Programme: E-ELT

Project/WP: System Engineering

ICD between the ELT TRS and the TRS Clients

Document Number: ESO-331947

Document Version: 2

Document Type: Interface Control Document (ICD)

Released On: 2021-06-07

Document Classification: ESO Internal Use [Confidential for Non-ESO Staff]

Owner: Chiozzi, Gianluca

Validated by PM: Kornweibel, Nick

Validated by PM: González Herrera, Juan Carlos

Validated by SE: González Herrera, Juan Carlos

Validated by PE: Biancat Marchet, Fabio

Approved by PGM: Tamai, Roberto

Authors

Name	Affiliation
C.Soenke	ESO

Change Record from previous Version

Affected Section(s)	Changes / Reason / Remarks
	See CRE ET-1085
2.2	Updated RD1 to Version 2
6.1.2	Added URI for request/reply interface
6.2	Specified port number for diagnostic interface
6.2	Added fields appSpecific2-4
4	Minor textual improvements/clarifications
7	Added new section for Linux PTP monitoring daemon local interface
4.3.3	Added requirement related to non-PTP traffic.
6.1.3	Changed command names to UpperCamelCase
6.3	Added paragraph for monitoring of PTP enabled network devices in LCIs

Contents

1. Introduction

1.1 Scope

[I-TRS/TRS-5] This Interface Control Document (ICD) defines the interface requirements between the ELT Time Reference System (TRS) and its clients.
/ / /

[I-TRS/TRS-6] In particular this ICD covers the following interfaces categories:
/ / /

- High accuracy synchronization via PTP (IEEE1588)
- Medium accuracy synchronization via NTP
- House Keeping System

[I-TRS/TRS-7] This ICD does not cover the interfaces related to:
/ / /

- ESO-provided software for Linux servers related to clock synchronisation, accessing time, task scheduling and time scale conversions.

1.2 Interfacing Elements List

[I-TRS/TRS-9] For the purpose of this ICD the TRS is composed by:
/ / /

- The Master Clock System
- The Time Distribution Network
- The House Keeping System

[I-TRS/TRS-10] For the purpose of this ICD the clients are:
/ / /

- Any system that needs synchronization via PTP
- Any system that needs synchronization via NTP
- Any system that needs to communicate with the House Keeping System (to request information or to send client diagnostic data)
- any PTP enabled network device that is to be monitored by the TRS HKS

- any Linux application software that needs to access the status of the local PTP monitoring daemon

1.3 Definitions and Conventions

1.3.1 Statement Identifiers and Verification Tags

[I-TRS/TRS-13] / / / Each statement in this document is labelled with an identifier following the format [I-TRS/TRS-NNN], where NNN is a unique, non-speaking number.

[I-TRS/TRS-14] / / / NNN numbers do not necessarily follow a sequential order. They do not change across all versions of this document. Within this document, cross-references to an item are made by referring to the number NNN preceded by the prefix "#".

1.3.2 Abbreviations and Acronyms

[I-TRS/TRS-16] / / / The following abbreviations and acronyms are used in this document:

API	Application Programming Interface
BC	Boundary Clock
CII	Core Integration Infrastructure (software)
CS	(ELT) Control System
GMC	Grand Master Clock
GNSS	Global Navigation Satellite System
HKS	House Keeping System
LCI	Local Communication Infrastructure
MCS	Master Clock System
MJD	Modified Julian Date
NTP	Network Time Protocol
PTP	Precision Time Protocol
SCP	Service Connection Point
TAI	Temps Atomique International
TBC	To Be Confirmed
TBD	To Be Defined
TC	Transparent Clock
TDN	Time Distribution Network

TRS	Time Reference System
UT1	Universal Time
UTC	Universal Time Coordinated

2. Related Documents

2.1 Applicable Documents

[I-TRS/TRS-19] / / / The following documents, of the exact version shown, form part of this document to the extent specified herein. In the event of conflict between the documents referenced herein and the content of this document, the content of this document shall be considered as superseding.

