

Recommendation

ITU-T H.627.3 (12/2022)

SERIES H: Audiovisual and multimedia systems

Broadband, triple-play and advanced multimedia services –
Advanced multimedia services and applications

Protocols for intelligent video surveillance systems

ITU-T H-SERIES RECOMMENDATIONS
AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence, immersive environments, virtual and extended reality	H.420–H.439
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
VEHICULAR GATEWAYS AND INTELLIGENT TRANSPORTATION SYSTEMS (ITS)	
Architecture for vehicular gateways	H.550–H.559
Vehicular gateway interfaces	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
Content delivery and ubiquitous sensor network applications	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	H.700–H.719
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SYSTEMS, SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	H.820–H.859
Multimedia e-health data exchange services	H.860–H.869
Safe listening	H.870–H.879

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T H.627.3

Protocols for intelligent video surveillance systems

Summary

Recommendation ITU-T H.627.3 defines protocols for intelligent video surveillance systems, including the functional architecture, functional interface, overall requirements of the protocol, message flows and relevant protocols. This Recommendation is based on Recommendation ITU-T H.626.5, *Architecture for intelligent video surveillance systems*.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T H.627.3	2022-12-14	16	11.1002/1000/15204

Keywords

Intelligent video surveillance, protocol.

* To access the Recommendation, type the URL <http://handle.itu.int/> in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2023

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

	Page
1 Scope	1
2 References.....	1
3 Definitions	1
3.1 Terms defined elsewhere	1
3.2 Terms defined in this Recommendation	2
4 Abbreviations and acronyms	3
5 Conventions	3
6 Introduction to functional architecture of an IVS system.....	4
7 Overall requirements.....	4
7.1 Introduction of relevant protocol.....	4
7.2 Interface description.....	7
8 Message flows and relevant protocols	16
8.1 Message flow	16
8.2 Interface definition	19
Annex A – Data type and object JSON-Schema description	36
A.1 Data type	36
A.2 JSON-Schema description	38
Bibliography	74

Recommendation ITU-T H.627.3

Protocols for intelligent video surveillance systems

1 Scope

This Recommendation defines protocols for intelligent video surveillance (IVS) systems, including the functional architecture, functional interface, overall requirements of the protocol, message flows and relevant protocols. The protocols of the video surveillance (VS) system refer to [ITU-T H.627].

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [ITU-T H.626.5] Recommendation ITU-T H.626.5 (V2) (2022), *Architecture for intelligent video surveillance system*.
- [ITU-T H.627] Recommendation ITU-T H.627 (V2) (2020), *Signalling and protocols for a video surveillance system*.
- [IETF RFC 2045] IETF RFC 2045 (1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*.
- [IETF RFC 2046] IETF RFC 2046 (1996), *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*.
- [IETF RFC 3339] IETF RFC 3339 (2002), *Date and Time on the Internet: Timestamps*.
- [IETF RFC 3986] IETF RFC 3986 (2005), *Uniform Resource Identifier (URI): Generic Syntax*.
- [IETF RFC 7616] IETF RFC 7616 (2015), *HTTP Digest Access Authentication*.
- [IETF RFC 9110] IETF RFC 9110 (2022), *HTTP Semantics*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 centre management unit (CMU) [ITU-T H.626.5]: The centre management unit (CMU) is located at the centre of the video surveillance (VS) system. Its main functions include centralized system management, service operation management, etc.

3.1.2 client intelligent video (CIV) [ITU-T H.626.5]: An intelligent identification module in the customer unit (CU). It identifies required information from input video and outputs the result and retrieves the recorded video data with specified information.

3.1.3 data storage and service (DSS) [ITU-T H.626.5]: The storage and sharing centre for video and image information within the intelligent video surveillance system. The DSS accepts registration, keepalive reports from the intelligent premises unit (IPU), intelligent video management (IVM) or intelligent application service (IAS) and obtains video and image information from them. It can also

provide information network sharing services such as information querying, subscription, guard and alarm, etc.

