected communication port, or excessive low-level time-outs), it will close the respective communication port, reopen it, and send an LS waiting for the other system to reconnect.

Database Synchronization

DR - Database Resync request

DS - Database Resync start

DE - Database Resync end

These records are used to request an initialization or refresh of the system database and indicate the start or end of that resync. The PMS regards its databases as the 'master copy' with few exceptions. The PMS can intermix database records with real-time records, and the **DS** and **DE** records ensure that the other system knows its request has been correctly received and that all database resync information has been sent.

The records sent as part of the database resync are the same as those sent during real-time situations with the addition of the swap flag field (**SF**); this allows the other system to determine the difference between the resync records and real-time messages. Resync records will contain the swap flag field (**SF**); real-time records will not. It is strongly recommended that database resyncs are supported.

However, external systems may NOT send any records during the database swap to ensure the integrity of the data. Mainly, any message record must be held until the DE record is received.

 **NOTE:**

A Database-Swap may NOT be requested after every startup. It puts significant overhead on communications, especially at larger installations. It should only be requested if data is not synchronous anymore. NEVER request Database Swap requests periodically.

A good rule should be:

request a swap when the database was wiped out, e.g., during a new installation

request a swap when the database was wiped out, e.g., due to a hardware failure

Check-In/-Out commands are queued on the Fidelio side. There is NO need to request a Swap only because the system was restarted.

 **NOTE:**

Newer Fidelio systems have been modified to avoid unnecessary Swap-Requests being ignored without any notification.

Consider possible delay in response from PMS for **DR** record request. See related note - [Time outs waiting for Responses](#)

DR - Database Resync request

Record ID	Field ID	Description	Format	Direction
DR	DA	Date	D	To PMS
	TI	Time	T	To PMS

DS - Database Resync start, DE - Database Resync end

Record ID	Field ID	Description	Format	Direction
DS, DE	DA	Date	D	From PMS
	TI	Time	T	From PMS

LR records for Database Resync (DR, DS, DE) are not required to send during the initialization sequence.

The record definition is hardcoded within the IFC application and does not explicitly need to be defined by the LR record.

Examples

The other system requests a database resync (**DR**):

→ **DR|DA001005|TI125045|**

The PMS responds with a start (**DS**), data (i.e., **GI** and **GO**), and end (**DE**) records. This example assumes that the other system only requested the Room Number (**RN**), Reservation Number (**G#**), and Swap Flag (**SF**) fields in the Link Record (**LR**) describing the Guest In (**GI**) and Guest Out (**GO**) records during the link startup sequence (i.e., **LRGI|FLRNG#GSSF, LRGO|FLRNG#GSSF**):

← **DS|DA001005|TI125047|**
 ← **GI|RN1001|G#12345|GSN|SF|**
 ← **GO|RN1002|GSN|SF|**
 ← **GI|RN1003|G#12002|GSN|SF|**
 ← **GO|RN1004|GSN|SF|**
 ← **GI|RN1003|GSY|G#12329|**
 ← **GI|RN1005|G#12234|GSN|SF|**
 ← **DE|DA001005|TI1252001|**

NOTE:

The sixth record sent in this example is a real-time check-in record; the last record received for any room, or guest always reflects the current status. Also, there is no **G#** included in **GO** as these rooms are empty. In addition, at the end of a database resync that is guest-oriented (i.e., the **GI** records contain the Reservation Number, **G#**), if the other system has not received a **GI** record during the resync for a previously checked-in guest, but the room is still

occupied in its system by another guest, the missing guest has checked out and should be deleted from the other system's database.

It is recommended that during DB-Swap, no records are sent to the PMS interface, as a possible response (e.g., a **PA** towards a received **PS**) may not be returned as the next record but only after some other DB-Swap records.

Night Audit

NS - Night Audit Start, NE - Night Audit End

NS - Night Audit Start

NE - Night Audit End

These two records notify other systems about the time frame when the nightly procedures in the ORACLE PMS system are executed.

E.g., EFT systems should simultaneously use these records to run end-of-day procedures to match reports.

It should be taken into account that standard PMS practice is to accept the time of posting as sent by the other system but to replace the date of postings with the 'Hotel' Business date (as opposed to the calendar date).

As a result, postings between midnight and 'Night Audit' are listed as revenue of 'yesterday.'

Record ID	Field ID	Description	Format	Direction
NS, NE	DA	Date	D	From PMS
	TI	Time	T	From PMS

Example

← NS|DA130425|TI030400|

← NE|DA130425|TI032500|

NOTE:

The date & time fields in the night-audit records have no relation to the ORACLE PMS system date but are usually sent with System-date/-time.

The fact of the Night-Audit record is the trigger that signals that the nightly routines are running. The date has no relevance.

Guest Data

GI - Guest Check-in

GO - Guest Check-out

GC - Guest data change

These records transmit data concerning guests: any information required to set or update the guest data will be included in these records. The records can contain similar data fields, but the Record Type specifies the actions required to be performed.

A **GI** record for a previously empty room, i.e., the record contains a Guest Share flag, **GS** set to 'N,' sent as an online message (does not contain the Swap Flag, **SF**) should set all statuses as specified in the record (unspecified statuses should have defaults).

A **GI** record with a Swap Flag (**SF**) should only be used to compare statuses and update what has changed; it should not set unspecified statuses to their defaults. This is also true of **GC** records. Only statuses listed in the record should be changed. All other statuses should remain at their current settings.

NOTE:

If multiple guests per room (Sharers) are supported, it is required to use the Reservation Number (**G#**) and Guest Share (**GS**) fields; this is to prevent overwriting current guest data. Reservation Number (**G#**) is a unique number (assigned in the PMS) that provides a means of identifying guests, even during name changes. It is recommended for use with all systems; it is required for systems that provide multi-occupancy features (Sharers) or can change guest-related information after check-in.

Another item to be aware of is name format; when Guest Name (**GN**) is used, the format of the name is configurable in the PMS.

Specific fields (i.e., **TV**, **MR**) are supported here; however, it is more common to have them defined in room-oriented records, as the **RE** record would be used just to change rights. Please see [Room Equipment \(RE\)](#) section below for further details.

The NoPost (**NP**) field is of pure informational status. It does NOT mean that an extension should be barred. Barring & Unbarring is handled through the respective right (e.g., **CS** or **TV**)

Profile-IDs (**G+**):

The ID is of informational purpose only at this point. Please note that the ID is NOT available in the Check-Out record (**GO**) because a profile cannot be checked out, just a reservation.

Guest In notification

Record ID	Field ID	Description	Format	Direction
GI (Guest Check-In)	G# 1	Reservation Number	N, max. 10	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	GS 1	Share Flag	AN, 1 char (Y/N)	From PMS
	A0 - A9 2,3	User Definable Fields	ANS, variable	From PMS
	CS	Class of Service	ANS, max. 1 (see Class of Service table)	From PMS
	DA	Date	D	From PMS
	G+	Profile Number	AN, max 10	From PMS
	GA	Guest Arrival Date	D	From PMS
	GD	Guest Departure Date	D	From PMS
	GF	Guest First Name	ANS, max. 80	From PMS
	GG	Guest Group Number	AN, max. 10	From PMS
	GL	Guest Language	ANS, max 10 (see Guest Language table)	From PMS
	GN	Guest Name	ANS, max. 200	From PMS
	GT	Guest Title	ANS, max. 20	From PMS
	GV	Guest VIP Status	AN, max. 20 (normally numeric values)	From PMS
	MR 2,4	Minibar Rights	ANS, 2 chars (see Guest Rights table)	From PMS
	NP 5	No Post Status	Y/N (Do NOT use to bar/unbar an extension.)	From PMS
	SF	Swap Flag	No data (if this field is sent, the record is part of the database swap)	From PMS
TI	Time	T	From PMS	
TV 2,4	TV Rights	ANS, 2 chars (see Guest Rights table)	From PMS	
VR 2,4	Video Rights	ANS, 2 chars (see Guest Rights table)	From PMS	
WS	Workstation ID	ANS, max. 16	From PMS	

1 – mandatory for guest-oriented systems

2 – requires special configuration in PMS

3 – the data expected in these fields may not be available in every installation. ORACLE recommends not to base any business logic on these fields

4 - not available with all PMS systems, requires IFC version 8

5 - The PMS NoPost status is of pure informational status. It does NOT mean that an

extension should be barred. Barring is handled through the respective right (e.g., **CS** or **TV**)

Guest Out notification

Record ID	Field ID	Description	Format	Direction
GO (Guest CheckOut)	G# ^{1,2}	Reservation Number	N, max. 10	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA- PMS)	From PMS
	GS ¹	Share Flag	AN, 1 char (Y/N)	From PMS
	DA	Date	D	From PMS
	SF	Swap Flag	No data (if this field is sent, the record is part of the database swap)	From PMS
	TI	Time	T	From PMS
	WS	Workstation ID	ANS, max. 16	From PMS

1 – mandatory for guest-oriented systems

2 – may not be available during database swap

NOTE:

It is not possible or intended to send the guest's name (**GN**) in a check-out (**GO**) record. The Check-out record (**GO**) is intended to remove all existing information from an extension/room on the vendor system. Only RoomNumber (**RN**) and possibly the Reservation-ID (**G#**) as unique identifiers are necessary to perform this functionality.

Guest Data Change notification

Record ID	Field ID	Description	Format	Direction
GC (Guest Info Change / Name Change /	G# ¹	Reservation Number	N, max. 10	From PMS
	RN	Room Number (destination room during room move)	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS

Record ID	Field ID	Description	Format	Direction
RoomMove)	GS ¹	Share Flag	AN, 1 char (Y/N)	From PMS
	A0 - A9 ^{2,3}	User Definable Fields	ANS, variable	From PMS
	CS	Class of Service	ANS, max. 1 (see Class of service../CS_-_Class_of_Service_(COS).htm table)	From PMS
	DA	Date	D	From PMS
	G+	Profile Number	AN, max 10	From PMS
	GA	Guest Arrival Date	D	From PMS
	GD	Guest Departure Date	D	From PMS
	GF	Guest First Name	ANS, max. 80	From PMS
	GG	Guest Group Number	AN, max. 10	From PMS
	GL	Guest Language	ANS, max 10 (see Guest Language table)	From PMS
	GN	Guest Name	ANS, max. 200	From PMS
	GT	Guest Title	ANS, max. 20	From PMS
	GV	Guest VIP Status	ANS, max. 20 (normally numeric values)	From PMS
	MR ^{2,4}	Minibar Rights	ANS, 2 chars (see Guest Rights table)	From PMS
	NP ⁵	NoPost Status	Y/N	From PMS
	RO ⁶	Old Room Number (the source room)	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
TI	Time	T	From PMS	
TV ^{2,4}	TV Rights	ANS, 2 chars (see Guest Rights table)	From PMS	
VR ^{2,4}	Video Rights	ANS, 2 chars (see Guest Rights table)	From PMS	
WS	Workstation ID	ANS, max. 16	From PMS	

1 – mandatory for guest-oriented systems

2 – requires special configuration in PMS

3 – the data expected in these fields may not be available in every installation. ORACLE

recommends not to base any business logic on these fields

4- not available with all PMS systems, requires IFC version 8

5 - The PMS No-Post status is of pure informational status. It does NOT mean that an extension should be barred. Barring is handled through the respective right (e.g., **CS** or **TV**)

6 - mandatory for systems that support room moves opposed to C/O of the old room and C/I of the new room

Examples

Guest In

Check-in (**GI**) for Room (**RN**) 2781, Reservation Number (**G#**) 12345, Guest Name (**GN**) Mr. Guest, Language (**GL**) English, VIP status (**GV**) 3, Group Number (**GG**) A123, non-share (**GS**) to an unoccupied room (**GSN**):

← **GI|RN2781|G#12345|GN**Guest, Mr.|**GLEA|GV3|GGA123|GSN|**

NOTE:

It is possible on the ORACLE PMS side to incorporate the Guest's title and the first name into the **GN** field. However, it is recommended to use the respective fields separately.

Guest Data Change

Change guest information (**GC**) for Room (**RN**) 2781, Reservation Number (**G#**) 12345, Guest Name (**GN**) is now Hr. Gast, Language (**GL**) German, all other statuses remain the same:

← **GC|RN2781|G#12345|GN**Gast, Hr.|**GLGE|**

Guest In Sharer

Check-in (**GI**) for Room (**RN**) 2781, Reservation Number (**G#**) 12381, Guest Name (**GN**) Dr. Sharer, Language (**GL**) English, VIP status (**GV**) 0, Group Number (**GG**) A123, to an occupied room (**GSY**):

← **GI|RN2781|G#12381|GN**Sharer, Dr.|**GLEA|GV0|GGA123|GSY|**

Move (**GC**) Reservation Number (**G#**) 12345 from Room (**RO**, source room) 2781 to Room (**RN**, destination room) 9327. The Guest Share (**GS**) flags indicate the new room is unoccupied, but the old room is still occupied. The room move should be treated as a Check-in for the new room, but the only effect on the old room would be to remove the information for Reservation Number (**G#**) 12345:

← **GC|RN9327|GSN|G#12345|GN**Guest, Mr.|**GLEA|GV3|GGA123|RO2781|GSY|**

 **NOTE:**

It is the responsibility of the receiving system to properly set or change statuses when moving a guest from a share or to a share. It is also expected that if a guest is moved from a room that is now empty, this will function the same as a GO record; if the guest is moved to a previously unoccupied room, all statuses, Wake-up calls, etc. will be transferred accordingly.

Database resync update for Room (RN) 9327/Reservation Number (G#) 12345 and Room (RN) 2781/Reservation Number (G#) 12381, with a refresh of available statuses:

← **GI|RN9327|G#12345|GN**Gast, Hr.|**GLGE|GV2|GGA123|GSN|SF|**

← **GI|RN2781|G#12381|GN**Sharer, Dr.|**GLEA|GV0|GGA123|GSN|SF|**

Guest Out

Check-out (**GO**) Room (**RN**) 9327, Reservation Number (**G#**) 12345, no sharing situation exists in the old room (**GSN**):

← **GO|RN9327|G#12345|GSN|**

Case studies:

Globally two different concepts need to be understood:

The ORACLE PMS allows situations where more than one guest/reservation is checked into the same room. A vendor system using FIAS needs to decide if he can support such a 'guestbased' system - or if he prefers a 'room-based' mode.

The selection of GI/GC/GO fields should be taken accordingly. The main-logic centers on the fields **G#** (Reservation-ID) and **GS** (Share flag). **G#** should only be used by systems that can genuinely separate between reservations and store and handle the different IDs. Additionally, **GC** for room moves should only be used by a system that can truly support the moving of exiting guest data from one extension to another. Else the record should not be used - or only be used for updates to guest names.

It is recommended to use the ShareFlag (**GS**) for Roombased systems.

Room-based approach

(typically used by PBX or BMS systems):

Sample link records:

→ **LD|DA081013|TI151544|V#2.5|IFPB|**

→ LR|RIGI|FLRNGNG#SF|

→ LR|RIGO|FLRNG#SF|

→ LA|DA081013|TI151544|

A check-in of sharing reservations to room 204 would now look like this:

→ GI|RN204|GNShare1|G#1|

→ GI|RN204|GNShare2|G#2|

ORACLE has now signaled to the external system that two guests have checked into the same room - and has sent corresponding names.

It remains the external system's decision to see if:

multiple names can be supported,

or if just the first name should be used,

or if always the last received name is used.

Room Move:

Both guests moved to a different room:

← GO|RN204|G#1|

← GI|RN130|GNShare1|G#1|

← GO|RN204|G#2|

← GI|RN130|GNShare2|G#2|

NOTE:

the order of the above is not fixed and can look like this too:

← GO|RN204|G#1|

← GO|RN204|G#2|

← GI|RN130|GNShare1|G#1|

← GI|RN130|GNShare2|G#2|

Guest-based approach

(typically used by enhanced PBX or Video systems):

Sample link records:

→ LD|DA070705|TI091707|V#2.0.0|IFPB|

→ LR|RIGI|FLRNG#GNLGVGGGAGDGSSF|

→ LR|RIGO|FLRNG#GSSF|

→ LR|RIGC|FLRNG#GNLGVGGGAGDGSRO|

→ LR|RIRE|FLRNVMLRSCSDN|

→ LA|DA070705|TI091714|

A check-in of sharing reservation to room 332 would now look like this:

→ GI|RN332|G#35869|GNShare1|GLGE|GV0|GA090616|GD090617|GSN|

→ GI|RN332|G#35870|GNSharer2|GLGE|GV0|GA090616|GD090617|GSY|

For the first **GI** record, the share flag (**GS**) is "N," as at this point; there is no sharing situation in the room. For the second **GI**, the share-flag (**GS**) is set to "Y" as now more than one reservation is checked into this room (Room already Occupied).

Room Move of sharer reservation:

Reservation (**G#**) 35869 is moved from Room (**RO**) 332 to the room (**RN**) 312. The share-status (**GS**) of the new room is "N" and for the old room = "Y" to indicate the old room is still occupied.

→
GC|G#35869|GNShare1|GLGE|GV0|GA090616|GD090617|RN312|GSN|RO332|GSY|

Now the second reservation is moved to the same room:

→
GC|G#35870|GNSharer2|GLGE|GV0|GA090616|GD090617|RN312|GSY|RO332|GSN|

The shared flag (**GS**) is sent twice in the Guestdatachange-record (**GC**). This is necessary to signal the 'new' room (**RN**) and the 'old' room (**RO**). The Share-field (**GS**) is always sent right after the room-field to which it refers.

In the above example, the share-flag (**GS**) for the new room (**RN**) is set to "Y" as now both reservations are checked into room 312.

The share-flag (**GS**) for the old Room (**RO**) is set to "N" as this room is no longer occupied (if not, another guest is still checked in).

Virtual Numbers

NOTE:

Virtual Number (DID) functionality is not available with OPERA Cloud PMS version.

Virtual Number fields are used to assign DID, virtual, or phantom telephone extensions dynamically. Please note that 'Virtual Numbers' requires an additional module in the PMS.

(RA records are only available until Interface version 7. As of Interface version 8, DID handling has been added to the GI/GC/GO records.)

Virtual numbers can be seen as enhancing the guest data as described above. The functionality requires that Guest-based records are supported (= G# and GS are used in all records).

The number can be assigned Room-Based or Guest-Based in the PMS. Multiple Numbers can be assigned in the PMS; however, not more than one number per Pool. Depending on the PMS setup, the DID fields may include multiple values. The value will be sent separated by a semi-colon. Subsequently, no maximum field size can be defined for the DID fields.