2.1.1 ESO Documents

AD1 ICD between the Network Equipment and its Clients
ESO-320983 Version 1
https://pdm.eso.org/kronodoc/HQ/ESO-320983/1

2.1.2 External Standards

AD2 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems
IEEE Std 1588-2008

AD3 RFC 5905, Network Time Protocol Version 4

2.2 Reference Documents

[I-TRS/TRS-26] / / / The following documents, of the exact version shown herein, are listed as background references only. They are not to be construed as a binding complement to the present document.

RD1 Technical Requirements Specification for the E-ELT Time Reference System
ESO-308444 Version 2
https://pdm.eso.org/kronodoc/HQ/ESO-308444/2

RD2 MUDPI Format
ESO-302020 Version 4
https://pdm.eso.org/kronodoc/HQ/ESO-302020/4

RD3 Design Description for CII MAL (Middleware Abstraction Layer);
ESO-348603 Version 1
https://pdm.eso.org/kronodoc/HQ/ESO-348603/1

3. Interface Description

3.1 Interface Overview

[I-TRS/TRS-32] / / /

This document describes the various interfaces between the ELT Time Reference System (TRS) and its clients.

Clients requiring high accuracy synchronization will interface with the TRS via the Precision Time Protocol (PTP) provided on the Time Distribution Network. For this purpose, section 4 describes the 'ELT PTP client profile' which defines all relevant protocol options.

The achievable synchronization **accuracy for PTP will be < 2µs** w.r.t. the MCS (see RD1).

Clients requiring medium accuracy synchronization will interface with the TRS via the Network Time Protocol (NTP). Section 5 defines this interface.

The achievable synchronization **accuracy for NTP will be < 30ms** w.r.t. the MCS (see RD1).

The TRS House Keeping System provides information such as TRS status and other time-related parameters. Clients can request this information through a dedicated interface described in section 6.1.

Section 6.2 defines a network protocol by which clients shall report their own status and performance parameters to the TRS House Keeping System.

Section 6.3 defines the requirements for PTP enabled network devices which are part of an LCI to be monitorable by the TRS HKS.

Section 7 specifies the interface between the Linux PTP monitoring daemon and Linux software applications.

4. PTP (IEEE1588) Interface

4.1 Protocol and version

[I-TRS/TRS-35] D/ / / The protocol for high accuracy synchronization is PTP (IEEE1588) version 2 as specified in AD2.

4.2 Network interfaces

[I-TRS/TRS-37] D/ / / PTP is provided on the Time Distribution Network which is available in the computer room and on selected SCP-Bs

[I-TRS/TRS-38] D/ / / The supported physical layers are described in AD1.

4.3 The ELT PTP client profile

4.3.1 PTP attributes

[I-TRS/TRS-41] D/ / / The ELT PTP profile is based on the Delay Request-Response Default PTP profile as specified in section J.3 of AD2

[I-TRS/TRS-42] D/ / / The PTP transport mechanism is UDP over IPv4

[I-TRS/TRS-43] D/ / / PTP messages shall be sent using IP multicast

[I-TRS/TRS-44] D/ / / The PTP domain is 0

[I-TRS/TRS-45] D/ / / The timescale is PTP (TAI)

[I-TRS/TRS-46] D/ / / The best master clock algorithm shall be the algorithm specified by section 9.3.2 of AD2 (default algorithm)

[I-TRS/TRS-47] D/ / / Clients shall use the End-to-end (Delay_Req/Delay_Resp) delay mechanism

[I-TRS/TRS-48] D/ / / Clients shall work in slave-only mode

[I-TRS/TRS-49] D/ / / Clients shall use the following priorities:

- priority 1: 255
- priority 2: 255

[I-TRS/TRS-50] D/ / / Clients shall accept 2-step correction synchronization messages, i.e. Sync + FollowUp