3.1.4 intelligent application service (IAS) [ITU-T H.626.5]: The intelligent application service (IAS) is located at the application service core of the intelligent video surveillance system. Its main functions include providing application services about the video and image information, such as information query and retrieval, subscription and notification, manual annotation, etc.; providing external operations such as the creation and start-up of intelligent analysis tasks based on intelligent video management (IVM); obtaining the list of devices from the central control server (CCS) or data storage and service (DSS) and sharing them with the IVM so that it can schedule intelligent video units (IVUs) for video stream analysis, etc.

3.1.5 intelligent customer unit (ICU) [ITU-T H.626.5]: The client subsystem within the intelligent video surveillance system. The client intelligent video (CIV) is added to the customer unit (CU) in order to achieve comprehensive intelligent video analysis.

3.1.6 intelligent premises unit (IPU) [ITU-T H.626.5]: The premises subsystem within the intelligent video surveillance system. The premises intelligent video (PIV) module is added to the premises unit (PU) for intelligent video analysis.

3.1.7 intelligent video management (IVM) [ITU-T H.626.5]: The intelligent video management (IVM) unit supports strategies configuration of intelligent applications by users and video sources schedules dynamically. The IVM accepts registration, deletion, capability reports from intelligent video units (IVUs) and schedules the IVUs dynamically. It can also store, manage, and schedule those capabilities dynamically.

3.1.8 intelligent video unit (IVU) [ITU-T H.626.5]: The intelligent video unit (IVU) identifies specific objects automatically, and outputs recognition results to an intelligent video management (IVM) system. The recognition information includes triggered events and acquired data. One or more intelligent analysis algorithms can be loaded or unloaded on an intelligent video unit (IVU) according to different requirements.

3.1.9 media distribution unit (MDU) [ITU-T H.626.5]: The media distribution unit (MDU) is used to transport media from the premises unit (PU) to the customer unit (CU). Its main functions include media receiving, media processing, media routing, media transmission, media forwarding and media replication.

3.1.10 media storage unit (MSU) [ITU-T H.626.5]: The media storage unit (MSU) is used to retrieve and store media and provide media serving capability. Its main functions include media storage, media serving, media indexing and media downloading.

3.1.11 premises intelligent video (PIV) [ITU-T H.626.5]: An intelligent identification module in the premises unit (PU). It identifies required information from input video and outputs the result.

3.1.12 service-control unit (SCU) [ITU-T H.626.5]: The service-control unit (SCU) is located at the centre of the video surveillance (VS) system. It is a distributed network equipment, fulfilling the access functions of the premises unit (PU) and the customer unit (CU). Its main functions include access registration, access authentication, identification, authorization, call control, location, presence and target media serving function selection.

3.1.13 video and image information [ITU-T H.626.5]: Video clips, images, related files and content description information.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

APE	Acquisition and Processing Equipment
API	Application Programming Interface
CCS	Central Control Server
CIV	Client Intelligent Video
CMU	Centre Management Unit
CU	Customer Unit
DSS	Data Storage and Service
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol over Secure Socket Layer
IAS	Intelligent Application Service
ICU	Intelligent Customer Unit
IPU	Intelligent Premises Unit
IVM	Intelligent Video Management
IVS	Intelligent Video Surveillance
IVU	Intelligent Video Unit
JSON	JavaScript Object Notation
MDU	Media Distribution Unit
MS	Media Server
MSU	Media Storage Unit
PIV	Premises Intelligent Video
PU	Premises Unit
REST	Representation State Transfer
SBI	Searched By Image
SCU	Service-Control Unit
URI	Universal Resource Identifier
URL	Uniform Resource Locator

5 Conventions

In this Recommendation, the following conventions apply:

- The keywords "is required to" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.
- The keywords "is recommended" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

6 Introduction to functional architecture of an IVS system

The functional architecture of an IVS system is illustrated in Figure 6-1. For the detailed entities and functions, see clause 6.2 of [ITU-T H.626.5].