Please note that all other records (e.g., ChargePosting, Wakeup, etc.) do not support DID numbers. It is expected that all those records use the physical extension.

As an extra feature, ORACLE PMS can attach a virtual number to a guest's profile to ensure that he always gets the same number whenever he returns to the Hotel. Subsequently, Assign or Un-Assign records may be sent without a relation to a room number.

Following are the descriptions of the necessary fields and records for DID handling:

 **NOTE:**

The tables below show the ADDITIONAL GI/GC/GO fields necessary to support DID functionality.

EN, ES, and EP can be filled with multiple values. In that case, the values are separated by semicolons. (see examples)

VN Guest in Notification

Record ID	Field ID	Description	Format	Direction
GI (Guest Check-In)	EN	Equipment Number	AN, no max. value	From PMS
	ES	Equipment Status	AN, no max. value (see ES-table)	From PMS
	EP	Pool-ID	AN, no max. value	From PMS

VN Guest Data Change Notification

Record ID	Field ID	Description	Format	Direction
GC (Room)	EN	Equipment Number	AN, no max. value	From PMS

Move)	ES	Equipment Status	AN, no max. value (see ES-table)	From PMS
	EP	Pool-ID	AN, no max. value	From PMS
	EO	Equipment Number of source room	AN, no max. value	From PMS
	ET	Equipment Status of source room	AN, no max. value (see ES-table)	From PMS
	EI	Pool-ID of source room	AN, no max. value	From PMS

VN Guest Out notification

Record ID	Field ID	Description	Format	Direction
GO (Guest Check-Out)	EN	Equipment Number	AN, no max. value	From PMS
	ES	Equipment Status	AN, no max. value (see ES-table)	From PMS
	EP	Pool-ID	AN, no max. value	From PMS

Virtual Number response notification

Record ID	Field ID	Description	Format	Direction
VA (Virtual Number notification)	EN	Equipment Number	AN, no max. value	To PMS
	AS	AnswerStatus	AN, 2 chars (see Answer Status table)	To PMS
	CT	Cleartext	ANS, max. 40	To PMS
	RN	RoomNumber	ANS, max. 8	To PMS
	ES	Equipment Status	AN, no max. value	To PMS
	EP	Pool-ID	AN, no max. value	To PMS
	G#	Reservation ID	N, max. 8	To PMS
	GP	Guest PIN	ANS, max 6	To PMS

 **NOTE:**

The **VA** record is OPTIONAL. It can signal a PIN for a specific DID back after an assignment. Or, e.g., to signal an error scenario.

Examples for a Room-based DID handling.

The idea of a Room-based virtual number handling is that a virtual number is attached to a room. If a second guest is checked into the same room, this guest will not get a new number, as there is already one assigned.

Check-in

(GI) for Room (RN) 11323, Reservation Number (G#) 35774, Guest Name (GN) Smith, Language (GL) English, Virtual Number (EN) 1062 from Pool-ID (EP) 1, and action is assigned (ES=1):

← GI|RN11323|G#35774|GNSmith|GFPaul|GLEA|EN1062|EP1|ES1|

Response from PBX:

Successful assignment of Virtual number (EN) 1062 to Room (RN) 11323, the assigned PIN (GP) is 4455

→ VA|EN1062|ASOK|CTASSIGNMENT EXECUTED|RN11323|ES1|EP1|
G#35774|GP4455|

Check-in (GI) for Room (RN) 244, Reservation Number (G#) 33611, Guest Name (GN) Borgward, Language (GL) german, Not-sharing reservation (GS), Guest-VIP (GV) status is 4, Virtual Number (EN) 1033 from Pool-ID (EP) 1 and action is assigned (ES=1), second: Virtual Number (EN) 2050 from Pool-ID (EP) 2 and action is set (ES=1)

← GI|RN244|G#33611|GNBorgward|GLGE|GSN|GV4|EN1033;2050|EP1;2|ES1;1|

RoomMove:

Mr. Smith is moved from Room (RO) 11323 to Room (RN) 11221. His virtual number stays the same and is moved (ES = 3) to the new room.

← GC|RN11221|G#35774|GNSmith|GFPaul|GLEA|RO11323|EN1062|EP1|ES3|
EO1062|E1|ET3|

Change of virtual number during guest's stay:

The existing virtual number 1032 is removed, and a new virtual number (1050) is assigned to room 372

← GC|G#34870|GNAsaro|GD090725|GLGE|GV4|EN1032;1050|EP1;1|ES0;1|RN372|
GSN|

NOTE:

There is no limit to virtual numbers which can be changed within one record. Typically this is limited by the number of available pools. If not, more than one number can be assigned per pool.

Check-Out:

← **GO|G#35774|RN11221|EN1062|EP1|ES0|**

Permanent Virtual number Assignment:

The Reservation-ID (**G#**) is sent as '0' because this guest does not currently have an active reservation.

← **GC|G#0|GNSmith|GFPaul|GLEA|EN1034|EP1|ES1|**

Check-In of a reservation, where the guest has a permanent assignment: Action is "keep" (**ES = 2**)

← **GI|RN11221|G#35774|GNSmith|GFPaul|GLEA|EN1034|EP1|ES2|**

Extended Guest Data

XL - Guest message text – online / also used to create messages in the ORACLE PMS system

XM - Guest message request

XT - Guest message text and other details

XD - Guest message delete/received

XR - Guest bill request

XI - Guest bill item

XB - Guest bill balance

XC - Remote Check-out request

These Record Types provide a mechanism to request and pass guest-specific information of a more comprehensive nature. They are designed for guest-oriented systems only. It is possible to send message text (**XL**) as an online process, that is, without requests, but as they occur in real-time.

Please note that most of these records require additional configuration in the PMS.

Text Messages:

There are different approaches to message handling. A system can notify a guest of the existence of a message, or it can display the message itself to the guest.

While handling messages, always keep in mind that different guests can be checked into a room, so managing messages must be done with the Reservation-ID (**G#**) and not in a room (**RN**).

Notification:

A notification is done by using the [RoomEquipment \(RE\) record](#). Using RE with the message-light field (**ML**) can be done room-based (only using **RN**) or guest-based (**RN + G#**)

It is required to use this feature guest-based if messages should be handled instead of switching on a light on a telephone set.

The ORACLE PMS will signal new messages by sending out the **RE** record. The external system can now alert the guest about the new message. Once the guest acknowledges the alert and requests to see the message, it can now be requested through the message request record (**XM**).

The status of the message in the ORACLE PMS system can now be changed to "retrieved." The external system can adjust this behavior by using [RequestType \(RT\)](#) in the **XM** record. Not sending **RT** or setting the value to '1' means: changing the status to "received." Setting **RT** to '0' signals to the ORACLE PMS system that the status should stay untouched so that this message can be requested again.

Consider possible delay in response from PMS for XM record requests. See related note - [Time outs waiting for Responses](#)

Used records: **RE**, **XM**, and **XT**, possibly **XD**

On-Line messages:

If online messages should be used, then different records must be defined: **XL** to retrieve messages right when they are entered and **XD** to signal back to the Fidelio system that a guest has now read a message.

When using On-Line messages, NOT use RoomEquipment (RE) with MessageLight (ML) is recommended. The existence of a message cannot be appropriately signaled by sending the On-Line-Message record; the message is already marked as "passed to the external system" in the PMS.

Message Delete

The XD record is used to signal that the guest has retrieved a message. This record works bi-directional as a guest can retrieve a message through various external systems or through the PMS itself.

NOTE:

Requesting a message may result in the actual message directly followed by an XD record from the PMS system, as by requesting the message, the external system has just signaled to the PMS system that a guest has retrieved the messages. Received messages can NOT be requested again.

Message Text Online

Record ID	Field ID	Description	Format	Direction
XL (Guest Message text - Online)	G#	Reservation Number	N, max. 10	Both
	MI	Message ID	N, max. 8	From PMS
	MT	Message Text	ANS, variable (max 2000)	Both
	RN	Room Number	ANS, max 8 (can be longer with Suite8 or OPERA-PMS)	Both
	\$J ¹	External Message ID	N, max 8	To PMS
	DA	Date	D	Both
	TI	Time	T	Both

1 - mandatory for the creation of messages

Message Request

Record ID	Field ID	Description	Format	Direction
XM (Guest Message Request)	G#	Reservation Number	N, max. 10	To PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	To PMS
	DA	Date	D	To PMS
	MI	Message ID	N, max. 8	To PMS

	RT	RequestType	N, max 4	To PMS
	TI	Time	T	To PMS

Message Text

Only with a prior Message Request!

Record ID	Field ID	Description	Format	Direction
XT (Guest Message Text)	G#	Reservation Number	N, max. 10	From PMS
	MI	Internal Message ID	N, max.8	From PMS
	MT	Message Text	ANS, variable (max. 2000)	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	DA	Date	D	From PMS
	TI	Time	T	From PMS

Message Delete

PMS can send it to the vendor or by a vendor to PMS.

Record ID	Field ID	Description	Format	Direction
XD (Guest Message 'delete,' used to signal that a message was retrieved)	G#	Reservation Number	N, max. 10	Both
	MI	Internal Message ID	N, max. 8	Both
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	Both
	DA	Date	D	Both
	TI	Time	T	Both

Some PMS systems may reply to an incoming XD a message with another XD record.

Guest Bill Request and Bill Item display:

Guest Bill request functionality offers a request for current bill items of a checked-in reservation and displays it on the external system (typically on TV or mobile devices). The external system sends a request command (**XR**) to the PMS containing the required Room Number (**RN**) and Reservation number (**G#**).

PMS will then send all bill items (XI) and Bill amount balance (XB) back to the external system.

In PMS, one can define which bill items shall be displayed to the guest - usually, there is only one billing window which the guest should see - for the bill items, the guest will pay himself (e.g., not the accommodation which his company pays).

PMS will respond with two message types:

the Bill Items (**XI**) contain details of each billposting. It also includes the Bill window number (**F#**) and whether the item shall be displayed to the guest (**FD**).

The Bill Balance (**XB**) contains the total balance amount of the reservation for all bill windows - also for those items which shall not be displayed.

Guest Bill Request:

Record ID	Field ID	Description	Format	Direction
XR (Guest bill request)	G#	Reservation Number	N, max. 10	To PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	To PMS
	DA	Date	D	To PMS
	TI	Time	T	To PMS

Guest Bill Item

Record ID	Field ID	Description	Format	Direction
XI (Guest bill item)	BD	Item Description	ANS, max. 300	From PMS
	BI	Item Amount	N, max. 20	From PMS
	DC	Department Code	N, max. 20	From PMS
	G#	Reservation Number	N, max. 10	From PMS

Record ID	Field ID	Description	Format	Direction
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	F#	Window/Folio Number	N, 2	From PMS
	FD	Item Display Flag	AN, 1 char (Y/N)	From PMS
	DA	Date	D	From PMS
	TI	Time	T	From PMS

Guest Bill Balance

Record ID	Field ID	Description	Format	Direction
XB (Guest bill balance)	BA	Balance Amount	N, max. 20	From PMS
	G#	Reservation Number	N, max. 10	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	AS	Answer Status	ANS, 2 chars (see Answer Status table)	From PMS
	DA	Date	D	From PMS
	TI	Time	T	From PMS

Remote Checkout Request:

A guest/reservation can use remote checkout functionality from the external system (typically TV or mobile devices) and check out himself instead of doing this at the reception desk.

Usage of Guest Bill view functionality is required to receive the proper Balance amount for the related Bill Window the guest can check out.

It also depends on PMS configuration and reservation settings if a Remote check out can be performed.

The external system sends a Remote checkout request command (**XC**) to PMS to initiate the checkout request.

PMS will respond with (**XC**) record including related Answer Status code (**AS**) - to tell if Checkout is allowed. Defining the (CT) field is recommended to see why the request was denied.

This response message does not indicate that the checkout is done, but PMS will start processing it.

The remote checkout is only completed when you receive the corresponding GuestOut command (**GO**) for this reservation.

You should add a related message to the guest that checkout is in progress for up to 2 minutes (PMS has an internal process running that will control the checkout then). In case of issues or checkout not completed, there will be no message sent back to your system – for this, the time out should be set.

Consider possible delays in response from PMS for **XR** and **XC** record requests. See related note - [Time outs waiting for Responses](#)

Record ID	Field ID	Description	Format	Direction
XC (Remote Check Out Request)	AS ¹	Answer Status	ANS, 2 chars (see Answer Status table)	From PMS
	BA ²	Balance Amount	N, max. 20	Both
	CT ¹	Clear Text	ANS, max. 40	From PMS
	G#	Reservation Number	N, max. 10	Both
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	Both
	DA	Date	D	Both
	TI	Time	T	Both

1 - sent from PMS to show the status of a request

2 - sent as part of Remote Check-out request

Examples

Guest Message Text - Online (XL) sent by PMS

Message # (**MI**) 903 sent online (**XL**, immediately after entry in PMS) for Reservation Number (**G#**) 12345 in Room (**RN**) 2781 entered in Front Office on 31 October 2000 (**DA**) at 12:47:53 PM (**TI**):

```
← XL|RN2781|G#12345|MI903|MTThis is a sample message.<CR><LF>It contains
formatting information<CR><LF> because it will be printed directly by<CR><LF>the other
system.<FF>|DA001031|TI124753|
```

Guest Message Request (XM) sent by an external system, PMS responds with (XT)

Request for the text of [all] guest messages (**XM**) for Room (**RN**) 2781, Reservation Number (**G#**) 12345:

```
→ XM|RN2781|G#12345|
```

Response to guest message request (**XT**) - the same message as shown in the **XL** record above:

← **XT|RN2781|G#12345|MI903|MT**This is a sample message.<CR><LF>It contains formatting information<CR><LF> because it will be printed directly by<CR><LF>the other system.<FF>**|DA001031|TI124753|**

Request for the text of [all] guest messages (XM) for Room (RN) 2781, Reservation Number (G#) 12345 with a negative response as no unread messages exist:

→ **XM|RN2781|G#12345|**

← **XT|RN2781|G#12345|DA001031|TI124753|**

(XT without MI/MT signals that no message exists for the inquiry information)

Guest Message delete/receive (XD) sent by PMS

When the Message is set to received or is deleted in the PMS, an XD the command is sent out to the external system

Request to change the status (XD) of Message # (MI) 903 for Reservation Number (G#) 12345 in Room (RN) 2781:

→ **XD|RN2781|G#12345|MI903|**

Guest Message delete/receive (XD) sent by external System

used even XL or XM & XT Message handling is used.

Request to change the status (XD) of Message # (MI) 903 for Reservation Number (G#) 12345 in Room (RN) 2781:

XD|RN2781|G#12345|MI903|

PMS will respond to this XD with another XD a record indicating the message being set to "received" in PMS

XD|RN2781|G#12345|MI903|

Create messages:

The XL record creates a new text message in the PMS system. To use XL for incoming messages, it is required to define the \$J (External Message-ID) field in the LR Record for XL.

This will indicate that incoming Messages shall be supported.

The message ID of the external system must be sent.

→ **XL|RN248|G#35850|MT**Please come to the restaurant, you forgot your glasses.**|\$J4711|**

A valid combination of RN and G# must be used to create a message.

Note: **XL** records with invalid criteria will be ignored. There will not be any notification of unsuccessful message creation.

The external system can use **XM** with **RT** set to '0' to check for existing messages.

 **NOTE:**

The external message ID (**\$J**) is not processed but only used for auditing by the PMS system; an internal message ID (**MI**) is assigned to each message. To delete (= mark as read) a specific message, the external system must inquire (**XM**) all messages to retrieve the necessary ID (**MI**).

Bill view:

Request to view bill (XR) for Reservation Number (GN) 12345 in Room (RN) 2781:

→ **XR|RN2781|G#12345|**

Response to bill request (**XI**), bill items (**BI**) for Reservation Number (**G#**) 12345 in Room (**RN**) 2781 with item information - PMS department code (**DC**), item amount (**BI**), item description (**BD**), date (**DA**) & time (**TI**) of posting, balance record (**XB**) has a folio total (**BA**) of 138.50:

← **XI|RN2781|F#1|G#12345|DC327|BI350|BDTelephone|DA001031|TI124753|FDY|**

← **XI|RN2781|F#1|G#12345|DC400|BI2500|BDLobby Bar|DA001031|TI1843000|FDY|**

← **XI|RN2781|F#2|G#12345|DC100|BI11000|BDRoom&Tax|DA001101|TI031000|FDN|**

← **XB|RN2781|G#12345|BA13850|DA001101|TI071500|**

 **NOTE:**

The balance (**XB**) **BA** fields reflect the total of the items sent. This may not be the same as the total of the entire guest folio as there may be items that the guest will not pay (i.e., postings covered by a travel agent), and that should not be displayed to the guest. These items are generally marked with **ItemDisplayFlag (FD)** "N." It is recommended that this value is not shown to the guest and that the displayed **BalanceAmount (BA)** is recalculated based on the total of all items (**FDY**) displayed to the guest

The recalculated **BA** needs to be stored temporarily if **XC** records are used, as the recalculated **BA** must be sent in the **XC** request.

Billview request with invalid request data - or bill view is not enabled:

→ **XR|G#23116|RN387|**

← **XB|ASUR|BA0|RN387|G#23116|**

Remote Check out:

Remote check-out request (**XC**) for Reservation Number (**G#**) 12345 in Room (**RN**) 2781, balance (**BA**) 138.50. Note that balance (**BA**) must be included in **XC** records (ORACLE PMS may check if the received balance matches the current folio total and may refuse the request in case these balances do not match. Value in **BA** should be taken from the **XB** record.):

→ **XC|RN2781|G#12345|BA13850|DA001101|TI071600|**

Response to remote check-out request (**XC**) for Room (**RN**) 2781, Reservation Number (**G#**) 12345 with positive answer status (**AS**) (check-out allowed and will be done as a background process):

← **XC|RN2781|G#12345|ASOK|CTVideo Checkout in Progress|DA001101|TI071602|**

 **NOTE:**

Please consider that PMS will respond with an XC record with an Answer Status code and Clear Text.

This response message does not indicate that the checkout is done, but PMS will start processing it.

The remote checkout is only completed when you receive the corresponding GuestOut command (**GO**) for this reservation.

You should add a related message to the guest that checkout is in progress for up to 2 minutes (PMS has an internal process running that will control the checkout then). In case of issues or check out not completed, there will be no message sent back to your system – for this, the time out should be set.