4.3.2 Message rates

[I-TRS/TRS-52] D/ / / logAnnounceInterval: The default initialization value shall be 0 (1s). The configurable range

shall be 0 to 2 (1s to 4s)

Note: intervals are specified in logarithmic notation, i.e.

$$interval[s] = 2^{logInterval}$$

[I-TRS/TRS-53] D/ / / logSyncInterval: The default initialization value shall be 0 (1s). The configurable range shall be −3 to +1 (1/8 s to 2s)

[I-TRS/TRS-54] D/ / / logMinDelayReqInterval: The default initialization value shall be 0 (1s). The configurable range shall be -3 to 3 (1/8 s to 8s)

[I-TRS/TRS-55] D/ / / announceReceiptTimeout: The default initialization value shall be 3. The configurable range shall be in the range 2 to 10

Note:

$$timeout[s] = announceReceiptTimeout \times 2^{logAnnounceInterval}$$

4.3.3 Network related aspects

[I-TRS/TRS-57] D/ / / Clients may be connected to the Time Distribution Network through boundary clocks, transparent clocks or PTP agnostic network devices. The network path between client and GMC may be a combination of all before mentioned devices.

Note: for compatibility with transparent clocks, clients must take into account the correction field in PTP messages. In order to work with the different types of transparent

clocks (one-step & two-step), the client must sum the correction fields of both Sync and FollowUp messages (see AD2).

[I-TRS/TRS-58] D/ / / The Time Distribution Network does not provide any particular support for redundant client connections. For clients that require a redundant PTP connection:

- on the SCP-B access switches, redundancy is not possible
- in the computer room, clients can connect to the Time Reference Network with two separate ports on two separate access switches. The ports on the switches are handled as individual unrelated ports and the network infrastructure is unaware of them being used for redundancy purposes. This allows clients for example to use channel bonding in active-backup mode or to implement other redundancy mechanisms which do not require support from the network infrastructure.

[I-TRS/TRS-59] D/ / / Failures in the TRS - either caused by the master clocks, or the Time Distribution Network - will be recovered automatically. In such cases, clients will experience transition periods of up to approx. 30s where the PTP network reconfigures - a normal and foreseen scenario.

[I-TRS/TRS-60] D/ / / Due to the use of multicast, it is mandatory that clients implement IGMP (see also AD1).

[I-TRS/TRS-61] D/ / / Clients may receive the PTP traffic related to other clients, in particular Delay_Req & Delay_Resp messages.

Note: this implies that clients shall only use Delay_Response messages when they can be associated to a previously send Delay_Request message of the client itself (fields requestingSourcePortIdentity & requestingSequenceId). In a segment of the network where TCs are used, a client could in theory receive several hundred messages per second (assuming for example 100 clients on the segment and 2 DelayReq + 2 DelayResp messages per second)

[I-TRS/TRS-112] / / / Clients shall not emit any traffic on their TDN interface other than:

- PTP protocol messages
- PTP management messages
- 'control pane' traffic such as IGMP, LLDP, ARP, ...
- traffic related to TRS house-keeping

5. NTP Interface

5.1 Protocol and version

[I-TRS/TRS-64] D/ / / The protocol for medium accuracy synchronization is NTP version 4 as specified in AD3.

5.2 Network interfaces

[I-TRS/TRS-66] D/ / / NTP is available in the following networks:

- General services
- Control Network, non-deterministic
- Interlock and Safety

[I-TRS/TRS-67] D/ / / The supported physical interfaces are specified in AD1.

5.3 Client settings

[I-TRS/TRS-69] D/ / / Clients shall use the following NTP settings unless they have special requirements:

- minpoll 5 (32s)
- maxpoll 9 (512s)

6. House Keeping System Interface

6.1 Command interface

6.1.1 Purpose

[I-TRS/TRS-73] D/ / / The House Keeping System implements a request/reply interface for clients to query status and time-related information.