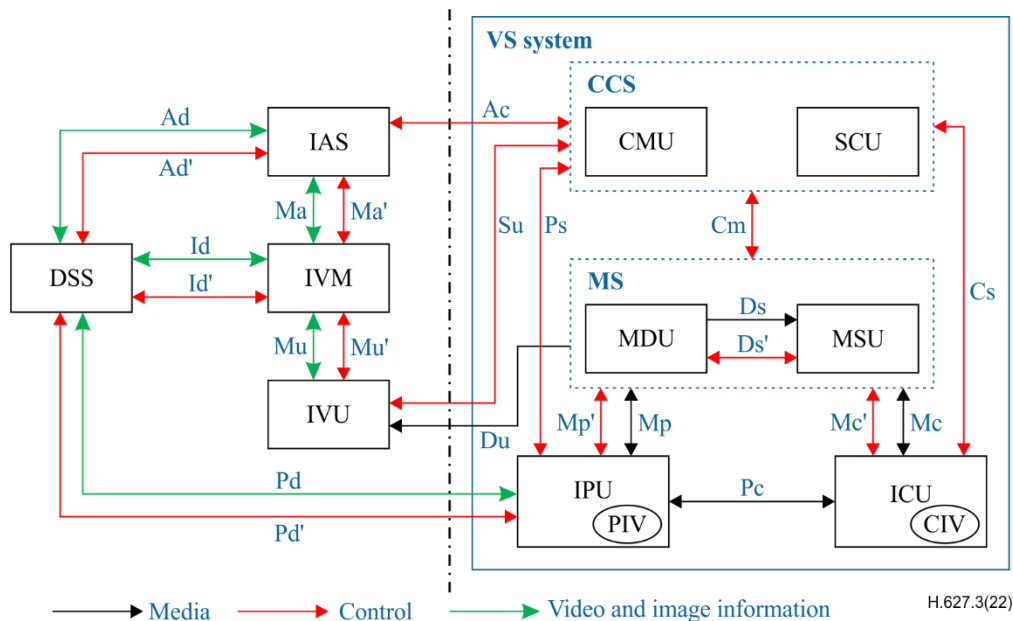


Figure 6-1 – Functional architecture of IVS system

7 Overall requirements

7.1 Introduction of relevant protocol

This Recommendation uses the representation state transfer (REST) style [b-REST] to design the application programming interface (API). The implementation of the protocol based on Hyper Text Transfer Protocol (HTTP) or Hyper Text Transfer Protocol over Secure Socket Layer (HTTPS).

7.1.1 Protocol requirements

7.1.1.1 URI

Universal resource identifier (URI) uses nouns to describe the provided resources. The definition is required to meet the requirements of section 4.2 of [IETF RFC 9110] and support Path Templating. Path Templating refers to the usage of curly brace({}) to mark a section of a uniform resource locator (URL) path as replaceable using path parameters.

7.1.1.2 Request header

The request header field is required to be extended to add <User-Identify>. The field name User-Identify is used to carry the system ID of the requester to identify the requester's identity.

7.1.1.3 Request method

Request method (section 9 of [IETF RFC 9110]) is used to describe operations performed on resources, the specific correspondence is as follows:

- GET: Read resources, call cannot change the resource state, only return data to the requester without any side effects.
- POST: Create a resource, it can only be used to add a resource that does not exist before.
- PUT: Update resources, generally used to update existing resources, if the specified resource does not exist, it can also be created.

d) **DELETE:** Delete resources.

7.1.1.4 Message body

The encoding method of the message body adopts UTF-8. Under normal circumstances, the interface message body is required to be encapsulated in JavaScript object notation (JSON) format, and the Content-Type header field is required to be set to application/json. When carrying video clips or images, Base64 encodes the data into a string and stores it in JSON. The encoding method is required to meet the requirements of [IETF RFC 2045]; when transmitting large video clip data or image data is recommended to be encapsulated by multipart, and the encapsulation method is required to meet the requirements of section 5.1 of [IETF RFC 2046].

7.1.1.5 Status code

Status code is used to describe the result of operations performed on resources, and its definition is required to meet the requirements of section 15 of [IETF RFC 9110].

7.1.2 Query requirements

The query part of the URI is required to meet the requirements of section 3.4 of [IETF RFC 3986], and is required to support regular expressions.