Locators

LO - Locator On

LF - Locator Off

LP - Locator Retrieve

Guest locators indicate where a guest is in the hotel if not in their room. A typical situation is when a guest waits for an important call or fax but goes to the restaurant for lunch. A locator set (**LO**) by the POS can inform the Front Desk or switchboard personnel where the guest can be found. However, if the functionality is required, any system may send or retrieve locators.

Please note that there can only be one active locator for a guest. This might lead to some problems if multiple systems are setting the locator, but in reality, the guest can only be in one place at a time.

Locator records must always include the Reservation Number (**G#**), as they are a guest, not room, a related feature. Suppose the locator record is sent from a system that does

not have the Reservation Number (G#). In that case, this can be retrieved by looking up the guest in question using a Posting Request (PR) record containing a Posting Info (PI) field (See [SPA & other Charge systems](#) section for details). This record is typically used by POSs but can be used by any system making a basic inquiry to get a list of guests and their room and guest numbers.

When turning a locator on (LO), the record must also include the current guest location sent as clear text (CT) and the time at which the locator should automatically expire (LT), i.e., for how long the locator is valid. When turning a locator off (LF), it is advisable that the external system first retrieves (LP) the current (if existing) locator for that guest to verify that it is not turning off a locator-set by another method. It is not necessary to turn off locators; in many cases, especially when dealing with locators of short duration, it is easier to let the locator expire on its own.

Consider possible delay in response from PMS for LP record requests. See related note - [Time outs waiting for Responses](#)

Locator On

Record ID	Field ID	Description	Format	Direction
LO (Locator On)	CT	Clear Text	ANS, max. 80	To PMS
	G#	Reservation Number	N, max. 10	To PMS
	LT	Locator expiry time	HHMM	To PMS
	TI	Time	T	To PMS
	DA	Date	D	To PMS
	RN	Room Number	ANS, max. 8	To PMS

Locator Off

Record ID	Field ID	Description	Format	Direction
LF (Locator Off)	G#	Reservation Number	N, max. 10	To PMS
	DA	Date	D	To PMS
	RN	Room Number	ANS, max. 8	To PMS
	TI	Time	T	To PMS

Locator Retrieve

Record ID	Field ID	Description	Format	Direction
LP (Locator retrieve)	AS ¹	Answer Status	ANS, 2 chars (see Answer Status table)	From PMS
	CT ¹	Clear Text	ANS, max. 96	From PMS
	G#	Reservation Number	N, max. 10	Both
	LT ¹	Locator Expiry Time	HHMM	From PMS
	DA	Date	D	Both
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	Both
	TI	Time	T	Both

1 – only required in response from PMS

Examples

Turn on a locator (**LO**) for Reservation Number (**G#**) 19683 from the Lobby Bar (**CT**), which expires (**LT**) at 14:30:

→ **LO|G#19683|CTLobby Bar|TI123000|LT1430|**

Turn off the locator (**LF**) set for Reservation Number (**G#**) 19683:

→ **LF|G#19683|**

Retrieve locator (**LP**) for Reservation Number (**G#**) 19683:

→ **LP|G#19683|**

Guest locator found with location (**CT**) and expiration time (**LT**):

← **LP|G#19683|ASOK|CTLobby Bar|LT1430|**

No guest locator was found for this guest (**AS, CT**):

← **LP|G#19683|ASN|CTNo Current Locator|**

Room Data

RE - Room equipment status

RE records are used to control the status of any room equipment (i.e., set/clear items such as

DND Do Not Disturb (DN) – set in PMS

Room Maid status, Guest Service Status (RS) – incoming & outgoing

**message waiting status (ML),
Class of Service (CS) for TMSs,
set/clear TV privileges for Video systems (TV),
Minibar Status,
Voicemail notification**

These records are generally room-oriented and need to be configured in the PMS. In some cases (i.e., **TV** and **MR**), it is possible to configure them in the Guest Data records (**GI**, **GC**). In that case, the rights may NOT be used in RE. (Always: Either in **GI/GC** or **RE**.)

The DND (Do Not Disturb) handling using the **DN** field informs the PBX to disable the telephone operators from transferring calls to the guest room. Instead, the external system might divert the call to the operator or to the internal voice box for that particular line.

All employees will set the DND status from the IFC submenu for hotels that use this function, including the telephone operators. **DN** field cannot be used to be sent to the PMS!

DND & Make-Up Room

If the external system wants to use a **DND & Makeup Room** status as known from the **“Makeup Room” & “Do Not Disturb” door sign**, then such status is to be sent by using the Room Status field (**RS**) in addition to the used Room Maid status values. In PMS, this is known as Guest Service Status and must be active!

There are only 3 Guest Service Status values:

1. DND ON
2. Make Up Room ON
3. All OFF

It is not expected that both DND & MUR will be ON simultaneously. If the value All OFF is sent, PMS will set the Guest Service Status to OFF, no matter which status was set to ON before.

NOTE:

Two or more statuses may be changed in the same record!
Some PMS systems cannot simultaneously support the Do-not-Disturb functionality and the TV rights.

Record ID	Field ID	Description	Format	Direction
RE (Room equipment status)	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	Both
	CS ¹	Class of Service	ANS, max. 1 see Class of Service table)	From PMS
	CT ²	Clear Text	ANS, max. 40	To PMS
	DN	Do-not-Disturb	AN, max. 1 (Y/N)	From PMS

G# ³	Reservation Number	N, max. 10	From PMS
ID	UserId	ANS, max. 16	To PMS
ML ³	Message Light Status	AN, 1 char (Y/N)	From PMS
MR ⁴	Minibar Rights	ANS, 2 chars (see Guest Rights table)	From PMS
PP ²	Printer Port	N, 1	To PMS
PU ⁶	Number of Persons	N, max. 2	To PMS
RS	Room Status	N, 2 (see Room Maid Status table)	To PMS
TV ⁵	TV Rights	ANS, 2 chars (see Guest Rights table)	From PMS
VM	Voice Mail	ANS, max. 4	To PMS

1 – required only if line COS (bar/unbar) functionality is available and used

2 – it can only be used together with PP / CT

3 – required only if Message Lamp functionality is available and used guest based (i.e., **G#** is used in **GI** records too) - typically used as notification for a text message; **only** use this if [Extended Guest Data messages](#) are supported.

4 – required only if Minibar functionality is available and used

5 – required only if TV Rights functionality is available and used

6 – can only be used together with RS

6 – currently only processed by Fidelio Suite8 and not by OPERA PMS

Examples

Message Notification

Turn Message Light (**ML**) on for Room (**RN**) 2781

← RE|RN2781|MLY|

Notify reservation (**G#**) 12345 of the existence of a message (see [Extended Guest Data](#))

← RE|RN2781|MLY|G#12345|

Do Not Disturb

Turn DND (**DN**) on for Room (**RN**) 2781:

← RE|RN2781|DNY|

Class Of Service

Set COS (**CS**) to '3' for Room (**RN**) 2781:

← RE|RN2781|CS3|

 **NOTE:**

Class of Service (**CS**) can either be changed through RoomEquipment (**RE**) or Check-In records (**GI/GC**). This depends on whether the vendor system can handle rights on a reservation or room level.

Set COS (**CS**) to '2' for Room (**RN**) 2781 and turn DND (**DN**) off :

← RE|RN2781|CS2|DNN|

Voice Mail Notification

Voice Mail (**VM**) notification for Room (**RN**) 2781:

→ RE|RN2781|VMY|

or

Voice Mail (**VM**) notification with unread (1)/Read (3) counts for Room (**RN**) 2781:

→ RE|RN2781|VM0103|

Room Maid Status

Maid status notification (**RS**) (clean/vacant) for Room (**RN**) 2781 (default maid statuses are listed in the [Room Maid Statuses Table](#) in Appendix B):

→ RE|RN2781|RS3|

Maid status notification (**RS**) (clean/vacant) for Room (**RN**) 2781, number of persons (**PU**) in the room is **3**:

→ RE|RN2781|RS3|PU3|

Maid status notification (**RS**) (clean/vacant) for Room (**RN**) 2781, number of persons (**PU**) in the room is **3**, the ID (**ID**) of the user changing the status is 'Maid5':

→ RE|RN2781|RS3|PU3|IDMaid5|

Maid status notification (**RS**) with text (**CT**) to print on the printer (**PP**) 1 for Room (**RN**) 2781:

→ RE|RN2781|RS1|PP1|CTSend maintenance personnel|

Room Status Guest Service Status

(Make up Room & Do Not Disturb):

Sending Do Not Disturb ON status (**RS**) for Room (**RN**) 2781:

RE|RN2781|RS8|

Sending Make Up Room ON status (**RS**) for Room (**RN**) 2781:

RE|RN2781|RS9|

Sending Guest Service Status OFF status (**RS**) for Room (**RN**) 2781:

RE|RN2781|RS10|

Clear Text Info

Text (**CT**) to be printed on the printer (**PP**) 0 for Room (**RN**) 2781:

→ RE|RN2781|PP0|CTGuest in 2781 needs assistance.|

NOTE:

The printer port (**PP**) and text (**CT**) can be used with **RE** records to print a message on a specified printer (must be configured); this only occurs if both fields exist in the record. This action will also be performed if other fields are included (i.e., set room status – **RS**).

NOTE:

Newer interface versions will expect the value '0' in **PP** and will do an internal definition of which printer to use. It is not recommended to use any other matter for **PP**.

Minibar Right

Set Minibar rights (**MR**) to normal vending (i.e., no alcoholic articles) for Room (**RN**) 2781:

← RE|RN2781|MRMN|

TV Right

Set Pay TV rights (**TV**) to block Adult movies in Room (**RN**) 2781:

← RE|RN2781|TVTX|

Notes: Pay TV rights to have the following precedence: **TN**, no rights (no TV channels); **TM**, all Pay channels blocked; **TX**, Adult Pay channels blocked; **TU**, all rights (includes all Pay channels). With TV rights, it is impossible to block normal Pay channels and allow Adult pay channels.

Freely definable rights can also be sent; these would be transferred as a numerical value from '4' to '9' and need to have their logic attached to the video system.

Building Management Systems

ORACLE Suite8 PMS system supports the option of sending RE records before the arrival of the guest to activate the in-room units. It is called **EMS Scheduler**.

NOTE:

OPERA PMS does not support EMS Scheduler handling.

These pre-arrival records can only be RoomEquipment (**RE**) records. The PMS would send an additional **GI** record once the guest arrives.

To support this functionality BM-systems need to support **RE** record with **CS**. **CS** values could e.g. be interpreted as:

'3' = Aircondition 100%

'2' = Aircondition 75%

'1' = Aircondition 50%

'0' = Aircondition in idle-mode

Send out intermediate ClassOfService **pre-arrival** – x mins before estimated arrival time

← RE|RN2781|CS2|

Send out intermediate ClassOfService **past-arrival time** (Guest not arrived x mins after estimated arrival time).

← RE|RN2781|CS1|

Outgoing Room Status

With Opera PMS, it is possible to get a notification of RoomStatus changes from the PMS. There is currently no plan to realize this functionality with Suite8 PMS.

 **NOTE:**

Suite8 PMS does not support outgoing Room Status handling.

 **NOTE:**

This functionality can **NOT** be combined with any other LinkRecord and requires a separate connection (and separate license on the PMS side):

Record ID	Field ID	Description	Format	Direction
RE (outgoing room equipment status)	RN	Room Number	AN, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	RS	Room Status	N, 2 (see Room Maid Status table)	From PMS

With this driver, only one record with the above two fields is possible:

Advise that the status for room 2781 (**RN**) has been set to status (**RS**):

RE|RN2781|RS3|

It is possible to request a [DataBase Synchronization](#) with normal routines.

ORACLE does not recommend using this functionality. It creates a case where the same data is stored both in the PMS and on the external system (i.e., redundant data). The external system cannot see if it went out of sync and can only verify this by requesting a DB-Sync.

Note: a Database-Synchronization may not be requested more than once per day, as this causes a lot of performance overhead on the PMS side.

Wakeup

WR - Wakeup request

WA - Wakeup answer

WC - Wakeup clear

Wakeup records allow both systems to set (**WR**) and clear (**WC**) wakeup calls. In addition, the external system must report the success or failure status of the call (**WA**) to the PMS after execution time. No response is necessary to a **WR** or **WC** record.

The PMS can be set to send wake-up requests in advance or right at wake-up time.

Wakeup-answer:

Just ONE result may be returned to the ORACLE PMS system for a specific wakeup. If the vendor system retries a wakeup, it may NOT send an intermediate result as it would be interpreted as the final result.

Wakeup Request

Record ID	Field ID	Description	Format	Direction
WR (Wakeup request)	DA	Date	D	Both
	RN	Room Number	ANS, max 8 (can be longer with Suite8 or OPERA-PMS)v	Both
	TI	Wake up Time	T	Both

Wakeup Clear

Record ID	Field ID	Description	Format	Direction
WC (Wakeup clear)	DA	Date	D	Both
	RN	Room Number	ANS, max 8 (can be longer with Suite8 or OPERA-PMS)	Both
	TI	Wake up Time	T	Both

Wakeup Answer

Record ID	Field ID	Description	Format	Direction
WA (Wakeup answer)	AS	Answer Status	ANS, 2 chars (See Answer Status table)	To PMS
	DA	Date	D	To PMS
	RN	Room Number	ANS, max. 8 v(can be longer with Suite8 or OPERA-PMS)	To PMS
	TI	Wake up Time	T	To PMS

Examples

Wakeup Request

from the PMS to set a wakeup request (**WR**) for Room (**RN**) 2781 at 7 AM (**TI**) on 31 October 2000 (**DA**):

← **WR|RN2781|DA001031|TI070000|**

Wakeup system Answer

Notifying PMS that the above wakeup call was unsuccessful (**AS**) because the telephone was busy, the value of **TI** is the wakeup time (NEVER the system time):

→ **WA|RN2781|DA001031|TI070000|ASBY|**

Wakeup Clear

Request from PMS to clear (**WC**) this wakeup call:

← **WC|RN2781|DA001031|TI070000|**

Request from wakeup system to clear (**WC**) all wakeup calls for this room (not recommended to be used):

→ **WC|RN2781|DA|TI|**

NOTE:

The structure of the **TI** field is HHMMSS. However, seconds **MUST** be sent as they were received in the WR regardless if wakeup can be handled by seconds or just by minutes. This mandatory behavior is necessary to properly link a wakeup result to the respective wakeup request in the PMS system.

Key Services

KR - Key request

KD - Key delete

KA - Key answer

KM - Key Data Change

KZ - Key Read

Overview

These are general-purpose keycard system records.

The PMS can use the Key Request (KR) record to make all possible requests to the Key Services system (KSS); different types of keys (i.e., new vs. duplicate keys) are specified by the fields sent in the record.

The Key Delete (KD) record is provided for those systems that prefer to get specific delete commands.

The Key Answer (KA) is supplied for completeness; the PMS may or may not pass responses from the KSS to the Front Office users. A key system must be able to support all three records.

Key Option / Access Rights

The specification currently supports multiple extra doors or areas that can be accessed with the guest key. These are sent in the **KO** field and are position-dependent, i.e., position 1 = Garage, 2 Minibar, etc. These are not hardcoded from the ORACLE viewpoint; they can vary from installation to installation.

Any blank position uses the defaults in the key card system; as ORACLE doesn't send trailing blanks if the field is shorter, any trailing positions should use default settings. Any position that contains a '0' is disabled. Any other character is significant only in the keycard system. If only a toggle is required, a '1' should be sent to enable this door/area. If a specific area has different access levels, specific characters are sent for the other levels. This method can handle rooms sometimes sold together as suites, sometimes sold as separate rooms.

It is also possible to support more than 20 Key Options. It is possible to send different values as per position.

Example

KO Pos 1 = 1 Garage

KO Pos 1 = 2 SPA

and so on

Key Request

Record ID	Field ID	Description	Format	Direction
KR (Key request)	KC	Key Coder	AN, max. 8	From PMS
	KT	Key Type	ANS, max. 2 (see Key Type table)	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	WS	Workstation ID	ANS, max. 16	From PMS
	\$1 ¹	Configurable Track 1	AN, max 40	From PMS
	\$2 ¹	Fidelio standard Track 2 format	AN, 16	From PMS
	A0 - A9 ^{2,3}	User Definable	ANS, variable	From PMS
	CT	Cleartext	ANS, variable (depends on usage and configuration)	From PMS
	DA	Date	D	From PMS
	DT ¹	Departure (Check-out) Time	HH:MM (as defined in PMS)	From PMS
	G# ^{1,5}	Reservation Number	N, max. 10	From PMS
	GA ¹	Guest Arrival Date	D	From PMS
	GD ¹	Guest Departure Date	D	From PMS
	GG	Guest Group Number	AN, max. 10	From PMS
	GN ⁶	Guest Name	ANS, max. 200	From PMS
	ID	User ID	ANS, max. 16	From PMS
	K# ⁴	Key Count	N, max. 2	
	KO ^{1,3}	Key Options	AN, max. 20	From PMS
	RT	RequestType	N, max 10	From PMS
	SI	SuiteInfo	ANS, max. 30 - values are separated by ';' (semicolon)	From PMS
TI	Time	T	From PMS	

1 – Not available with 'One Shot' Keys

2 – 'One Shot' Key only supports A0

3 – Requires special configuration in PMS

4 – Do NOT use for On-Line systems

5 - Mandatory for On-Line key systems

6 – Do NOT use when using CT field (e.g., to receive data to print on Key card)
OPERA only

Key Delete

Record ID	Field ID	Description	Format	Direction
KD (Key delete)	KC	Key Coder	AN, max. 8	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	WS	Workstation ID	ANS, max. 16	From PMS
	DA	Date	D	From PMS
	G# ¹	Reservation Number	N, max. 10	From PMS
	ID	User ID	ANS, max. 16	From PMS
	RT	RequestType	N, max. 10	From PMS
	SI	SuiteInfo	ANS, max. 30	From PMS
	TI	Time	T	From PMS

1 - Mandatory for On-Line key systems

KeyAnswer

Record ID	Field ID	Description	Format	Direction
KA (Key Answer)	AS	Answer Status	ANS, 2 chars (See Answer status table)	To PMS
	CT	Clear Text	ANS, max. 40	To PMS
	KC	Key Coder	AN, max. 8	To PMS
	WS	Workstation ID	ANS, max. 16	To PMS
	\$2 ²	Track 2 data	ANS, max 19	To PMS
	\$3	Track3 data	ANS, max 200	To PMS
	DA	Date	D	To PMS
	G# ¹	Reservation Number	N, max 10	To PMS
	TI	Time	T	To PMS

1 - Mandatory for On-Line key systems

2 - Only allowed if not used in **KR**

Key Data Change

KM records are used for On-Line key systems where key attributes can be changed without re-cutting the key. For example, this functionality is, e.g., used to conduct a room move or extend a card's validity. However, Guest Profile related data changes like Guest Last Name (GN) will not force sending out a KeyDataChange record. For On-Line systems, it is required that the key system tracks all cards by the PMS-Reservation-ID

(G#) and can address the data from a KM record to all cards made for a respective reservation. Subsequently, a key system must delete keys for a room based on one reservation-Id while leaving keys for the same room attached to a different reservation-Id intact.