Intended use-cases are:

- CCS requesting DUT1 at the beginning of the night
- CCS checking if leap seconds are scheduled to warn operator
- TRS display units requesting status information
- provide information such that devices in the control system can sanity-check their time related settings (leap seconds, time)
- engineering purposes

In general, the interface is intended for low-rate usage only.

Note: While GET_TIME will return the current time, clients should obviously not use this for synchronization but should use NTP or PTP.

6.1.2 Protocol

[I-TRS/TRS-75] D/ / / The interface uses the CII-standard request/reply protocol (see RD3).

[I-TRS/TRS-76] D/ / / The command interface is available in the non-deterministic control network.

[I-TRS/TRS-96] D/ / / The connection URI is zpb.rr://<trs-host>:6618/trsinfo

where <trs-host> is the hostname/IP address of the TRS House-Keeping server

6.1.3 Commands

[I-TRS/TRS-78] D/ / /

Command: **GetStatus**

Parameters: none

Description: Query the status of the TRS

Reply: status string
OK - when the TRS is operating normally
WARNING - reports a non-critical status
ERROR - at least one component of the TRS has failed

[I-TRS/TRS-79] D/ / /

Command: **GetTime**

Parameters: string: format

Description: Get the current time & date. The format parameter specifies one of the following supported timescales/representations:

- UTC
- TAI
- TAI_seconds
- UNIX
- MJD
- Sidereal

Note that precision is limited to 0.1 seconds.

Reply: The reply is a string containing one of the following:
- 'WRONG_ARG' if the argument is not valid
- UTC as 'YYYY-MM-DDThh:mm:ss.f'
- TAI as 'YYYY-MM-DDThh:mm:ss.f'
- TAI_seconds as seconds since 1970-01-01T00:00:00 TAI as double
- UNIX as seconds since 1970-01-01T00:00:00 UTC (POSIX) as double

- MJD as 'NNNNN.fffff' representing days since 1958-11-17
- Sidereal time at the configured telescope location in format 23h05m21.1199s'

[I-TRS/TRS-80] D/ / /

Command: **GetDut1**

Parameters: none

Description: Query the current UT1 - UTC offset.

Reply: string containing DUT1 as floating point value in seconds

[I-TRS/TRS-81] D/ / /

Command: **GetPolarOffsets**

Parameters: none

Description: Get the current polar offsets based on IERS data.

Reply: String containing the polar offsets in arcseconds as 'x.xxxxx y.yyyyy'

[I-TRS/TRS-82] D/ / /

Command: **GetLeapsecCurrent**

Parameters: none

Description: Get the current leap seconds.

Reply: String containing a signed integer in seconds.

[I-TRS/TRS-83] D/ / /

Command: **GetLeapsecDate**

Parameters: none

Description: Get the date of the next scheduled leap seconds insertion.

Reply: String containing:

'NOT_SCHEDULED' if no leap second is scheduled

'YYYY-MM-DDThh:mm:ss ss' if a leap second is scheduled. The first value is the date of the transition in UTC, the second value are the leap seconds after the transition.

[I-TRS/TRS-84] D/ / /

Command: **Convert**

Parameters: string: in_format

string: out_format

string: time

Description: Convert time from in_format to out_format. The format parameters specify one of the following supported timescales/representations:

- UTC
- TAI
- TAI_seconds
- UNIX
- MJD
- Sidereal

Reply: The reply is a string containing one of the follwoing:

- 'WRONG_ARG' if any argument is not valid
- UTC as 'YYYY-MM-DDThh:mm:ss'
- TAI as 'YYYY-MM-DDThh:mm:ss'
- TAI_seconds as seconds since 1970-01-01T00:00:00 TAI
- UNIX as seconds since 1970-01-01T00:00:00 UTC (POSIX)
- MJD as 'NNNNN.fffff' representing days since 1958-11-17
- Sidereal time at the configured telescope location in format '23h05m21.1199s'

6.2 Diagnostic logging interface for PTP clients

[I-TRS/TRS-86] D/ / / PTP clients shall send their status information to the TRS House Keeping System periodically.