7.1.2.1 Expansion requirements

This Recommendation extends the query content:

$$\text{query} = (\text{o.p1} \text{ OP } \text{v1}) * \& ((\text{o.p2} \text{ OP } \text{v2}) \text{ OP2 } (\text{o.p3} \text{ OP } \text{v3})) *$$

In the expression, o indicates the object to be queried, p indicates the property of the object, and * indicates that the preceding () and the content inside appear 0 or more times. Operator OP is required to comply with Table 1, Table 2 and Table 3, OP2 takes the value of AND or OR. The parameters of the object attribute limit, sort field, number of records and other parameters of the result returned by the query is required to comply with the regulations in Table 4.

Table 1 – Arithmetic operators

Operator	Meaning
+	addition
-	subtraction
*	multiplication
/	division
%	Returns the integer remainder of a division operation

Table 2 – Logical operators

Operator	Meaning
AND	If both Boolean expressions are TRUE, then TRUE
OR	If one of the two Boolean expressions is TRUE, then TRUE

Table 3 – Comparison operators

Operator	Meaning
=	equal
>	more than
<	Less than
>=	greater or equal to
<=	Less than or equal to
<>	not equal to
!=	not equal to
!<	not less than
!>	not greater than

Table 4 – Conventions of variable names and meanings

Variable name	Meaning	Description
Fields	Used to specify the returned query property collection	Multiple properties are separated by ","
Sort	Specify sort field	"-" means descending order, otherwise ascending order, placed before the sort field
PageRecordNum	Specify the number of records per page	
RecordStartNo	Specify the start record number of the first page	
MaxNumRecordReturn	Maximum number of records returned	

7.1.2.2 Query example

- a) Search for person records with red or pink packages, and the returned results are sorted by the upper age limit, and only two attributes of PersonID and SourceID are returned.

```
GET /Persons?((Person.BagColor = Red) OR (Person.BagColor =
Pink))&(Sort = Person.AgeUpLimit)&(Fields =
(Person.PersonID, Person.SourceID))
```

- b) Find person information, only query the first 200 records, 25 records per page, starting from the first record, sort by age upper limit, and return all attributes.

```
GET /Persons?(MaxNumRecordReturn = 200)&(PageRecordNum =
25)&(RecordStartNo = 1) &(Sort = Person.AgeUpLimit)
```

- c) Query ApeID contains 110109 ape information

```
GET /APes?(APE.ApeID like 110109)
```

7.1.3 Safety requirements

The two parties involved in the information exchange is required to conduct identity authentication. The authentication process is required to follow the digital digest authentication specified in [IETF RFC 7616] and is required to follow [IETF RFC 9110]. During digest authentication, the server random number nonce is required to be reset after it exceeds a limited period of time, and the limited period of time is recommended as 3600 s.

7.2 Interface description

7.2.1 Public interface

- It is required to support registration, de-registration, keep-alive, and time calibration.
- It is required to re-register after a random time within 300 s, when the registration fails.
- It is required to support both registration parties configure the "heartbeat interval" parameter (the default heartbeat interval is 90 s).
- It is required to support the registrant periodically send heartbeat messages to the registered party to keep the heartbeat alive according to the "heartbeat interval", after the successful registration.
- It is required to support both registration parties configure the "Keep-alive timeout times" parameter uniformly (the default number of heartbeat timeouts is 3 times).

Comparative table of registration, de-registration, keep-alive, and time calibration with interface message is shown in Table 5.

Table 5 – Comparative table of registration, de-registration, keep alive, time calibration with interface message

No.	Interface name	Description	Interface message clause	Method
1	registration	Registration	8.2.1	POST
2	de-registration	De-registration	8.2.2	POST
3	keep_alive	Keep alive	8.2.3	POST
4	time_calibration	Time calibration	8.2.4	POST

7.2.2 Interfaces of reference point Pd/Pd'

- It is required to support registration, de-registration, keep_alive, and time_calibration functions from IPU to DSS. Interfaces as shown in Table 5.
- It is required to support upload automatically acquisition video and image information from IPU to DSS. Interfaces as shown in Table 6.
- It is required to support retransmission, when upload failed, after uploading the information.
NOTE 1 – Upload failed means that the response is not received within a certain time (e.g., 5 s).
- It is required to support cache information, de-registration, and re-registration, when link problem happened. It is required to support transmission cache information after re-registration successful.
NOTE 2 – Link problem means that unsuccessful retransmission multiple times (e.g., 3 times).