Sharing reservations: Keys for sharers are treated like additional keys for an existing reservation, but the Reservation-ID (G#) would be different.

Record ID	Field ID	Description	Format	Direction
KM (Key Data Change)	G#	Reservation Number	N, max. 10	From PMS
	KC	Key Coder	AN, max. 8	From PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	RO ¹	Old Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
	WS	Workstation ID	ANS, max. 16	From PMS
	\$1 ²	Configurable Track 1	AN, max. 40	From PMS
	\$2 ²	Fidelio standard Track 2 format	AN, 16	From PMS
	CT	Cleartext	ANS, variable (depends on usage)	From PMS
	A0 - A9	User Definable	ANS, variable	From PMS
	DA	Date	D	From PMS
	DT	Departure (Check-out) Time	HH:MM (as defined in PMS)	From PMS
	GA	Guest Arrival Date	D	From PMS
	GD	Guest Departure Date	D	From PMS
	GG	Guest Group Number	AN, max. 10	From PMS
	GN ^{3,4}	Guest Name	ANS, max. 200	From PMS
	ID	User ID	ANS, max. 16	From PMS
	RT	RequestType	N, max 10	From PMS
SI	SuiteInfo	ANS, max. 30	From PMS	
TI	Time	T	From PMS	
UO	Suite Information of old room	ANS, max. 30	From PMS	

1 – Mandatory for Room move, not send for updated like validity changes.

2 – Not normally used, as the track data cannot usually be changed after the card has been encoded.

3 – Do NOT use when using CT field (e.g., to receive data to print on Key card).
OPERA only.

4 – Updating the Guest Name in PMS will not trigger a KeyDataChange message!

KeyRead

This functionality can display information about a key in the PMS.

Clear text (**CT**) in **KR** and **KM**: This field can be sent by some PMS systems with data printed on top of a key card (e.g., the guest's name, the validity of the key, etc.). The data itself is freely configurable. This should never contain the room number for security reasons if the key gets lost.

Note: In the following examples, references are made to sending commands to, or receiving commands from, the 'key coder.' However, this is for addressing and clarity's sake; there is only one physical connection between the ORACLE Interface PC and the KSS master PC.

Note: **K#** (KeyCount) will always be sent with value = '1'. If several keys should be made for a room, then several commands will be passed (with the respective KeyTypes - **KT**). This enables to identify a KA-records and link it to a request uniquely. PMS can display the Result-Code (**CT** value) to the employee. Optionally the vendor may omit the **K#** field and expect separate commands per key with different key types (**KT**). The first record would be for a new key, and all additional requests would be for duplicate keys.

HOWEVER, it is recommended not to define **K#** in the Link records.

Record ID	Field ID	Description	Format	Direction
KZ (KeyRead)	AS	Answer Status	ANS, 2 chars (See Answer status table)	To PMS
	CT	Clear Text	ANS, max. 40	To PMS
	KC	Key Coder	AN, max. 8	Both
	RN	RoomNumber	ANS, max. 8	To PMS
	WS	Workstation ID	ANS, max. 16	Both
	DA	Date	Date	Both
	DT	Departure Time	HH:MM	To PMS
	GN	Guest Name	ANS, max. 40	To PMS
	G#	Reservation ID	N, max. 10	To PMS
	GD	Departure Date	Date	To PMS
	KO	Key Options	ANS, max. 20	To PMS
	SI	SuiteInfo	ANS, max. 30	To PMS
TI	Time	Time	Both	

Examples

Key Request

New Key:

Key request (**KR**) from the workstation (**WS**) 3 for the key coder (**KC**) 1, 1 new key (**KT**) for Room (**RN**) 2781, (**KO**) area 1 enabled, areas 2 & 4 set to default, area 3 set to access level 2, area 5 enabled, areas 6-20 set to default, arrival date (**GA**) 29 December 1999, departure date (**GD**) 2 January 2000, Reservation Number (**G#**) 11122, Track 2 (**\$2**) should be encoded with the following string - 1000278100011122:

```
← KR|WS3|KC1|RN2781|KTN|KO1 2 1|GA991229|GD000102|G#11122|
$21000278100011122|
```

Additional/Duplicate Key:

Key request (**KR**) from the workstation (**WS**) 9 for the key coder (**KC**) 3, for a duplicate key (**KT**) for Room (**RN**) 2781, (**KO**) area 1 enabled, area 2 set to default, area 3 is disabled, area 4 set to access level 2, areas 5-20 set to default., arrival date (**GA**) 30 December 1999, departure date (**GD**) 5 January 2000, Reservation Number (**G#**) 12345, Track 2 (**\$2**) should be encoded with the following string - 1000278100012345, additional rooms (**SI**) 2788 and 2790 should be opened as well:

```
← KR|WS9|KC3|RN2781|KTD|KO1 02          |GA991230|GD000105|G#12345|
$21000278100012345|SI2788;2790|
```

NOTE:

The field list is the same for both key requests;
The content can be quite different (arrival/departure dates, optional areas, Track 2 information, etc.).
Each KSS decides how much information to maintain in its databases and how much information be duplicated from the original card to the duplicate.
The most important point is that 'New keys cancel any existing keys for the main room (both in databases and in the locks themselves) and that 'Duplicate keys do not. This is how the PMS treats the Keys already listed in its database.

Another important point is that the KSS should not attempt to interpret the data in Track 2 (**\$2**) as the contents of this data may be encoded and/or formats changed. The main purpose of such track encoding is to use the keys in a POS that supports EFT cards. Such POSs can then send the information to the PMS to interpret as needed; both the KSS and the POS should consider the track data transparent.

Key Answer

Response (**KA**) from key coder (**KC**) 3, answer status (**AS**) OK, Clear Text (**CT**) Key created, Reservation-Number (**G#**) 12345:

→ **KA|WS9|KC3|ASOK|CTKey created|G#12345|**

 **NOTE:**

It is necessary to specify both the PMS workstation and the Key Service system's coder in cases where more than one PMS workstation may be addressing one key coder.
ORACLE PMS will NOT send another Key-Request command automatically should a negative response be received as Answer status (**AS**). The user sees the response and decides if another try should be made.

Key Delete

Key delete (**KD**) from the workstation (**WS**) 9 for the key coder (**KC**) 3 for Room (**RN**) 2781, Reservation Number (**G#**) 12345:

← **KD|WS9|KC3|RN2781|G#12345|**

Response (**KA**) from key coder (**KC**) 3, answer status (**AS**) OK, ClearText (**CT**) Key deleted, Reservation Number (**G#**) 12345:

→ **KA|WS9|KC3|ASOK|CTKey deleted|G#12345|**

Request for a One-Shot key:

← **KR|WS3|KC1|RN2781|KTO|KO1 2 1|GA080312|GD080312|DT12:00|**

GD will typically be filled with the system date for a One-Shot Key.

Examples for KM records as used in On-Line systems:**Extension of a stay:**

Key data Change (**KM**) from the workstation (**WS**) 3 for the key coder (**KC**) 1, for Room (**RN**) 2781, arrival date (**GA**) 29 December 1999, departure date (**GD**) 4 January 2000, Reservation Number (**G#**) 11122,

← **KM|WS3|KC1|RN2781|GA991229|GD000104|G#11122|**

Response (**KA**) for request from Workstation (**WS**) 3 from key coder (**KC**) 1, answer status (**AS**) OK, ClearText (**CT**) Key updated, Reservation Number (**G#**) 11122:

→ **KA|WS3|KC1|ASOK|CTKey updated|G#11122|**

Move all keys to a reservation to a different room:

Key data Change (**KM**) from the workstation (**WS**) 3 for the key coder (**KC**) 1, for Room (**RN**) 3222, the old room was (**RO**) 2781, arrival date (**GA**) 29 December 1999, departure

date (**GD**) 4 January 2000, Reservation Number (**G#**) 11122, additional rooms (**SI**) to be opened are 3012 and 3013 :

← **KM|WS3|KC1|RN3222|RO2781|GA991229|GD000104|G#11122|SI3012;3013|**

 **NOTE:**

A system supporting "KM" MUST be able to separate keys by reservation number (**G#**). If several reservations are checked into a single room, only the respective keys for reservation number 11122 may be moved; the other keys must stay valid for the old room.

Key Read

Key Read (**KZ**) routine is requested at encoder number (**KC**) 01:

← **KZ|DA090401|TI125208|KC01|WSFO-PC1|**

Response from Key system after the key has been read. Optional data may be added to the key system's database response (e.g., the guest's name). The minimum returned information is the number of rooms that can be opened with this key.

→ **KZ|DA090401|TI125213|KC01|WSFO-PC1|G#12345678|RN11345|GNTest|GD090402|DT12:00|ASOK|CTKey Read OK|KO 1 1 23 1 |SI120,135|**

 **NOTE:**

The Key Read functionality is of pure informational purpose. None of the received information is stored or processed in the PMS, and it is only for display to the user.

A complete reservation-based scenario:

The first of two sharing reservations is checked into room (RN) 134:

← **KR|DA100317|TI152951|KC10|KO 1 1|KTN|IDVisor, Super|RN134|WSFO-PC1|G+35607|DT11:00|G#31717|GA100308|GD100310|GNVoigt, Thomas Herrn|\$21000013400031717|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

The second of two sharing reservations is checked into room (RN) 134:

← **KR|DA100317|TI152958|KC10|KO 1 1|KTD|IDVisor, Super|RN134|WSFO-PC1|G+35609|DT11:00|G#31719|GA100308|GD100310|GNVoigt, Kerstin Frau|\$21000013400031719|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

The first of two sharing reservations is moved from the room (RO) 134 to room (RN) 257:

← **KM|G#31719|KC10|RN257|RO134|WSFO-PC1|\$21000013400031719|DA100317|DT11:00|GA100308|GD100310|GNVoigt, Kerstin Frau|IDVisor, Super|TI153031|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

The second of two sharing reservations is moved from the room (RO) 134 to room (RN) 257:

← **KM|G#31717|KC10|RN257|RO134|WSFO-PC1|\$21000013400031717|DA100317|DT11:00|GA100308|GD100310|GNVoigt, Thomas Herm|IDVisor, Super|TI153044|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

The first of two sharing reservations is checked out of the room (RN) 257:

← **KD|DA100317|TI153127|KC10|IDVisor, Super|RN257|WSFO-PC1|G#31717|GNVoigt, Thomas Herm|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

The second of two sharing reservations is checked out of the room (RN) 257

← **KD|DA100317|TI153127|KC10|IDVisor, Super|RN257|WSFO-PC1|G#31719|GNVoigt, Kerstin Frau|**

→ **KA|KC10|WSFO-PC1|ASOK|CTKey created|DA100318|TI090411|**

Track2 Data

As described above, Track2-data can be used to identify POS systems. The PMS or the Key-system can generate the data, and the definition is done through the Link records.

\$2 may be defined in **KR** or **KA**.

Defining **KR** means that the PMS creates the unique ID and passes it to the Key-system for storage on Track2.

Defining **KA** means that the Key-System will generate a unique ID for the key (e.g., RFID UDID) and pass the ID back to the PMS in the Key Answer string.

Using **\$2** in **KA** will only be possible if the vendor system is prepared to receive multiple Key-Requests for the same room so that a unique **\$2** is returned for every key.

Track2 Data Examples:

New Key:

KR|FO-PC1|RN201|G#18901|GA080520|GD080521|GNTest|KC1|K#1|KTN|

KA|FO-PC1|KC1|\$201030FE159|ASOK|CTKey created|

The additional duplicate key for the same room (RN) and from the same Reservation-ID (G#):

KR|FO-PC1|RN201|G#18901|GA080520|GD080521|GNTest|KC1|K#1|KTD|
KA|FO-PC1|KC1|\$20104BD5D80|ASOK|CTKey created|

 **NOTE:**

The above example shows the \$2 value in HEX representation. This is the standard approach for Key-Systems, which use RFID cards and may, e.g., send the UID of the card as a \$2 value.

NEVER send the ID as a BINARY value. The number of characters would exceed storage capacities and would, in most cases, not be used for POS identification.

 **NOTE:**

Before implementing \$2 in KA, always ensure that the POS system in your target Hotels can read the \$2 value from the keys. Especially with RFID cards, this implies that the POS can interface with the respective reader, which may not always be the case.

SPA & Other Charge Systems

POS-Systems may not use the FIAS specification but the *Oracle HGBU-IFC8-XML POS Interface Specification* for an integration. Please contact the ORACLE Vendor Validation team for details.

PS - Posting (simple)

PR - Posting Request

PL - Posting List

PA - Posting Answer

The simple form of Posting records (**PS**) is for systems that do room postings without having to verify the guest (i.e., telephone, TV, etc.). These systems generally use the **GI/GO** records to ensure that the system in a specific room should be active. They also should use the Room Equipment status or Guest Data (Guest Rights) records to allow changing the equipment status after check-in (see examples in the sections above). In this case, it is also suggested to support a Class of Service (for example, a guest is checked in but cannot view Pay TV). This record does not support postings to specific guests (G#).

Another means of verifying guest privileges is by reading the information stored on a magnetic stripe card (i.e., normally Track 2 on the guest's room key); this is useful for Minibars, Vending machines, and other self-service POSs. When keys have been encoded with the standard Track 2 information, the POS can forward this track to identify guests and verify posting privileges.

 **NOTE:**

\$2 postings can only be done using Post Request (**PR**). It is mandatory to inquire to check if a valid card can be granted to a guest before any consumption.

 **NOTE:**

Never combine GI/GO with PR/PL handling. Guest-related postings must always be posted using the PR/PL combination. An inquiry is required before the posting.
The guest-related posting should not be posted based on data received from the GI/GO commands.

Posting Request records (**PR**) are intended to provide the functionality required by SPA systems and allow posting to PMS folios or accounts. The charges are generally guest-oriented and allow the user to make inquiries (**PI**) to the PMS to provide information such as room occupancy, guest hotel status or credit status, etc. The Posting Request record (**PR**) can be used to inquire and make the posting. If there is no Reservation Number (**G#**), or it is empty a valid **PI**-field and no TotalAmount (**TA**), then the request is treated as an inquiry. Else the request is treated as a posting. Postings using (**PR**) must have a preceding inquiry (**PR**).

Inquiries will only return a match on those guests who are currently checked in to the Hotel.

If a guest selection is needed, the PMS will return a Posting List record (**PL**); if multiple guest folios match the search criterion (i.e., sharers by room search), there will be multiple room number fields (**RN**), and multiple Reservation-ID fields (**G#**) returned. (Note: all blocks will begin with **RN** and **G#**; the order of any further fields may be different from installation to installation.)

PR records may **ONLY** be used for posting after a successful inquiry (**PR**). A posting with **PR** received without prior **PR** as inquiry (**PI**) will be ignored.

The Posting Answer record (**PA**) is required to ensure that the charges were posted properly and control the data flow. If specific fields are required to route a Posting Answer (**PA**) to a terminal or other posting location (**WS/SO**), these should be specified in the Field List (**FL**) for this Record Type during startup.

 **NOTE:**

Consider possible delays in response from PMS for **PS** and **PR** record requests. See related note - [Time outs waiting for Responses](#)

 **NOTE:**

Certain fields that may be defined in the Link Record (**LR**) for **PL** and/or **PA** will only contain data if sent in the **PS/PR** record by the other system, e.g., **P#**, **SO**, etc.

All amount fields (**TA**, and Subtotals **S1-S9**, Discounts **D1-D9**, and Taxes **T1-T9** and **TIP**) are expected without a decimal separator!

Taxes

Only use the tax fields in countries where taxes are calculated as "Add-On." In that case, the Subtotals should contain net amounts.

In other countries, subtotal fields should contain tax-inclusive amounts.

Before setting up taxes on the POS system, verify the respective ORACLE PMS installation settings. ORACLE PMS may expect to get net amounts in the subtotal fields and the Total amount field, i.e., WITHOUT any taxes, as they may be calculated within the PMS.

Linking Records

Mapping a response from the PMS to a request from the POS, it is recommended to implement **P#**. (For **PR/PL**, this is mandatory.) **P#** must contain a value between 1-99999999. This is a numeric field, so use it without leading zeroes or decimals, positive values only. The PMS will echo the sequence number from the request in the respective response record. This allows for the unique identification of responses in multi-thread scenarios.

If the POS server needs a method to identify the source POS workstation, **WS** is the correct approach. (For **PR/PL**, this is mandatory.) **WS** should contain the workstation-ID (or **COMPUTERNAME**) to allow unique identification. For **PR** posting, **WS** is a mandatory field.

Do not use the same **P#** for Inquiry and posting. The sequence number is used for record-based and not transaction-based identification.

Linking Postings

In case multiple records are sent for the same guest check, check number (**C#**), Date (**DA**), and Time (**TI**) **MUST** have the same value in all postings. Otherwise, it will be impossible for the PMS to link the postings together for the guest's folio imprint. = Cumulate by Check Number option in PMS.