Note: the diagnostic logging on Linux servers using the ELT DevEnv is provided by the TRS Linux Client Toolkit

[I-TRS/TRS-87] D/ / / The message rate shall be configurable between 1s and 60s. The default rate shall be 10s.

[I-TRS/TRS-88] D/ / / Messages shall be sent on the non-deterministic control network.

[I-TRS/TRS-89] D/ / / Messages shall use the MUDPI format as specified in RD2

[I-TRS/TRS-90] D/ / / The following UPD ports shall be used:

- Source: any
- Destination: configurable with default 6617

[I-TRS/TRS-91] D/ / / Messages shall be sent as unicast messages to the TRS House Keeping System host.

[I-TRS/TRS-92] D/ / / The payload data shall follow the format specified in chapter 6 (MUDPI messages) of RD2, i.e. they are little endian.

[I-TRS/TRS-93] D/ / / The MUDPI header shall be built using the following values:

- topicID: configurable with default 6617
- ComponentId: 0 or componentId if known
- ApplicationTag: 0
- FrameId: 1
- NumFrames: 1

[I-TRS/TRS-94] D/ / /

The payload has the following format:

Offset [bytes]	Field	Type	Description
0	Version	Unsigned 16	Version of the TRS diagnostics logging protocol. Fixed at 1.
2	deviceType	Unsigned 16	Type of the client device: 0: not defined 1: Linux server with HW timestamping 2: Linux server with SW timestamping 10: PLC – Beckhoff 19: PLC - other 20: WFS camera 30: Embedded device 40: Boundary clock 50: Meinberg N2X 51: Meinberg HPS100
4	hostname	Char[16]	Hostname, IP address or name of the device This field shall allow to unambiguously identify the device. Encoding: ASCII Padding: '\0'
20	portState1	Unsigned 16	State of the primary PTP port (matches section 8.2.5.3.1 of IEEE1588-v2, except for value 0): 0: not available 1: INITIALIZING 2: FAULTY 3: DISABLED: 4: LISTENING 5: PRE_MASTER 6: MASTER 7: PASSIVE 8: UNCALIBRATED 9: SLAVE
22	portState2	Unsigned 16	State of the secondary PTP port (if used), values as for portState2
24	offsetFromMaster	Signed 64	Current offset from master clock in ns.

			0 if not supported
32	meanPathDelay	Signed 64	Mean path delay in ns. 0 if not supported
40	announceMsgCount	Signed 32	Counter of Announce messages. -1 if not supported
44	syncMsgCount	Signed 32	Counter of Sync messages. -1 if not supported
48	delayRespCount	Signed 32	Counter of Delay_Resp messages. -1 if not supported
52	clockIdentity	Char[8]	Clock identity. 0x 00 00 00 00 00 00 00 00 if not supported.
60	gmIdentity	Char[8]	Grandmaster identity 0x 00 00 00 00 00 00 00 00 if not supported.
68	parentPortIdentity	Char[10]	Parent port identity 0x 00 00 00 00 00 00 00 00 00 00 if not supported.
78	currentUtcOffset	Signed 16	Current offset UTC/TAI. 999 if not supported.
80	stepsRemoved	Unsigned 16	Steps removed. 999 if not supported.
82	errorStatus	Unsigned 16	The error status may be used to communicate any kind of errors on the client. The errorStatus could be used to signal situations such as 'PTP network interface down', 'PTP service not running', etc … 0: not supported or no error 1 - 65535: application specific error
84	appSpecific1	Signed 32	Any application specific value to be processed/logged together with the other diagnostic information. 0 if not used
88	appSpecific2	Signed 32	See appSpecific1
92	appSpecific3	Signed 32	See appSpecific1
96	appSpecific4	Signed 32	See appSpecific1
100	reserved1	Signed 32	Reserved for future use.