Table 6 – Comparative table of automatic video and image information uploading with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_video_clips_create_and_query	Upload video clips automatically in batch.	DSS	IPU	8.2.6.1	POST
2	batch_image_create_and_query	Upload image in batch automatically.	DSS	IPU	8.2.7.1	POST
3	batch_motor_vehicles_Create_query_update_delete	Upload motor vehicle in batch automatically.	DSS	IPU	8.2.8.1	POST
4	batch_non-motor_vehicles_create_query_update_delete	Upload non-motor vehicle in batch automatically.	DSS	IPU	8.2.9.1	POST

7.2.3 Interfaces of reference point Ma/Ma'

- It is required to support registration, de-registration, keep_alive, and time_calibration functions from IVM to IAS. Interfaces as shown in Table 5.
- It is required to support retrieving target by image and attribution between IAS and IVM. Interfaces as shown in Table 7.

Table 7 – Comparative table of target retrieving by image and attribution with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	synchronously_target_query_by_image	Create a task to target query by image (synchronization).	IVM	IAS	8.2.10.1	POST
2	asynchronously_create_and_delete_target_query_by_image	Create and delete a task to target query by image (asynchronous).	IVM	IAS	8.2.10.2	POST, DELETE
3	create_and_delete_target_retrieval_by_image	Upload a result of object to target query by image.	IAS	IVM	8.2.10.3	POST
4	batch_motor_vehicles_create_query_update_and_delete	Query motor vehicle by attribution.	IVM	IAS	8.2.8.1	GET
5	batch_non-motor_vehicles_create_query_update_and_delete	Query non-motor vehicle by attribution.	IVM	IAS	8.2.9.1	GET

- It is recommended to support analysis task controlling and analysis task status query between IAS and IVM. Interfaces as shown in Table 8.

Table 8 – Comparative table of analysis task control and status query with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	analysis_task_control	Pause, resume, and stop analysis task.	IVM	IAS	8.2.12	POST
2	analysis_task_status_query	Query analysis task status.	IVM	IAS	8.2.11	GET

7.2.4 Interfaces of reference point Ad/Ad'

- It is required to support registration, de-registration, keep_alive, and time_calibration functions from IAS to DSS. Interfaces as shown in Table 5.
- It is required to support queries based on feature attributes such as video and image information objects and their combinations, business operations and structural objects from IAS to DSS. Interfaces as shown in Table 9.

Table 9 – Comparative table of feature attributes and their combination query with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_video_clips_create_and_query	Query video clips in batch.	DSS	IAS	8.2.6.1	GET
2	a_single_video_clip_query_update_and_delete	Query a single video clip.	DSS	IAS	8.2.6.2	GET
3	a_single_video_clip_query_update_and_delete_object_information	Query object information of video clips.	DSS	IAS	8.2.6.3	GET
4	batch_image_create_and_query	Query images in batch.	DSS	IAS	8.2.7.1	GET
5	a_single_image_query_update_and_delete	Query a single image.	DSS	IAS	8.2.7.2	GET
6	a_single_image_query_update_and_delete_object_information	Query image object information.	DSS	IAS	8.2.7.3	GET
7	batch_motor_vehicles_create_query_update_and_delete	Query motor vehicles in batch.	DSS	IAS	8.2.8.1	GET
8	batch_motor_vehicle_create_query_update_delete	Query a single motor vehicle.	DSS	IAS	8.2.8.2	GET
9	batch_non-motor_vehicles_create_query_update_and_delete	Query non-motor vehicles in batch.	DSS	IAS	8.2.9.1	GET
10	a_single_non-motor_vehicle_query_update_and_delete	Query a single non-motor vehicle.	DSS	IAS	8.2.9.2	GET