Posting Simple

Record ID	Field ID	Description	Format	Direction
PS (Posting Simple)	DA	Date	D	To PMS
	DD ¹	Dialed Digits	N, max. 20	To PMS
	DU ¹	Duration	T	To PMS
	MA ²	Minibar Article	N, max. 4	To PMS
	M# ²	Number of Articles	N, max. 2	To PMS
	MP ³	Meter or Tax Pulse	N, max. 10	To PMS
	PT	Posting Type	AN, 1 char (see Posting Type table)	To PMS
	RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	To PMS
	SO ⁴	Sales Outlet	N, max. 5	To PMS
	TA ⁵	Total Posting Amount	M, max 15	To PMS
	TI	Time	T	To PMS
	+0 ⁶	Subtotal for Cash POS Rounding Differences	M,max.15	To PMS
	C#	Check Number	N, max. 8	To PMS
	CO	Credit Limit Override Flag	AN, 1 char, (Y/N)	To PMS
	CT	Clear Text	ANS, max. 20	To PMS
	CV	Covers	N, max. 5	To PMS
	D1 - D9	Discount 1-9	M, max. 15	To PMS
	ID	User ID	ANS, max. 16	To PMS
	P#	Posting Sequence Number	N, max. 8 (positive value)	To PMS
	PC	Posting Call Type	AN, 1 char	To PMS
	PM	Payment Method	ANS, max. 5	To PMS
	PX	Posting Route (i.e. Trunk ID)	N, max. 6	To PMS
	S1 - S9	Subtotal 1-9	M, max. 15	To PMS
	SC	Service Charge	M, max. 15	To PMS
	ST	Serving Time	N, max. 4	To PMS
	T#	Table Number	N, max. 4	To PMS
T1 - T9	Tax 1-9	M, max. 15	To PMS	
TP	Tip	M, max. 15	To PMS	
WS	Workstation ID	ANS, max. 16	To PMS	

Record ID	Field ID	Description	Format	Direction
	X1	Cross Reference Data - additional Posting information	ANS, max. 25	To PMS

- 1 - if the Posting Type is 'T' and charge costing is done by PMS using Duration (DU), Dialed Digits (DD) **must** be sent. (unformatted values ONLY, like: |DD004989920920| (i.e. no separators or spaces)
- 2 - required if Posting Type is Minibar Charge ('M')
- 3 - required if Posting Type is Telephone Charge ('T') and charge costing is done by PMS using meter pulses
- 4 - required if the same interface uses more than one Posting Type
- 5 - required if Posting Type is Direct Charge ('C')
- 6 - POS Cash Rounding Difference amount only if Subtotal 1-9 (S1-S9) is already used. POS Cash rounding difference amount (i.e., when occurring in Cash Postings with rounded Total Amount containing discounts)

Posting Inquiry/Posting Request

Record ID	Field ID	Description	Format	Direction
PR (Posting Request)	DA ¹	Date	D	To PMS
	G# ¹	Reservation Number	N, max. 10	To PMS
	GN ¹	Guest Name	ANS, max. 200	To PMS
	PI ²	Posting Inquiry	ANS, max. 10	To PMS
	PM	Payment Method	ANS, max. 5	To PMS
	RN ³	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	To PMS
	P# ⁴	Posting Sequence Number	N, max. 8 (positive value)	To PMS
	TA ¹	Total Posting Amount	M, max 15	To PMS
	TI ¹	Time	T	To PMS
	WS	Workstation ID	ANS, max. 16	To PMS
	\$2	Fidelio standard Track 2 format	N, max 19	To PMS
	+0	Subtotal for Cash POS Rounding Differences	M,max.15	To PMS
	C#	Check Number	N, max. 8	To PMS
CO	Credit limit Override Flag This flag defines if the PMS Creditlimit should be ignored for this posting. Normally only allowed for POS supervisors.	AN, 1 char, " <blank>" = don't override 'N' = don't override 'Y' = override	To PMS	
CT	Clear Text	ANS, max. 20	To PMS	

Record ID	Field ID	Description	Format	Direction
	CV	Covers	N, max. 5	To PMS
	D1 - D9	Discount 1-9	M, max. 15	To PMS
	DD	Dialed Digits	N, max. 20	To PMS
	DU	Duration	T	To PMS
	G+	Profile Number	N, max. 10	To PMS
	ID	User ID	ANS, max. 16	To PMS
	MA	Article Number	N, max. 4	To PMS
	M#	Number of Articles	N, max. 2	To PMS
	MX	Maximum Guests	N, max 1	To PMS
	PC	Posting Call Type	AN, 1 char	To PMS
	PT	Posting Type (except 'T')	AN, 1 char (see Posting Type table)	To PMS
	S1 - S9	Subtotal 1-9	M, max. 15	To PMS
	SC	Service Charge	M, max. 15	To PMS
	SO	Sales Outlet	N, max. 5	To PMS
	ST	Serving Time	N, max. 4	To PMS
	T#	Table Number	N, max. 4	To PMS
	T1 - T9	Tax 1-9	M, max. 15	To PMS
	TP	Tip	M, max. 15	To PMS
	X1	Cross Reference Data - additional posting information	ANS, max. 25	To PMS

- 1 - required only after guest selection
- 2 - required only for inquiries with no guest selection
- 3 - mandatory in postings, but not in inquiries
- 4 – Posting Sequence Number shall be unique as per Message sent and shall not be the value from **Posting Inquiry** in corresponding **PostRequest** message.

Posting List

Record ID	Field ID	Description	Format	Direction
PL (Posting List)	G# ¹	Reservation Number	N, max. 10	From PMS
	GN ¹	Guest Name	ANS, max. 200	From PMS
	P# ³	Posting Sequence Number	N, max. 8 (positive value)	From PMS

RN ¹	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
WS	Workstation ID	ANS, max. 16	From PMS
A0 - A9 ²	User Definable	ANS, variable	From PMS
BA	Balance Amount	N, max. 20	From PMS
C#	Check Number	N, max. 8	From PMS
CL ²	Credit Limit	N, max. 15	From PMS
DA	Date	D	From PMS
G+	Profile Number	N, max. 10	From PMS
GA	Guest Arrival Date	D	From PMS
GD	Guest Departure Date	D	From PMS
GF	Guest First Name	ANS, max. 80	From PMS
GG	Guest Group Number	AN, max. 10	From PMS
GL	Guest Language	ANS, max 10 (see Guest Language table)	From PMS
GT	Guest Title	ANS, max. 20	From PMS
GV	Guest VIP Status	AN, max. 20	From PMS
ID	User ID	ANS, max. 16	From PMS
NP	No-Post Status	Y/N	From PMS
PM	PMS Payment Method	ANS, max. 5	From PMS
SO	Sales Outlet	N, max. 5	From PMS
TI	Time	T	From PMS

- 1** - required if the account(s) matching search information in **PI** are found
2 – requires configuration in PMS
3 – The posting Sequence Number will be valued from the corresponding **Posting Inquiry**

Posting Answer

Record ID	Field ID	Description	Format	Direction
PA (Posting answer)	AS	Answer Status	ANS, 2 chars (see Answer Status table)	From PMS
	CT ¹	Clear Text	ANS, max 50 ⁴	From PMS
	DA	Date	D	From PMS
	P# ³	Posting Sequence Number	N, max. 8 (positive value)	From PMS

RN	Room Number	ANS, max. 8 (can be longer with Suite8 or OPERA-PMS)	From PMS
TI	Time	T	From PMS
WS ³	Workstation ID	ANS, max. 16	From PMS
C#	Check Number	N, max. 8	From PMS
G# ²	Reservation Number	N, max. 10	From PMS
GN ²	Guest Name	ANS, max. 200	From PMS
ID	User ID	ANS, max. 16	From PMS
SO	Sales Outlet	N, max. 5	From PMS

1 - required only if the search fails (**PR** only)

2 - not available when **PS** is used

3 - required if posting is done with **PR** - will be value from corresponding **Posting Inquiry, Simple Posting, or Posting Request**

4 - on OPERA PMS, the length can be much longer than 50 characters, with no final limitation.

Examples Posting Simple

Posting (simple)/Answer

Telephone charge posting (PTC, i.e. call costed by other system) to Room (**RN**) 2781, cost (**TA**) 10.50, on 15 September 2000 (**DA**) at 12:35:45 (**TI**), sequence number (**P#**) 0729, dialed digits (**DD**) 004989920920, international call (**PC/CT**):

→ **PS|RN2781|TA1050|DA000915|TI123545|P#1729|DD004989920920|PC|CTInternational|PTC|**

Posting accepted (ASOK):

← **PA|RN2781|ASOK|P#1729|DA000915|TI123545|**

NOTE:

Only one **PS** posting may be sent at a time. The sending system must wait for **PA** before sending the next posting.

Telephone posting (PTT, i.e. call costed by PMS by pulse count) to Room (**RN**) 2781, 8 meter pulses (**MP**), on 15 September 2000 (**DA**) at 12:40:41 (**TI**), sequence number (**P#**) 0730, dialed digits (**DD**) 2123830, local call (**PC/CT**):

→ **PS|RN2781|PTT|MP8|DA000915|TI124041|P#1730|DD2123830|PCL|CTLocal|**

Posting accepted (ASOK):

← **PA|RN2781|ASOK|P#1730|DA000915|TI124041|**

Telephone posting (PTT, i.e. call to be costed by PMS by duration and dialed digits) to Room (**RN**) 2781, duration (**DU**) 3 minutes, 45 seconds, on 15 September 2000 (**DA**) at 12:42:54 (**TI**), sequence number (**P#**) 0731, dialed digits (**DD**) 5106850320, national call (**PC/CT**):

→ **PS|RN2781|PTT|DU000345|DA000915|TI124254|P#1731|DD5106850320|PCN|CTNational|**

Posting accepted (ASOK):

← **PA|RN2781|ASOK|P#1731|DA000915|TI124254|**

 **NOTE:**

For Telephone charge postings, the PMS will be configured to use only one posting method, i.e., pre-costed call (**PT** field set to C) or costing by a pulse (**MP**) or duration/dialed digits (**DU/DD**, **PT** field set to T).

If the costing is done by duration (**DU**), dialed digits (**DD**) must be provided. Date (**DA**) and time (**TI**) reflect the start of the call.

Posting Sequence (**P#**) in all cases should be incremented after every successful transmission.

Minibar posting (Direct Charge, PTC) to Room (**RN**) 2781, Sales Outlet (**SO**) 100 (this charge comes from a system that also sends laundry charges), cost (**TA**) 14.50, on 15 September 2000 (**DA**) at 12:42:54 (**TI**), sequence number (**P#**) 0732:

→ **PS|RN2781|PTC|SO100|TA1450|DA000915|TI124254|P#1732|**

Posting accepted (ASOK):

← **PA|RN2781|ASOK|P#1732|DA000915|TI124254|**

Even though this is a Minibar posting, it uses Posting Type (**PT**) set to C because the charge amount (**TA**) is sent.

Minibar posting (PTM) to Room (**RN**) 2781, guest consumption: article (**MA**) 1450 2 (**M#**) times on 15 September 2000 (**DA**) at 12:42:54 (**TI**), sequence number (**P#**) 0733:

→ **PS|RN2781|PTM|MA1450|M#2|DA000915|TI124254|P#1733|**

Posting accepted (ASOK):

← **PA|RN2781|ASOK|P#1733|DA000915|TI124254|**

Posting Type (**PT**) is sent as M to indicate that the PMS should calculate the charges itself based on article number (**MA**)/articles consumed (**M#**); this will be done even if a pre-calculated charge is sent.

Suppose **MA** is sent but no **M#**; the article count defaults to 1. The value in **M#** must be >0. Negative counts can *not* be accepted.

X1 Supplemental information - additional posting information in PS or PR record:

Instead of **CT** or in addition to the **CT** field, **X1** can be included in the posting record (**PS** or **PR**) to send additional postings information.

 **NOTE:**

Additional configuration in PMS Configuration is required to get the X1 field to the PMS posting details.

→ **PS|RN2781|PTC|TA1480|X12 Bottles of Apple Juice|DA000915|TI124254|P#1737|**
PS|RN2781|PTC|TA1480|CTMinibar|X12 Bottles of Apple Juice|DA000915|
TI124254|P#1737|

Should the **PS** record consist of several articles, and should these be sent to separate department codes in the PMS, will the PMS copy the **X1** information to all separated postings in full.

PS|RN2781|PTM|MA4000|M#1|MA4011|M#2|X1 Minibar |DA000915|
TI124300|P#1740|

 **NOTE:**

A split of the information contained in X1 is not possible.
Also, it is not possible to send separate CT or X1 fields as per the sent article in one PS record.
The first CT and/or X1 value will be sent to PMS.

PS|RN2781|PTM|MA4000|M#1|X1 Minibar Soft Drinks|MA4011|M#2|X1 Minibar Wine
|DA000915|TI124330|P#1741|

Examples Posting Inquiry / Posting List

Posting Request (Inquiries)/List/Answer

Posting Request from POS Sales Outlet (**SO**) 123, Terminal (**WS**) 456, User **ID** Eli, for Room (**PI**) 2781 using PayMethod (**PM**) ROOM:

→ **PR|SO123|WS456|IDELI|PI2781|DA000915|TI124254|P#1734|PMROOM|**

List of guests (PL) in Room (RN) 2781:

```
← PL|SO123|WS456|IDELI|RN2781|G#12345|GNGuest, Mr.|RN2781|G#12381|
GNSharer, Mr.|P#1734|
```

As seen in the example above, if guests matching the **PI** search criterion are found, the list is formatted as Room Number (**RN**)/Reservation Number (**G#**)/Guest Name (**GN**) triplets (these can occur multiple times if there are sharers in a room, but all three fields are sent for each guest). If the search data was ASCII (i.e., search by guest name), the Room Number/Reservation Number/Guest Name fields could also occur more than once:

```
<Guest List> := <Room List>[<Room List>][<Room List>]
```

```
<Room List> := RN<data>|G#<data>|GN<data>|
```

For A/R or City Ledger charges, inquiries are still required. However, since these accounts are not checked into rooms, the Room Number (**RN**) field will be filled with the A/R account number. **G#** will be sent with the value '0' as no respective guest number is available. The file cannot be omitted, as it is mandatory for **PR** records. It then takes the following form:

```
<Room List> := RN<data>|G#0|GN<data>|
```

Posting Request from POS Sales Outlet (**SO**) 123, Terminal (**WS**) 456, User **ID** Josh, for posting information (**PI**) 5781:

```
→ PR|SO123|WS456|IDJOSH|PI5781|DA000915|TI124254|P#1735|PMROOM|
```

Invalid room response (AS/CT):

```
← PA|SO123|WS456|IDJOSH|ASNG|CTINVALID ROOM|P#1753|
```

Posting request from POS Sales Outlet (**SO**) 123, alpha search (**PI**) for 'G' with "Maximum number of matching guests returned" (**MX**) set to 4:

Note: **MX** defines the number of matches returned in the PL record.

```
→ PR|SO123|WS456|IDELI|PIG|MX4|DA000915|TI124254|P#1736|PMROOM|
```

List of guests (PL), Room (**RN**) 2781 – Gast (**GN**), Room (**RN**) 352 – Gandhi and Garibaldi (**GN**, see room list description above):

```
← PL|SO123|WS456|IDELI|RN2781|G#12345|GNGast, Hr.|RN352|G#12940|GNGandhi,
Mr.|RN352|G#12875|GNGaribaldi, Mr.|P#1736|
```

Examples Posting Request

Posting Request (Charges)/Answer

Posting request from POS for Room (**RN**) 2781 with Reservation Number (**G#**) 12875 selected, Sales Outlet (**SO**) 123, total (**TA**) to post 105.75, F&B (**S1**) charges 80.00, tax (**T1**) 25.75, check number (**C#**) 1234, 2 covers (**CV**), serving time (**ST**) 4:

→ PR|SO123|WS456|IDJOSH|RN2781|G#12875|GNGaribaldi,
Mr.|TA10575|S18000|T12575|C#1234|CV2|ST4|DA000915|TI124254|P#1737|PMROOM
|

Posting accepted (ASOK):

← PA|SO123|WS456|IDJOSH|RN2781|G#12875|GNGaribaldi, Mr.|ASOK|
DA000915|TI124254|P#1737|

Total Amount calculation

NOTE:

In all cases, the sum calculated by adding all subtotal, tax, and discount fields (which means the amount in a discount field should be negative) must equal the Total Amount (TA) field (see check splitting example below).

The interface calculates as follows:

TA = S1 + [S2] + [S3] + T1 + [T2] + [T3] + D1 + [D2] + [D3] + [TP] + [SC]

Example:

|TA1000|S1800|D1-200|T1400|
=> 10.00 = 8.00 - 2.00 + 4.00

Posting request from POS for Room (RN) 2781 with Reservation Number (G#) 12345 selected, Sales Outlet (SO) number 123, total (TA) to post 228.50, food charges (S1) 80.00, beverage charges (S2) 60.00, miscellaneous (S3) 40.00, tax food (T1) 25.75, tax beverage (T2) 15.25, tax miscellaneous (T3) 10.50, discount food (D1) 10.00, Tip (TP) 5.00, Service Charge (SC) 2.00 check number (C#) 1234, serving time (ST) 4:

→ PR|SO123|WS456|IDELI|RN2781|G#12345|GNHr. Gast, Hr.|TA22850|S18000|
S26000|S34000|T12575|T21525|T31050|D1-1000|TP500|SC200|C#1234|ST4|
DA000915|TI124254|P#1738|PMROOM|

Posting accepted (ASOK):

← PA|SO123|WS456|IDELI|RN2781|G#12345|GNHr. Gast Hr.|ASOK|DA000915|
TI124254|P#1738|

It is unnecessary to send a subtotal, tax, or discount field if the value is 0. In the above example, even though there could be corresponding discounts for beverage (S2/D2) and miscellaneous (S3/D3), they are not sent because there was no discount given.

If the other system is a POS that supports splitting checks between guests or payment methods, the individual subtotals, taxes, and discounts should also be split to equal the Total Amount to be posted when added together. The same system handles all rounding corrections, and the revenue totals between the POS and the PMS will match.

For a split check, where only 110.75 should be posted, these items should be recalculated as follows:

→ PR|SO123|WS456|IDELI|RN2781|G#12381|GNSharer, Mr.|TA11075|S14000|S23000|S32000|T11287|T2763|T3525|C#1234|D1-500|ST4|DA000915|TI124254|P#1740|PMROOM|

The following example is **wrong** because the **subtotals, taxes, and discounts reflect the totals for the whole check and not the current total amount (TA):**

→ PR|SO123|WS456|IDJOSH|RN2781|G#12381|GNSharer, Mr.|TA11075|S18000|S26000|S34000|T12575|T21525|T31050|C#1234|D1-1000|ST4|DA000915|TI124254|P#1741|PMROOM|

Posting request from POS for payment method (PM) AMEX selected, Sales Outlet number 123, total (TA) to post 105.75, F&B (S1) charges 80.00, tax (T1) 25.75, check number (C#) 1234, serving time (ST) 4:

→ PR|SO123|WS456|IDJOSH|PMAMEX|TA10575|S18000|T12575|C#1234|ST4|DA000915|TI124254|P#1742|

Posting accepted (ASOK):

← PA|SO123|WS456|IDJOSH|ASOK|DA000915|TI124254|P#1742|

NOTE:

Inquiries for payment methods configured to post directly to one specific account (i.e., normally anything other than a room or A/R charges), for example, Cash or EFT charges, are neither required nor supported. These postings are accepted (ASOK), or the Answer Status field (AS) is accompanied by a Clear Text field (CT) with a failure message.