			0 if not used
104	reserved2	Signed 32	Reserved for future use 0 if not used

6.3 Monitoring of PTP enabled network devices in LCIs

[I-TRS/TRS-114] D/ / / If instruments or telescope subsystems employ network devices in their LCI that act as PTP boundary clocks, these devices shall be monitorable by the TRS HKS.

[I-TRS/TRS-120] / / / The TRS HKS periodically polls the information from the device with a period of 10s to 60s.

[I-TRS/TRS-115] D/ / / The following PTP related parameters shall be accessible for monitoring:

- portState1 (PTP state of the port that is the primary PTP uplink)
- portState2 (PTP state of the port that is secondary PTP uplink, only applicable in case of redundant uplinks)
- offsetFromMaster
- meanPathDelay
- clockIdentity
- grandmasterIdentity
- parentPortIdentity
- currentUtcOffset
- stepsRemoved

[I-TRS/TRS-116] D/ / / The monitored parameters shall be accessible over a protocol such as SNMP, RESTful or comparable. The underlaying protocol shall be unicast UDP or TCP.

[I-TRS/TRS-118] / / / The monitoring API shall be accessible over the Control non-deterministic network.

7. Linux PTP Monitoring Daemon Local Interface

[INFO-TRS/TRS-99] The PTP monitoring daemon provided with the TRS Linux Client Toolkit monitors the local PTP daemon and sends diagnostic data to the House-Keeping System as per paragraph 6.2. The daemon provides a local interface for applications to query status and receive notifications upon state changes - this interface is described hereafter.

The daemon is referred to as 'ptpmond'

[I-TRS/TRS-100] D/ / / The interface between the ptpmond and any application SW is based on Unix Domain Sockets and uses the socket type SOCK_SEQPACKET.

[I-TRS/TRS-111] D/ / / The ptpmond acts as the server and listens on the default address "\0ptpmon" (abstract socket address).

[I-TRS/TRS-101] D/ / / Once a client has connected to the server, it can either request status information as needed or subscribe to events to be notified on state changes. Both mechanisms can be combined in one connection.

[I-TRS/TRS-102] D/ / / The server accepts multiple simultaneous connections.

[I-TRS/TRS-103] D/ / / All messages exchanged between client and server start with a UINT8 field designating the payload type. Messages have a variable length depending on their payload type.

[I-TRS/TRS-104] D/ / / Payload type **STATUS_REQ** (0x0):

Client requesting the current status. No additional fields.

[I-TRS/TRS-105] D/ / / Payload type **STATUS_REPLY** (0x1):

Reply to a STATUS_REQ. Additional fields are:

status (UINT8) - 0x0 (GOOD) or 0x2 (BAD)

msg_len (UNIT8) - length in bytes of the msg_text

msg_text (CHAR * msg_len) - extended status information, ASCII encoding, not null terminated

[I-TRS/TRS-106] D/ / / Payload type **SUBSCRIBE_REQ** (0x2):

Client subscribing to event notifications. No additional fields.

[I-TRS/TRS-107] D/ / / Payload type **SUBSCRIBE_REPLY** (0x3):

Server acknowleging subscription.

[I-TRS/TRS-108] D/ / / Payload type **NOTIFICATION** (0x4):

Server sending an event notification. Additional fields as for STATUS_REPLY.

[I-TRS/TRS-109] D/ / /

The server sends notifications on the following events:

- when synchronization status changes between GOOD and BAD
- when the server is about to terminate

[I-TRS/TRS-110] D/ / /

The server will not notify all internal state changes of the PTP daemon. The status BAD will only be set upon fatal failures or when synchronization performance is insufficient. Transitional states that are caused e.g. by a switch-over in a bonded interface are not considered as failures as long as the recovery succeeds within certain boundaries.