Table 9 – Comparative table of feature attributes and their combination query with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
11	batch_video_image_event_create_and_query	Query video and image events in batch.	DSS	IAS	8.2.14.1	GET
12	a_single_video_image_event_query_update_and_delete	Query a single video and image event.	DSS	IAS	8.2.14.2	GET
13	a_single_video_image_event_query_update_and_delete_object_information	Query object information of video and image event.	DSS	IAS	8.2.14.3	GET
14	batch_disposition_create_query_update_and_delete	Query disposition task.	DSS	IAS	8.2.15.1	GET
15	disposition_notification	Query disposition notification record.	DSS	IAS	8.2.16	GET
16	batch_subscription_create_query_update_and_delete	Query subscription task.	DSS	IAS	8.2.17.1	GET
17	subscription_notification	Query notification record.	DSS	IAS	8.2.18	GET
18	create_and_delete_target_retrival_by_image	Query to retrieve the target result by image.	DSS	IAS	8.2.10.3	GET

- It is required to support the manual acquisition of video and image events and the creation, update and deletion of video and image information objects from IAS to DSS. Interfaces as shown in Table 10.

Table 10 – Comparative table of manual acquisition of video and image information object with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_video_image_event_create_and_query	Create video and image events in batch.	DSS	IAS	8.2.14.1	POST
2	a_single_video_image_event_query_update_and_delete	Update and delete a single video and image event.	DSS	IAS	8.2.14.2	PUT, DELETE
3	a_single_video_image_event_query_update_and_delete_object_information	Update and delete video and image event object information.	DSS	IAS	8.2.14.3	PUT, DELETE
4	batch_video_clips_create_and_query	Create manually acquisition video clips in batch.	DSS	IAS	8.2.6.1	POST

Table 10 – Comparative table of manual acquisition of video and image information object with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
5	a_single_video_clip_query_update_and_delete	Update and delete a single video clip acquisition manually.	DSS	IAS	8.2.6.2	PUT, DELETE
6	A_single_video_clip_query_update_and_delete_object_information	Update and delete the object information of manually acquisition video clips.	DSS	IAS	8.2.6.3	PUT, DELETE
7	batch_image_create_and_query	Create manually acquisition images in batch.	DSS	IAS	8.2.7.1	POST
8	a_single_image_query_update_and_delete	Update and delete a single acquisition image manually.	DSS	IAS	8.2.7.2	PUT, DELETE
9	a_single_image_query_update_and_delete_object_information	Update and delete manually acquisition image object information.	DSS	IAS	8.2.7.3	PUT, DELETE
10	batch_motor_vehicles_create_query_update_and_delete	Create, update and delete motor vehicles in batch.	DSS	IAS	8.2.8.1	POST, PUT, DELETE
11	batch_motor_vehicle_create_query_update_delete	Update and delete a single motor vehicle.	DSS	IAS	8.2.8.2	PUT, DELETE
12	batch_non-motor_vehicles_create_query_update_and_delete	Create, update and delete non-motor vehicles in batch.	DSS	IAS	8.2.9.1	POST, PUT, DELETE
13	a_single_non-motor_vehicle_query_update_and_delete	Update and delete a single non-motor vehicle	DSS	IAS	8.2.9.2	PUT, DELETE

- It is required to support query and update IPU from IAS to DSS, and to query video tollgates, lanes and other information. Interfaces as shown in Table 11.

Table 11 – Comparative table of query and update the acquisition device/system with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	acquisition_device_query_and_update	Query the acquisition device.	DSS	IAS	8.2.5	GET
2	acquisition_device_query_and_update	Change the password of the acquisition device.	DSS	IAS	8.2.5	PUT
3	video_tollgate_query	Query the video tollgate.	DSS	IAS	8.2.13.1	GET
4	lane_query	Query the lanes.	DSS	IAS	8.2.13.2	GET

- It is required to support the disposition and notification of video and image information objects and the retrieval of targets by image between IAS and DSS. Interfaces as shown in Table 12.