In addition, if payment methods are enabled for non-room charges, the Payment Method (PM) field should also be sent with room charges, e.g., PMROOM.

Inquiries Using The Magnetic Stripe (Track2) of a Guest's Key-Card

Some POS systems can read and pass information from Track 2 of magnetic key cards. With these systems, the track should be read and passed as is to the PMS (the data on Track 2 up to the end sentinel for the card number should be transparent to both the Key Service System and the POS).

→ PR|SO123|WS456|IDELI|PMROOM|\$24200278100012345|C#1234|DA000915|TI124254|P#1788|

← PL|SO123|WS456|IDELI|RN2781|G#12345|GNGuest, Hr.|P#1788|

The posting will follow the same rules as described above.

\$2 can only be used in the inquiry circle to identify a guest. The posting needs to be done using the returned room number and Reservation-ID for this guest.

POS Cash Rounding Difference Amount Handling

Some European countries like Italy, Sweden, Belgium, and Denmark enforce Cash Rounding on a legal basis, while it continues to be a market requirement for the Netherlands, Finland, and Switzerland.

Some Postings containing rounded total amounts and discount amounts might cause different amounts.

Please note that this is not around any room charges nor any non-cash payments to some PM; typically, none of such payments are getting rounded, nor will you see all cash payments being rounded. Typically prices use rounded values, and as such, this is only seen when, i.e., discounts are applied.

The Rounding Difference amount can be sent to the PMS so the sum of all subtotals, Discounts, Service Charges, Tax amounts, Tip amounts will still match the total amount. It is then up to the PMS to handle this Rounding Difference amount.

The Difference amount must always be sent in a defined Subtotal field (**S1-S9**) or the **+0** field.

+0 = POS Cash Rounding Difference amount (i.e., when occurring in Postings with rounded Total Amount containing discounts) when Subtotal1-9 (**S1 to S9**) already used for transfer of main article groups

```
→PR|P#0113|SO1|WS1|PTC|PM1|CV2|TA4380|S1100|S2200|S3300|S4400|S5500|  
S6600|S7700|S8800|S9900|D1-123|+03|C#123456|X1CheckNo#001123456|DA200428|  
TI101700|
```

POS Cash Rounding Difference amount can also be sent in S1-9 fields when not all 9 Subtotal fields are used.

```
→ PR|SO123|WS456|IDELI|TA890|S1520|S2500|D1132|S32|C#1234|DA200427|  
TI124200|P#1738|PM1|
```

7

FIPS – Fidelio Interface Protocol Specification

Physical Transmission Layer

Serial connections (RS232)

Connection Types

Communication is handled using an asynchronous data format. The transmission line characteristics are defined in CCITT V.24 and RS-232. The number of data bits may be seven (7) or eight (8); however, eight is necessary to transmit multiple character sets. The number of stop bits is set to one (1). Oracle suggests using a parity bit; however, parity should be configurable to be Odd, Even, or None. The baud rate for the transmission should be user-definable, with a recommended range of 1200 baud to 38400 baud. A peak rate of at least 9600 baud should be supported.

Hardware Handshake

The 'hardware handshake' lines can be employed to detect whether the other system is properly connected. However, Oracle raises the DTR and RTS lines only to indicate that the interface program is running; DTR and RTS do not fluctuate during a program operation, nor are they used for byte-level flow control. If the hotel user performs an orderly exit from the interface software, the signal is dropped. DSR and CTS, as received from the vendor, are recognized in the same fashion; if the signals are high, Oracle will assume the vendor is available for communication. Oracle assumes that the vendor has entered maintenance or another non-communicating state if the signal drops. Do not use the handshake lines for intra-record flow control.

NOTE:

Signal levels can remain high when systems experience operational difficulties; signal monitoring is recommended as a secondary connection integrity check.

Cabling

The Oracle Property Management System Interface hardware uses the following pinning of serial port connectors, which can be either DB-9 or DB-25; DB-9 is the standard. Oracle operates as a DTE (data terminal equipment). Connectors on the interface PC are typically male; the vendor cable connector is female. Note: Only pins which are used are shown.

DB-9 Connector

PIN #	NAME	DESCRIPTION
case		Shielding ground
1	DCD	Data Carrier Detect
2	RX	Receive
3	TX	Transmit
4	DTR	Data Terminal Ready
5	GND	Signal ground
6	DSR	Data Set Ready
7	RTS	Request to Send
8	CTS	Clear to Send

DB-25 Connector

PIN #	NAME	DESCRIPTION
1		Shielding ground
2	TXD	Transmit
3	RXD	Receive
4	RTS	Request to send
5	CTS	Clear to send
6	DSR	Data set ready
7	GND	Signal ground
8	DCD	Data carrier detect
20	DTR	Data terminal ready

General Transmission Layer Considerations

Most of the information in this section only applies to asynchronous serial connections. However, Oracle recommends that records are started and ended with link control bytes, even when using TCP/IP. Framing the data makes it easier to avoid record concatenation, as it is possible to find more than one record in the TCP/IP buffer when calling received().

Data Bytes Format

Records are composed of data bytes and link control bytes. The data portion should not contain bytes normally reserved for link control (Hex 00 through Hex 1B and Hex 7F). Some systems use the control characters from Hex 1C through Hex 1F as field separators; other control characters are sometimes used as data (such as CR - Hex 0D and HT - Hex 0A) to indicate display formatting. When this is the case, these characters are considered part of the normal data stream and do not require a preceding escape character; they are also not available for use as link control characters. For most transmissions, the rest of the standard ASCII character set is sufficient (Hex 20 through Hex 7E); however, to support multiple alphabets, extended ASCII (Hex 80 - Hex FF) may be used. Data is passed in an unpacked format; it should not be packed in nibblized, BCD, or other formats.

Link Control Bytes Format

Link control bytes indicate the beginning and end of records, positive and negative responses, and provide other functionality such as flow control. Control bytes fall into the range from Hex 00 through Hex 1B (though Hex 7F is also a control byte). Many have standard definitions, though interface protocol specifications determine some usages. The implementation and use of specific control bytes are described below.

Other Notes

Link level responses should always be sent within the timeouts detailed later. However, the sender always should be prepared to receive an <ACK>/<NAK> as soon as it transmits the LRC (or <ETX> if no transmission error checking is being done).

Some means of record level checking (as opposed to byte-level parity checking) for transmission errors are highly desirable; for the half-duplex protocol, it is required. This specification supports using a Longitudinal Redundancy Character (or Vertical Parity). The standard LRC is calculated using a seed value of 0 and bitwise XORing each byte following the <STX>, including the <ETX>. This value is transmitted as a single ASCII character immediately following the <ETX> as the final character of the

record. When the LRC that the receiver calculates the one sent by the sender's transmitted, the receiver replies with an <ACK> and processes the record; otherwise, the receiver responds with a <NAK> and discards the last received record. The LRC may be any of the 256 characters in the ASCII character set, so receiving routines should be able to distinguish by the LRC's position in a record that it is not a link control byte if the LRC is an <STX>, <ETX>, and so on.

LRC Calculation Example

The following example demonstrates an LRC calculation (the data is '12345'):

bit	7 6 5 4 3 2 1 0	
<STX>	0 0 0 0 0 0 1 0	unused
LRC	0 0 0 0 0 0 0 0	seed
Byte 1	0 0 1 1 0 0 0 1	'1'
LRC	0 0 1 1 0 0 0 1	
Byte 2	0 0 1 1 0 0 1 0	'2'
LRC	0 0 0 0 0 0 1 1	
Byte3	0 0 1 1 0 0 1 1	'3'
LRC	0 0 1 1 0 0 0 0	
Byte4	0 0 1 1 0 1 0 0	'4'
LRC	0 0 0 0 0 1 0 0	
Byte5	0 0 1 1 0 1 0 1	'5'
LRC	0 0 1 1 0 0 0 1	
<ETX>	0 0 0 0 0 0 1 1	
LRC	0 0 1 1 0 0 1 0	'2' = transmitted LRC

<STX>12345<ETX>2

Full-Duplex Protocol

Overview

Full duplex communications mean both systems can simultaneously transmit link or application-level messages. This does not mean implementing a half-duplex protocol on full-duplex hardware (see half-duplex specification). In addition, the baud rate used in a specification only applies to individual bits in a byte; all bits, including start, stop, and parity (if used), will be transmitted at the specified baud rate. However, asynchronous communications mean that the start bit of the next byte might not be transmitted within the next interval. Because of this, timeouts are implemented for transmission of a record; these, of course, depend on the maximum record length in an interface and the standard baud rate. There are also timeouts for link-level responses.

Though both systems may be transmitting application-level messages simultaneously, it is most likely they will not finish transmission at the same time. Link level messages (either responses or flow control) should not be intermingled with application messages. For example, do not put an <ACK>/<NAK> response in the middle of the record being sent. If the other system finishes its transmission first, finish the local transmission completely and then transmit any line control sequences.

Basic Full-Duplex Description

There is no line bid (for full-duplex, the <ENQ> is a response status inquiry after a response timeout, see below). Unless one system has sent an application-level request to stop the link or lowered the hardware handshake lines (used only when the connection is idle), the link is always considered active.

The beginning of a record is marked by an <STX>. The data in the record is then sent in ASCII format, and the record terminates with an <ETX>. A general rule of thumb for calculating the transmitted record timeout is to divide the baud rate by 10 (1 start bit, 8 data bits, & 1 stop bit) to get many characters per second, divide the result by the length of the longest record, then multiply the second result by 2. This is about twice as long as it should take to transmit a record with no inter-byte pauses.

Oracle strongly suggests using an LRC (see description above). The LRC is transmitted immediately following the <ETX>. Whether an LRC is used or not, when the receiving system determines it has gotten a complete record, it sends a link-level response within 3 seconds (this is a maximum timeout; hopefully, the response will be much quicker). If the receiving system is transmitting a data record, the beginning of the 3 second timeout period starts immediately after the end of the transmission. If no LRC is used, the response is always an <ACK>; if an LRC is used, the response (<ACK>/<NAK>) is based on the result of the LRC calculation.

If the sender receives an <ACK>, the link is now available for transmitting another record if more records are sent. If the sender gets a <NAK>, the original record is transmitted again, retrying 3 times (for 4 transmission attempts) as necessary. Suppose a successful transmission cannot be completed (including <ACK>/<NAK> timeouts). In that case, the record is discarded, the error logged (implementation of error logging is up to the individual system), and link activity proceeds as if a <ACK> had been received.

It is possible that the sender does not receive an <ACK> or <NAK> within the 3 second data record response timeout (timer 1). If this occurs or the sending system receives any character, it should respond with a <ENQ> within 3 seconds (timer 2). This is a prompt to the receiving system to retransmit its last <ACK> or <NAK> (do NOT retransmit the record), using the same 3 second timeout (timer 1) as for the initial response. Any <ENQ>s received (other than as an LRC) after the 3 second timeout (timer 2) are invalid (respond with a <NAK>), (though this can trigger the retransmission of a record, this violates the timeouts specified in this protocol). Any character received other than an <ENQ> also resets the last response to a <NAK>. This prevents double postings in systems where charge data is transmitted.

Full Duplex Link Control Bytes

<STX> (Hex 02) - <STX> is used to mark the beginning of a record. The data portion of the record immediately follows it.

<ETX> (Hex 03) - <ETX> is used to mark the end of the data portion; however, it is not part of the data.

<ENQ> (Hex 05) - <ENQ> is used to inquire for logical layer response when an incorrect (not <ACK> or <NAK>) or no response was received. After transmitting a record, the sender should receive a response within 3 seconds. If no response is received by the end of the timeout, or an unexpected character (not <ACK>, <NAK> or <STX>) is received, the sender should transmit an <ENQ>. This process is retried up to 3 times as necessary. If an <ACK>/<NAK> response is never received, the record is discarded, and the sender logs an error.

<ACK> (Hex 06) - <ACK> represents positive acknowledgment of receipt of a valid transmission.

NOTE:

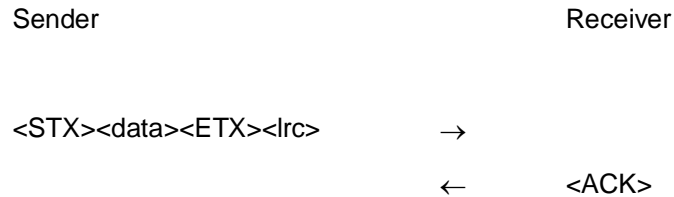
This is used for link-level validation only; if a higher level validation is required (such as validating record types or other record data), the receiving system should return a high-level data record with the data acceptance/rejection information. However, a <ACK> response to the original record is always required first

<NAK> (Hex 15) - <NAK> is a request from the receiving system to have the sender retransmit the last data record because the receiver detected a transmission error. This is

link level only (see the note above in <ACK> description). Records that are <NAK>' ed should not be processed by the receiver at any higher level.

Examples

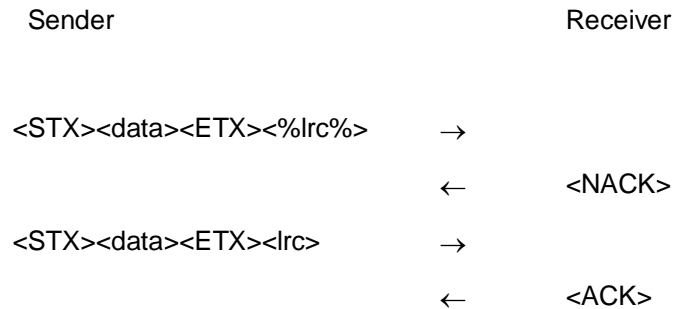
Normal transmission:



 **NOTE:**

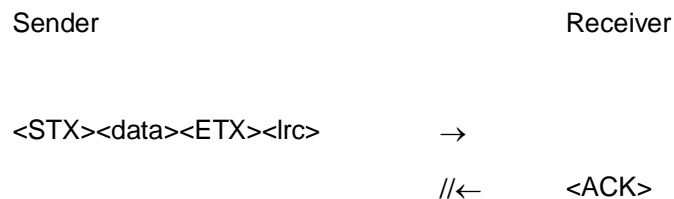
The receiver should respond with a <ACK> within 3 seconds for normal transmission.

Message received with incorrect LRC:



Note: If the receiver does not receive a valid LRC, it should respond with a <NAK> within 3 seconds. The sender then retransmits the original message. If the message cannot be successfully transmitted after three retries (a total of 4 transmission attempts), the message is discarded, and both systems should log a transmission error.

Low-level response not recognized or not received within timeout:



timeout
or
← <??>
<ENQ> →
← <previous response>

 **NOTE:**

If the sender receives an invalid response (not <ACK> or <NAK>) or does not receive any response within the 3 second timeout, it should send a <ENQ> within 3 seconds. The receiver then retransmits the initial response.

TCP/IP

Overview

TCP/IP is available as a transport protocol layer from Version 7 of the ORACLE Interface program. As TCP/IP is a robust protocol that supports its handshaking, transmission integrity checking, etc., it is not the intention of this document to redescribe how TCP/IP works. However, there are some options that ORACLE has chosen to implement in a specific manner; for example, ORACLE uses stream type TCP/IP, not datagram. These implementation details are listed below.

Framing

Many applications that use TCP/IP send unframed data (i.e., they don't use link control bytes). However, since the low-level handshaking is performed by the TCP/IP stack, a second data block may be sent before the receiving application has actually retrieved the first data block, even in a stream implementation. This results in a situation where the receiver has to be able to parse the data to ensure that trailing data blocks are not accidentally discarded.

One approach is to implement a handshake at the application level; however, this can slow the throughput, defeating one of the main gains of using TCP/IP. The other method is to add the start and end of data link control bytes as a mechanism to mark a record. Even if the application receives more than a single data block, it can easily parse out the first complete block and process it, storing any extra characters until it can locate

another complete block. ORACLE, therefore, requires the use of link control bytes even with TCP/IP.

Client/Server

The interface program always acts as the TCP/IP connection server. A port or service number is configured at installation; this is the port the client should try to connect. The interface program listens indefinitely on that port for a connection request. Upon receiving a request, the listening thread accepts it and starts a communication thread for the data exchange on the defined port. The communication thread reads any data received and checks at least once every 10 milliseconds whether there is something to be sent.

When the client's connection is closed, the interface keeps on listening for a new connection request.

Therefore, other systems that want to communicate using TCP/IP should implement a client that establishes one connection and uses this for all communications.

NOTE:

Only one client at a time can have a connection for a given port number; thus, the ORACLE interface communicates to a single application ONLY. (Multiport connections are only possible with references to Interface Version 8 but depend on the specific usage. Please contact your local ORACLE office about availability.)

After a successful connection, the interface will keep listening on the given port. Should another connect request be received, will the interface drop the original connection and connect to the client which connected last.

This behavior allows a vendor to reestablish a broken link without restarting the ORACLE Interface.

However, as restarting the link may take several seconds, it is mandatory to keep an established connection open.

Connection Mode

NOTE:

The communication is designed to be contained within a local area network infrastructure and not for transmission through active components over the internet.

A persistent connection to the ORACLE interface is always required.

The connection should be left up and running until one of the systems needs to shut down for maintenance purposes.

TCP/IP SSL Stream

Overview

In addition to the above TCP/IP connection, it is possible to establish a secured TCP/IP connection between the Interface application (as of Version 8.14.5.0) and the vendor system using Server/Client certificates.

At the TCP/IP connection, a server certificate (optional additional client certificate) will be validated between both parties.

Mutual authentication is required as a security level, ensuring both parties will authenticate each other.

The certificates should be defined as *.pfx and *.crt files.

Certificates can be generated by Certification Authority (CA) or can be self-signed certificates.

As they will work as point-to-point connection validation only valid for these two parties, it is not necessary to import them to the Microsoft Cert store; the Interface application supports even it.

TCP/IP SSL Stream uses the same handling (Framing, Client/Server, Connection Mode) as the described TCP/IP connection above.

A sample TCP SSL Client project can be provided, which will explain the alternative validation of self-signed certificates in file format via the "RemoteCertificateValidationCallback" function without import in Microsoft's cert store. Contact the Oracle validation team for more details.

8

Usage of LS LD LA LE Records

Overview

Please note that these are only examples showing how LA and LE records are used; the contents of other records and data in the fields therein may be incomplete or may not represent valid data.

Additional comments are marked with 'green text.'

General Considerations

Oracle recommends that a system using FIAS does not send a record right at startup but first waits for incoming records for up to 3 seconds.

If the Oracle interface is running while the external system start, it will receive a record from the Oracle interface in this time frame. In TCP connections, this will be an **LS** record. In RS232 connections, it will more likely be an **ENQ** character.

The external system should initiate the communication if no record is received within 3 seconds.

Communication can be initiated by sending **LS** or **LD**.

Alive-Check

The LS record can be used to check the activity of the link to the Oracle interface. (Might be useful on TCP connections through active devices like switches etc.) Oracle Recommends NOT to send an LS as live check more than once every 5 minutes.

[07.28/14:31:01]#0/0 -> LS|DA160728|TI143101|

[07.28/14:31:01]#0/0 <- LA|DA160728|TI143101|

NOTE:

The Oracle Interface will NOT respond with an LA to an LA while in Alive status.

TCP-IP Implementations

Implementations using TCP/IP can be considered less complicated than serial connections.

1) LS Link Start records

a) IFC starts, then the external system

```
[07.28/14:31:01]#1/0 <- LS|DA160728|TI143101|
```

Upon detection of TCP connect, we always start with LS and expect an LD record as a response.

When connecting to our TCP-Port, the vendor system receives this LS and can process it accordingly, sending LD records.

```
[07.28/14:31:01]#0/0 -> LD|DA160728|TI143059|V#1.01|IFWW|
```

Now we are waiting for the LR records and the LA record which signals the end of the Link Record sequence

```
[07.28/14:31:01]#0/1 -> LR|RIPR...
```

```
[07.28/14:31:01]#0/2 -> LR|RIPL...
```

```
[07.28/14:31:01]#0/3 -> LR|RI...
```

```
[07.28/14:31:01]#0/4 -> LA|DA160728|TI143059|
```

We will reply with an LA record indicating initialization is finished; both sides are now in sync.

```
[07.28/14:31:01] <MessLvl3> ChangeLinkState:Alive
```

```
[07.28/14:31:01]#1/0 <- LA|DA160728|TI143101|
```

b) IFC starts, while external system still running

```
[07.28/14:31:01]#1/0 <- LS|DA160728|TI143101|
```

Upon detection of TCP connect, we always start with LS and expect an LD record as a response. When connecting to our TCP-Port, the vendor system receives this LS and can process it accordingly, sending LD records.

```
[07.28/14:31:01]#0/0 -> LD|DA160728|TI143059|V#1.01|IFWW|
```

Now we are waiting for the LR records and the LA record, which signals the end of the Link Record sequence.

```
[07.28/14:31:01]#0/1 -> LR|RIPR...
```

```
[07.28/14:31:01]#0/2 -> LR|RIPL...
```

```
[07.28/14:31:01]#0/3 -> LR|RIPA...
```

```
[07.28/14:31:01]#0/4 -> LA|DA160728|TI143059|
```

We will reply with an LA record indicating Init is finished; both sides are now in sync.

[07.28/14:31:01] <MessLvI3> ChangeLinkState:Alive

[07.28/14:31:01]#1/0 <- LA|DA160728|TI143101|

c) External system starts, then IFC

As IFC is not running, no TCP connection can be made. The external system should regularly try to reconnect to see if IFC has been restarted. Once IFC can be reached, the sequence is the same as listed above under a):

[07.28/14:31:01]#1/0 <- LS|DA160728|TI143101|

Upon detection of TCP connect, we always start with LS and expect an LD record as a response. When connecting to our TCP-Port, the vendor system receives this LS and can process it accordingly, sending LD records.

[07.28/14:31:01]#0/0 -> LD|DA160728|TI143059|V#1.01|IFWW|

Now we are waiting for the LR records and the LA record, which signals the end of the Link Record sequence.

[07.28/14:31:01]#0/1 -> LR|RIPR...

[07.28/14:31:01]#0/2 -> LR|RIPL...

[07.28/14:31:01]#0/3 -> LR|RIPA...

[07.28/14:31:01]#0/4 -> LA|DA160728|TI143059|

We will reply with an LA record indicating Init is finished; both sides are now in sync.

[07.28/14:31:01] <MessLvI3> ChangeLinkState:Alive

[07.28/14:31:01]#1/0 <- LA|DA160728|TI143101|

d) External System starts while IFC is still running

IFC will drop the TCP-Port upon receiving a LE and re-open it, waiting for a connection request. Subsequently, the startup sequence will be the same as listed above under a).

[07.28/14:31:01]#1/0 <- LS|DA160728|TI143101|

Upon detection of TCP connect, we always start with LS and await an LD record as a response. When connecting to our TCP-Port, the vendor system receives this LS and can process it accordingly, sending LD records.

[07.28/14:41:01]#0/0 -> LD|DA160728|TI144059|V#1.01|IFWW|

Now we are waiting for the LR records and the LA record, which signals the end of the Link Record sequence.

[07.28/14:41:01]#0/1 -> LR|RIPR...

[07.28/14:41:01]#0/2 -> LR|RIPL...

[07.28/14:41:01]#0/3 -> LR|RIPA...

[07.28/14:41:01]#0/4 -> LA|DA160728|TI144159|

We will reply with an LA record indicating Init is finished; both sides are now in sync.

[07.28/14:31:01] <MessLvl3> ChangeLinkState:Alive

[07.28/14:31:01]#1/0 <- LA|DA160728|TI144101|

4) LE (Link End) records

Each system should wait for the LE from the other system before dropping the port. IFCs timeout for waiting for this response is 2 seconds.

a) External System shuts down while IFC still running, then IFC shuts down

[07.28/15:16:40]#0/0 -> LE|DA160728|TI151638|

[07.28/15:16:40]#1/1 <- LE|DA160728|TI151640|

[07.28/15:16:40] <MessLvl3> ChangeLinkState:End

[07.28/15:16:40] <MessLvl3> ChangeCommState:Off

[07.28/15:16:46] <MessLvl3> Close MonClass -->done

[07.28/15:16:46] <MessLvl2> ShutDown -----

b) IFC shuts down

[07.28/14:44:00]#1/0 <- LE|DA160728|TI144400|

[07.28/14:44:00]#0/0 -> LE|DA160728|TI144358|

The TCP port is now dropped.

[07.28/14:44:00] <MessLvl3> ChangeLinkState:End

[07.28/14:44:00] <MessLvl3> ChangeCommState:Off

[07.28/14:44:00] <MessLvl3> Close MonClass -->done

[07.28/14:44:00] <MessLvl2> ShutDown -----

RS232 communication

1) LS (Link Start) records

a) IFC starts, then External System.

[11.17/10:29:27]#1/0 <- LS|DA161117|TI102926|w

We always start with LS and expect an ACK in response.

[11.17/10:29:30]#1/0 <- ♣

ENQ (looking for a response to LS). Note: ENQ represents 'transmitter did not receive any response from the receiver regarding the last record,' i.e., our LS.

[11.17/10:29:33]#1/0 <- ♣

[11.17/10:29:36]#1/0 <- ♣

IFC resends LS record after 3 ENQs are sent without response.

[11.17/10:29:39]#1/0 <- LS|DA161117|TI102926|w

[11.17/10:29:42]#1/0 <- ♣

[11.17/10:29:45]#1/0 <- ♣

[11.17/10:29:48]#1/0 <- ♣

[11.17/10:29:51]#1/0 <- LS|DA161117|TI102926|w

[11.17/10:29:54]#1/0 <- ♣

[11.17/10:29:57]#1/0 <- ♣

[11.17/10:30:00]#1/0 <- ♣

The vendor sends ACK as it receives an ENQ - now might send LS.

[11.17/10:30:00]#1/0 -> ♠

[11.17/10:30:20]#0/0 -> LS|DA161117|TI103116|f

[11.17/10:30:20]#0/1 <- ♠

IFC resends LS records as it receives LS records from an external system.

[11.17/10:30:20]#1/1 <- LS|DA161117|TI103020|y

[11.17/10:30:20]#1/1 -> ♠

[11.17/10:30:20]#0/0 -> LD|DA161117|TI103116|V#1.0|IFWW|?

[11.17/10:30:20]#0/1 <- ♠

[11.17/10:30:21]#0/0 -> LR|RIGI...

[11.17/10:30:21]#1/1 <- ♠

[11.17/10:30:21]#0/0 -> LR|RIGO..

[11.17/10:30:21]#1/1 <- ♠

[11.17/10:30:21]#0/0 -> LR|RIGC..

[11.17/10:30:21]#1/1 <- ♠

.
. .
. .

[11.17/10:30:22]#0/0 -> LA|DA161117|TI103118|a

[11.17/10:30:22]#1/1 <- ♠

We will reply with an LA record indicating Init completed; both sides are now in sync.

[11.17/10:30:24]#1/0 <- LA|DA161117|TI103024|o

[11.17/10:30:24]#1/0 -> ♠

b) IFC starts, while External System was still running

```
[11.17/10:21:07]#1/0 <- LS|DA161117|TI102107||  
[11.17/10:21:07]#1/0 -> ♠  
[11.17/10:21:07]#0/0 -> LD|DA161117|TI102203|V#1.0|IFWW|9  
[11.17/10:21:07]#0/1 <- ♠  
[11.17/10:21:08]#0/0 -> LR|RIGI|FLRNGNGVG#GLGGSSFA0A1A2A3A4A5A6A7A8A9|  
[11.17/10:21:08]#0/1 <- ♠  
[11.17/10:21:09]#0/0 -> LA|DA161117|TI102204|n  
[11.17/10:21:09]#0/1 <- ♠  
[11.17/10:21:11]#1/0 <- LA|DA161117|TI102111|j  
[11.17/10:21:11]#1/0 -> ♠
```

c) External System starts, then IFC

(We do not see their LS so that we may receive an ENQ).

```
[11.17/10:21:06]#1/0 -> ♣  
[11.17/10:21:06]#0/1 <- ♠  
[11.17/10:21:07]#1/0 <- LS|DA161117|TI102107||  
[11.17/10:21:07]#1/0 -> ♠  
[11.17/10:21:07]#0/0 -> LD|DA161117|TI102203|V#1.0|IFWW|9  
[11.17/10:21:07]#0/1 <- ♠  
[11.17/10:21:08]#0/0 -> LR|RIGI|FLRNGNGVG#GLGGSSFA0A1A2A3A4A5A6A7A8A9|  
[11.17/10:21:08]#0/1 <- ♠  
[11.17/10:21:09]#0/0 -> LA|DA161117|TI102204|n  
[11.17/10:21:09]#0/1 <- ♠  
[11.17/10:21:11]#1/0 <- LA|DA161117|TI102111|j  
[11.17/10:21:11]#1/0 -> ♠
```

d) Both systems running - External System sends Alive check

(Link was previously Alive)

```
[11.17/11:21:07]#1/0 -> LS|DA161117|TI112107|j  
[11.17/11:21:07]#1/0 <- ♠  
[11.17/10:21:11]#1/0 <- LA|DA161117|TI112111|j  
[11.17/11:21:11]#1/0 -> ♠
```

2) LE (Link End) records

a) External System shuts down while IFC is still running, then IFC shuts down

[11.17/10:25:36]#0/0 -> LE|DA161117|TI102631|h

[11.17/10:25:36]#0/1 <- ♠

[11.17/10:25:36]#1/1 <- LE|DA161117|TI102536|i

[11.17/10:25:36]#1/1 -> ♠

[11.17/10:25:36] <MessLvl3> ChangeLinkState:End

[11.17/10:25:36] <MessLvl3> ChangeCommState:Off

[11.17/10:25:40] <MessLvl3> ChangeCommState:Sync

[11.17/10:25:40] <MessLvl3> ChangeLinkState:CommOn

IFC is now in 'LE' state will await for a LS and re-initialization with LD, LR, LA.

[11.17/10:25:40]#1/0 <- LS|DA161117|TI102540|{

[11.17/10:25:40]#1/0 -> ♠

[11.17/10:25:40]#0/0 -> LD|DA161117|TI102636|V#1.0|IFWW|;

[11.17/10:25:40]#0/1 <- ♠

[11.17/10:25:40]#0/0 -> LR|RIGI..

[11.17/10:25:40]#0/1 <- ♠

[11.17/10:25:40]#0/0 -> LR|RIGO..

[11.17/10:25:40]#0/1 <- ♠

[11.17/10:25:40]#0/0 -> LR|RIGC..

[11.17/10:25:40]#0/1 <- ♠

[11.17/10:25:42]#0/0 -> LA|DA161117|TI102637|j

[11.17/10:25:42]#0/1 <- ♠

[11.17/10:25:42] <MessLvl3> ChangeLinkState:Start

[11.17/10:25:43] <MessLvl3> ChangeLinkState:Alive

[11.17/10:25:43]#1/0 <- LA|DA161117|TI102543|j

[11.17/10:25:44]#1/0 -> ♠

b) IFC shuts down while External System is still running

[11.17/10:45:25]#1/0 <- LE|DA161117|TI104524|i

[11.17/10:45:25]#1/0 -> ♠

External system to reply with LE.

[11.17/10:45:25]#0/0 -> LE|DA161117|TI104620|n

[11.17/10:45:25]#0/1 <- ♠

```
[11.17/10:45:25] <MessLvl3> ChangeLinkState:End  
[11.17/10:45:25] <MessLvl3> ChangeCommState:Off  
[11.17/10:45:25] <MessLvl3> Close MonClass -->done  
[11.17/10:45:25] <MessLvl2> ShutDown -----
```

c) IFC shuts down while External System not running

(External System sent no LE before shutdown).

[11.17/10:28:41]#1/0 <- LE|DA161117|TI102841|a

IFC will wait 3 seconds, awaiting a reply, then shuts down.

```
[11.17/10:28:44] <MessLvl3> ChangeLinkState:End  
[11.17/10:28:44] <MessLvl3> ChangeCommState:Off  
[11.17/10:28:45] <MessLvl3> Close MonClass -->done  
[11.17/10:28:45] <MessLvl2> ShutDown -----
```

9

Appendix A - FAQ

This section contains answers to frequently asked questions.

Frequently Asked Questions:

Do I have to send the link startup sequence (LD/LR)?

We strongly recommend sending the link startup sequence if you receive a Link Start (LS) record from Oracle Interface. You will receive only default records with default formats if it is not sent. In newer versions of the Oracle Interface program, you will not receive any records anymore, and incoming records will be ignored. There are very few situations where the defaults are useful, as they are quite limited, not defined in the specification, and may change at any time. There may be a point where no default record formats are supported.

Which records should I describe in the link startup sequence?

It is best to send a Link Record (LR) for all records that you wish to use, not just the ones that you will receive but also the ones that you will send (not currently required, though helpful for installation and maintenance, and may be needed in future versions). The only records you don't need to describe are the Link records themselves (LS/LD/LR/LA/LE) and the Database records (DR/DS/DE); these records have fixed formats and cannot be changed.

What do I include in the Link Record (LR) as Field List (FL) if a record has multiple uses?

Include all fields in the FL that you will use, regardless of which direction the record is sent. For example, the Room Equipment (RE) record can be used both to control Message Lamps (ML) and Do Not Disturb (DN) and to report Room Status (RS) from the external system. The same applies to Guest Data change (GC); it can be used for Guest Info/Name change and Room Moves. Only send one LR for such records.

Do I have to send the LD/LR/LA sequence every time at startup?

No. This is dependent on what you receive as a response to your Link Start (LS). If you receive an LS, the Oracle interface has been restarted while your software was stopped; you must re-describe your record formats. If you receive a Link Alive (LA) when you send an LS, Oracle Interface still recognizes your interface. You may resend the LD and LR records if you wish to change your configuration or just send the LA to finish opening the link.

Shall I answer Link Alive (LA) records with an LA record?

Only if you did not send an LS or LA. This is, in most cases, sent by Oracle in response to one of these two records having been sent by the other system

What should I do if I receive an <ENQ> at startup?

This means that Oracle has sent a record, usually a Link Start (LS) or Link End (LE). If you are using the full-duplex low-level protocol, respond with a <NAK> to indicate that you have not received a valid record (Half-Duplex is not supported anymore - January 2003). (For legacy interfaces using a half-duplex protocol, you should respond to the first <ENQ> with a <NAK> to resynchronize the protocol.)

Do I need to inquire about posting charges?

If your system can support guest identification through other means (for example, virtual numbers used as PIN codes), or if the charges you send are room-based (such as Minibar), then no inquiry is necessary. For restaurant charges, inquiries should be sent only for payment methods that require guest identification. No inquiry should be sent for cash or other payment types that are sent for audit purposes (all charges are posted to a pre-configured account).

What are the recommended features for POS?

We recommend that POS systems (generally referring to guest-oriented charges) support inquiries and postings. Most hotels are interested in tracking charges by the time of day; you should include the Serving Time (ST) field to indicate breakfast, lunch, dinner, or other meal periods. Itemization (i.e., sending subtotal fields with respective tax and discount fields where applicable for various menu categories such as food, beverage, etc.) is also a high priority by many hotels. Lastly, many hotels wish to have the transfer of non-room charges such as cash, credit cards, and A/R supported.

Can monetary fields contain a decimal character? If not, do they always have two implicit decimal places?

Monetary fields contain no implicit decimal character. As most currencies support two decimal places, this is the default behavior. If you work with currencies without decimal places, you should still include them in monetary fields. If you work with more than two decimal currencies, send your amounts as is (but without the decimal character). Oracle PMS can be configured to scale the charges down by 10 to obtain the correct amount.

Do I need to send response messages for Wake-ups?

It is strongly recommended that you send them so that if a Wake-up fails, the hotel staff can be notified to wake the guest by some other means.

10

Appendix B – Code Tables

IF - Interface Types

(Used by PMS to determine the screen display for the requested interface type and activate/deactivate certain functionalities.)

Interface Type (Family)	Code
Call Accounting	CA
Key Services System (Door Locking)	DL
Energy Management	EM
Minibar	MB
TMS / PBX Gateway	PB
POS	PO
Pay TV / Extended Video Services	VI
Voice Mail	VM
Miscellaneous / Data Retrieval System	MS
In-Room Internet Systems	WW

AS - Answer Statuses

Sent by external systems to notify if a PMS request is successful or not:

Code	Supported Interface Types	Meaning
AA	PBX - DID-Response	Virtual Number already assigned
AN	PBX - DID-Response	Virtual Number not found
BM	VSS/remote check-out	Balance mismatch
BY	Wakeup/ Key Services	Telephone / Encoder Busy
CD	VSS/remote check-out	Check-out date is not today
CO	POS Systems	Posting denied because overwriting the credit limit is not allowed (not used with Suite8)
DE	Wakeup/Key	Wakeup/Key has been deleted

Code	Supported Interface Types	Meaning
DM	POS systems	The Sum of subtotals doesn't match TotalAmount
DN		Request denied
FX	Guest related requests	Guest not allowed this feature
IA	Guest related requests	Invalid account
NA	All systems	Night Audit
NF	VSS/remote check-out	Feature not enabled or Check-out process not running
NG	All information requests	Guest not found
NM	Message/Locator request	Message/Locator not found
NP	POS Systems	Posting denied for this guest (NoPost flag has been set)
NR	Wakeup	No Response
OK	All systems	Command or request completed successfully
RY	All systems	Retry
UR	All systems	Unprocessable request, this request cannot be carried out; no retry

GL - Guest Languages

The following language codes are supported:

Language	Code
English / American	EA
French	FR
German	GE
Italian	IT
Japanese	JA
Spanish	SP

 **NOTE:**

The above **red marked** codes are non-ISO code values.

FIAS will not allow translating these values into ISO Codes like DE, EN, and ES in the above cases.

Further language codes are possible to send (require special setup on the IFC application side)

Abkhazian AB	Limburgan; Limburger; Limburgish LI
Afar AA	Limburger; Limburgan; Limburgish; LI
Afrikaans AF	Limburgish; Limburger; Limburgan LI
Akan AK	Lingala LN
Albanian SQ	Lithuanian LT
Amharic AM	Luba-Katanga LU
Arabic AR	Luxembourgish; Letzeburgesch LB
Aragonese AN	Macedonian MK
Armenian HY	Malagasy MG
Assamese AS	Malay MS
Avaric AV	Malayalam ML ML
Avestan AE	Maltese MT
Aymara AY	Manx GV
Azerbaijani AZ	Maori MI
Bambara BM	Marathi MR
Bashkir BA	Marshallese MH
Basque EU	Moldavian MO
Belarusian BE	Mongolian MN
Bengali BN	Nauru NA
Bihari BH	Navaho, Navajo NV
Bislama BI	Navajo; Navaho NV
Bokmål, Norwegian; Norwegian Bokmål NB	Ndebele, North ND
Bosnian BS	Ndebele, South NR
Breton BR	Ndonga NG
Bulgarian BG	Nepali NE
Burmese MY	North Ndebele ND
Castilian; Spanish ES	Northern Sami SE
Catalan; Valencian CA CT	Norwegian Bokmål; Bokmål, Norwegian NB
Chamorro CH	Norwegian NO NR
Chechen CE	Norwegian Nynorsk; Nynorsk, Norwegian NN
Chewa; Chichewa; Nyanja NY	Nyanja; Chichewa; Chewa NY
Chichewa; Chewa; Nyanja NY	Nynorsk, Norwegian; Norwegian Nynorsk NN
Chinese ZH	Occitan (post 1500); Provençal OC

Chuang; Zhuang ZA	Ojibwa OJ
Church Slavic; Slavonic; Church Slavonic; Old Bulgarian; Old Church Slavonic CU	Old Bulgarian; Old Slavonic; Church Slavonic; Church Slavic; Old Church Slavonic CU
Church Slavonic; Church Slavic; Old Slavonic; Old Bulgarian; Old Church Slavonic CU	Old Church Slavonic; Old Slavonic; Church Slavonic; Old Bulgarian; Church Slavic CU
Chuvash CV	Old Slavonic; Church Slavonic; Old Bulgarian; Church Slavic; Old Church Slavonic CU
Cornish KW	Oriya OR
Corsican CO	Oromo OM
Cree CR	Ossetian; Ossetic OS
Croatian HR CR	Ossetic; Ossetian OS
Czech CS CZ	Pali PI
Danish DA DK	Panjabi; Punjabi PA
Divehi DV	Persian FA
Dutch; Flemish NL NL	Polish PL PL
Dzongkha DZ	Portuguese PT PO
Esperanto EO	Provençal; Occitan (post 1500) OC
Estonian ET ET	Punjabi; Panjabi PA
Ewe EE	Pushto PS
Faroese FO	Quechua QU
Fijian FJ	Raeto-Romance RM
Finnish FI FN	Romanian RO RO
Flemish; Dutch NL	Rundi RN
French FR FR	Russian RU RL/ RS
Frisian FY	Samoan SM
Fulah FF	Sango SG
Gaelic; Scottish Gaelic GD	Sanskrit SA
Gallegan GL GL	Sardinian SC
Ganda LG	Scottish Gaelic; Gaelic GD
Georgian KA	Serbian SR
Gikuyu; Kikuyu KI	Shona SN
Greek, Modern (1453-) EL GR	Sichuan Yi II
Greenlandic; Kalaallisut KL	Sindhi SD
Guarani GN	Sinhala; Sinhalese SI
Gujarati GU	Sinhalese; Sinhala SI
Haitian Creole; Haitian HT	Slovak SK SC
Haitian; Haitian Creole HT	Slovenian SL SL

Hausa HA	Somali SO
Hebrew HE	Sotho, Southern ST
Herero HZ	South Ndebele NR
Hindi HI	Sundanese SU
Hiri Motu HO	Swahili SW
Hungarian HU HU	Swati SS
Icelandic IS	Swedish SV SW
Ido IO	Tagalog TL
Igbo IG	Tahitian TY
Indonesian ID RI	Tajik TG
Interlingua (International IA	Tamil TA
Interlingue IE	Tatar TT
Inuktitut IU	Telugu TE
Inupiaq IK	Thai TH TH
Irish GA	Tibetan BO
Italian IT IT	Tigrinya TI
Japanese JA JA	Tonga (Tonga Islands) TO
Javanese JV	Tsonga TS
Kalaallisut; Greenlandic KL	Tswana TN
Kannada KN	Turkish TR TR
Kanuri KR	Turkmen TK
Kashmiri KS	Twi TW
Kazakh KK	Uighur; Uyghur UG
Khmer KM	Ukrainian UK
Kikuyu; Gikuyu KI	Urdu UR
Kinyarwanda RW	Uyghur; Uighur UG
Kirghiz KY	Uzbek UZ
Komi KV	Valencian; Catalan CA
Kongo KG	Venda VE
Korean KO	Vietnamese VI
Kuanyama; Kwanyama KJ	Volapük VO
Kurdish KU	Walloon WA
Kwanyama, Kuanyama KJ	Welsh CY
Lao LO	Wolof WO
Latin LA	Xhosa XH
Latvian LV	Yiddish YI

Letzeburgesch; Luxembourgish LB	Yoruba YO
Limburgan; Limburger; Limburgish LI	Zhuang; Chuang ZA
	Zulu ZU

KT - Key Types

Code	Meaning
N	New key request. Cancels any existing keys
D	Duplicate key request. Any existing keys remain valid/active.
O	One-shot key. The Key is only valid for use once

PT - Posting Types

Code	Meaning
C	Direct charge, the record must include the Total Amount (TA) field
M	Minibar charge, the record must include Minibar Article (MA) field, and Minibar count(M#), posting is by PMS using article number/count
T	Telephone charge, the record must include Meter Pulse (MP) field, and PMS calculates the call charge. (Not supported by PR record, only PS record.)

CS - Class of Service (COS)

Code	Meaning
0	Barred/hotel internal only
1	Local
2	National
3	No restrictions

MR, VR, TV - Guest Rights

Type	Accepted statuses
MR – Minibar rights	MU - unlock Minibar
	MN – Minibar normal vending
	ML - lock Minibar
TV – Pay TV rights	TU – unlimited pay channels (default)
	TM - no Pay movies
	TX - no Adult movies
	TN - no TV rights
VR – Video rights	VA - view bill & remote c/o (default)
	VB - only view bill
	VN - no video rights

Video rights have the following precedence:

VN, no rights;

VB, view bill only;

VA, all rights (view bill and remote check-out allowed).

It is impossible to block view bill rights and still allow remote check-out.

Pay-TV rights have the following precedence:

TN, no rights (no TV channels);

TM, all Pay channels blocked;

TX, Adult Pay channels blocked;

TU, all rights (includes all Pay channels).

With TV rights, it is impossible to block normal Pay channels and allow Adult pay channels.

RS - Room Maid Statuses

Code	Room Maid Status
1	Dirty/Vacant
2	Dirty/Occupied
3	Clean/Vacant
4	Clean/Occupied
5	Inspected/Vacant
6	Inspected/Occupied

Further values may be possible depending on the Hotels PMS setup. The maximum length for the RS field is defined with N,2. Some PMS systems can support even longer values.

 **NOTE:**

It is NOT possible to change a room to a status like "Out-of-Order" or "Out-of-Service." This influences the number of available rooms and cannot be done on an external system but only in the PMS itself.

RT - Request Types

RT in LD (Request Values In LC Record)

Code	Meaning
1	Request the configured Room payment methods in RP
4	Request PMS version in A1
8	<i>not used</i>
16	Request IFC version in A3
32	Request IFC Driver Version in A2
64	<i>not used</i>
128	Request DLS-timeout in A0

The above values can be added to combine the request.

RT in KR

Code	Meaning
1	<i>not used</i>
2	<i>not used</i>
4	Request key data in the \$3 field in the KA record

RT in XM

Code	Meaning
0	do not change message status to "received" during request action
1	change message status to "received" during request action

ES/ET - Equipment Status (DID)

EquipmentStatus (**EP**) and EquipmentStatusOld (**ET**)

Code	Meaning
0	Un-assign a DID number
1	Assign a DID-number
2	Keep an assigned number
3	Move an assigned number (Only possible during Room-Move)

11

Appendix C - Field ID

Field ID list

Field ID	Description	Format (see Data Type)	Record IDs where this field is allowed
\$1	Track 1	AN, max. 40	KR, KM
\$2	Fidelio standard Track 2 format Track2 data (UDID) from Key card Media	AN, max. 40 AN, max. 16	PR, KR KA, KM
\$3	Track3 data from Key systems -> data which is used on the key device to open the guest room	AN, max. 200	KA
\$J	Vendor Message ID (sent from vendor)	ANS, max 16	XL
A0 - A9	User Definable Fields	ANS, variable	GI, GC KR, KM PL
AS	Answer Status	ANS, 2 chars (see Answer Status table)	KA, KZ LP, PA, XC (RCKO Response),XB WA
BA	Balance Amount	N, max. 20 M, max. 20 (may include decimal point depending on local currency)	XB, XC (RCKO request), PL
BD	Item Description	ANS, max. 25	XI
BI	Item Amount	N, max. 20	XI
C#	POS - Check Number	N	PA, PL, PR, PS
CG	Cryptogram	ANS, max. 150 chars	LR, LC
CL	Credit Limit	M, max 15 (may include decimal point depending on local currency)	PL
CO	Credit Limit Override Flag	AN, 1 char (Y/N)	PR, PS
CS	Class Of Service	ANS, max. 1 (see COS table)	RE

Field ID	Description	Format (see Data Type)	Record IDs where this field is allowed
			GI/GC
CT	Clear Text	ANS, variable (depends on usage)	KR, KM, KA, KZ LO, LP, PA, PS, PR RE (VM, DN, RS), XC (RCKO response)
CV	Number Of Covers	N, max. 5	PR, PS
D1 - D9	Discount 1 – 9	M, max. 15	PR, PS
DA	Date	D YYYYMMDD	DE, DR, DS, GC, GI, GO, KA, KD, KR, KM, KZ LA, LD, LE, LS, LF, LO, LP, NS, NE, PA, PL, PR, PS, XB, XC, XD, XI, XL, XM, XR, XT, WR, WC, WA
DC	Department Code	N, max. 20	XI
DD	Dialed Digits	N, max. 20	PS, PR
DN	Do-Not-Disturb Status	AN, max. 1 (Y, enable/N, disable)	RE
DT	Departure (Check-out) Time	HH:MM (as defined in PMS)	KR (KTN, KTD), KM, KZ
DU	Duration	T	PS
EN	Equipment Number	ANS, max. 8	GI, GC, GO, VA
EP	Equipment Pool-ID	N, max. 2	GI, GC, GO, VA
ES	Equipment Status	AN, 1 char (A, assign / U, unassign)	GI, GC, GO, VA
ET	Equipment Status of source room	AN, 1 char	GC
F#	Window/Folio Number	N, 1	XI
FD	Item Display Flag	AN, 1 char (Y/N)	XI
FL	Field List	ANS, variable	LR
G#	Reservation Number	N	KD, KR (KTN, KTD), KA, KM, KZ GI, GC, GO, LO, LF, LP, PR, PL, PA, RE (ML), XB, XC, XD, XI, XL, XM, XR, XT
G+	Profile Number	N	PL, PR, GI, GC
GA	Guest Arrival Date	D	GI, GC KR (KTN, KTD), KM

Field ID	Description	Format (see Data Type)	Record IDs where this field is allowed
			PL
GD	Guest Departure Date	D	GI, GC KR (KTN, KTD), KM, KZ PL
GF	Guest First Name	ANS, max. 80	GC (Guest Info/Name Change), GI, PL
GG	Guest Group Number	AN, max. 10	GC (Guest Info/Name Change), GI, KR, KM PL
GL	Guest Language	ANS, max. 10 (see Guest Language table)	GC (Guest Info/Name Change), GI, PL
GN	Guest Name	ANS, max. 200	GC (Guest Info/Name Change), GI, KR (KT=N, KT=D), KM, KZ PA (Response to PR), PL, PR
GP	Guest PIN (DID)	N	VA
GS	Share Flag	AN, 1 char (Y/N)	GC, GI, GO
GT	Guest Title	ANS, max. 20	GC (Guest Info/Name Change), GI, PL
GV	Guest VIP Status	AN, max. 20	GC (Guest Info/Name Change), GI, PL
ID	User ID	ANS, max. 16	PA, PL, PR, PS, KR, KD, KM RE
IF	Interface Family	ANS, 2 chars (see Interface Type table)	LD
K#	Key Count	N, max. 2	KR
KC	Key Coder	ANS, max. 8	KA, KD, KR, KZ, KM
KO	Key Options	ANS, max. 20	KR (KT=N & KT=D), KZ
KT	Key Type	AN, max. 1 (see Key Type table)	KR
LT	Locator Expiry Time	HHMM	LO, LP
M#	Number Of Articles	N, max. 2	PR, PS
MA	Minibar Article	N, max. 4	PR, PS
MX	Maximum Guest Match	N, max. 2	PR
MI	Message ID	N	XD, XL, XM, XT

Field ID	Description	Format (see Data Type)	Record IDs where this field is allowed
ML	Message Light Status	AN, 1 char (Y/N)	RE
MP	Meter Or Tax Pulse	N, max. 10	PS
MR	Minibar Rights	ANS, 2 char (see Guest Rights table)	GC (Guest Info/Name Change), GI, RE (Minibar)
MT	Message Text	ANS, variable (max 1000, or as defined in the LD-record)	XL, XT
NP	No Post Flag	Y/N	GI, GC, PL
P#	Posting Sequence Number	N, max. 8 (value > 0 only)	PA, PL, PR, PS
PC	Posting Call Type	AN, 1 char	PR, PS
PH	Hotel-ID	ANS, max. 20	LC
PI	Inquiry Data	ANS, max. 10	PR
PM	Vendor Payment Method / PMS Payment Method	ANS, max. 5 ANS, max. 5	PR, PS PL
PP	Printer Port	N, 1	RE (VM, DN, RS),
PT	Posting Type	AN, 1 char (see Posting Type table)	PR (except PTT), PS
PX	Posting Route (i.e. Trunk)	N, max. 6	PS
PU	Number of Persons	N, max. 2	RE
RI	Record ID	ANS, 2 chars	LR
RL	Maximum <i>Message</i> Record Length	N, variable (max. record length is 2000)	LD
RN	Room Number	ANS	GC, GI, GO, RA, RE, KD, KR, KM, KZ LF, LO, LP, PA, PL, PR, PS, XB, XC, XD, XI, XL, XM, XR, XT, WA, WC, WR VA
RO	Old Room Number	ANS	GC (Room Move) KM (Room Move)
RP	Configured Roompayment methods	ANS, no max.	LC
RS	Room Maid Status	N, 1 (see Room Maid Status table)	RE
RT	Request Type	AN, (see Request Type table)	LD KR, KM XM

Field ID	Description	Format (see Data Type)	Record IDs where this field is allowed
S1 -S9	Subtotal 1 – 9	M, max. 15	PR, PS
SC	Service Charge	M, max. 15	PR, PS
SF	Swap Flag	No data (if this field is sent, the record is part of a DB swap)	GI, GO
SI	Suite Info	ANS, max. 30	KR, KD , KM, KZ
SO	Sales Outlet	N, max. 5	PA, PL, PR, PS
ST	Serving Time	N, max. 4	PR, PS
T#	Table Number	N, max. 4	PR, PS
T1 - T9	Tax 1 – 9	M, max. 15	PR, PS
TA	Total Posting Amount	N, max 15	PS, PR,
TI	Time (System time)	T, HHMMSS	DE, DR, DS, GC, GI, GO, KA, KD, KR, KM, KZ LA, LD, LE, LS, LF, LO, LP, NE, NS, PA, PL, PR, PS, XB, XC, XD, XI, XL, XM, XR, XT,
	Wakeup time	T, HHMMSS	WA, WC, WR
TP	Tip	M, max. 15	PR, PS
TV	TV Rights	ANS, 2 char (see Guest Rights table)	GC (Guest Info/Name Change), GI, RE
UO	Suite Info for Old Room	ANS, max. 30	KM
V#	Vendor Version Number	ANS, max. 10	LD
VM	Voice Mail	ANS, max. 4	RE
VR	Video Rights	ANS, 2 char (see Guest Rights table)	GC (Guest Info/Name Change), GI,
WS	Workstation ID	ANS, max. 16	GI, GC, GO, KA, KD, KR, KM, KZ PA, PL, PR, PS
X1	Cross Reference Data	ANS, max. 25	PS, PR