Table 12 – Comparative table of disposition and notification with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_disposition_create_query_update_and_delete	Disposition in batch.	DSS	IAS	8.2.15.1	POST
2	batch_disposition_create_query_update_and_delete	Update and delete disposition tasks in batch.	DSS	IAS	8.2.15.1	PUT, DELETE
3	withdraw_disposition	Withdraw disposition.	DSS	IAS	8.2.15.2	PUT
4	disposition_notification	Notification.	IAS	DSS	8.2.16	POST
5	disposition_notification	Delete notification record.	IAS	DSS	8.2.16	DELETE
6	synchronously_target_query_by_image	Create a task to target query by image (synchronization).	DSS	IAS	8.2.10.1	POST
7	asynchronously_create_and_delete_target_query_by_image	Create and delete a task to target query by image (asynchronous).	DSS	IAS	8.2.10.2	POST, DELETE
8	create_and_delete_target_retrieval_by_image	Upload a result of object to target query by image.	IAS	DSS	8.2.10.3	POST

- It is required to support subscriptions and notifications of video and image information objects, IPU directories, etc. between IAS and DSS. Interfaces as shown in Table 13.

Table 13 – Comparative table of subscription and notification with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_subscription_create_query_update_and_delete	Subscription in batch.	DSS	IAS	8.2.17.1	POST
2	batch_subscription_create_query_update_and_delete	Update and delete subscription tasks in batch.	DSS	IAS	8.2.17.1	PUT, DELETE
3	unsubscribe	Unsubscription.	DSS	IAS	8.2.17.2	POST
4	subscription_notification	Notification.	IAS	DSS	8.2.18	POST
5	subscription_notification	Delete notification record.	IAS	DSS	8.2.18	DELETE

7.2.5 Interfaces of reference point Id/Id'

- It is required to support registration, de-registration, keep alive, and time calibration functions from IVM to DSS. Interfaces as shown in Table 5.
- It is required to support upload automatically acquisition video and image information from IVM to DSS. Functions as shown in Table 14.
- It is required to support retransmission, when upload failed, after uploading the information.
NOTE 1 – Upload failed means that the response is not received within a certain time (e.g., 5 s).
- It is required to support cache information, de-registration, and re-registration, when link problem happened. It is required to support transmission cache information after re-registration successful.

NOTE 2 – Link problem means that unsuccessful retransmission multiple times (e.g., 3 times).

Table 14 – Comparative table of uploading the automatically acquisition video and image information with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_video_clips_create_and_query	Upload automatically acquisition video clips.	DSS	IVM	8.2.6.1	POST
2	batch_image_create_and_query	Upload automatically acquisition images.	DSS	IVM	8.2.7.1	POST
3	batch_motor_vehicles_create_query_update_delete	Upload automatically acquisition motor vehicles.	DSS	IVM	8.2.8.1	POST
4	batch_non-motor_vehicles_create_query_update_delete	Upload automatically acquisition non-motorized vehicles.	DSS	IVM	8.2.9.1	POST

- It is required to support queries based on feature attributes such as video and image information objects and their combinations from IVM to DSS. Interfaces as shown in Table 15.

Table 15 – Comparative table of feature attributes and their combination query with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_video_clips_create_and_query	Query video clips in batch.	DSS	IVM	8.2.6.1	GET
2	a_single_video_clip_query_update_and_delete	Query a single video clip.	DSS	IVM	8.2.6.2	GET
3	a_single_video_clip_query_update_and_delete_object_information	Query object information of video clips.	DSS	IVM	8.2.6.3	GET
4	batch_image_create_and_query	Query images in batch.	DSS	IVM	8.2.7.1	GET
5	a_single_image_query_update_and_delete	Query a single image.	DSS	IVM	8.2.7.2	GET
6	a_single_image_query_update_and_delete_object_information	Query image object information.	DSS	IVM	8.2.7.3	GET
7	batch_motor_vehicles_create_query_update_and_delete	Query motor vehicles in batch.	DSS	IVM	8.2.8.1	GET
8	batch_motor_vehicle_create_query_update_delete	Query a single motor vehicle.	DSS	IVM	8.2.8.2	GET
9	batch_non-motor_vehicles_create_query_update_and_delete	Query non-motor vehicles in batch.	DSS	IVM	8.2.9.1	GET
10	a_single_non-motor_vehicle_query_update_and_delete	Query a single non-motor vehicle.	DSS	IVM	8.2.9.2	GET
11	batch_video_image_event_create_and_query	Query video and image events in batch.	DSS	IVM	8.2.14.1	GET
12	a_single_video_image_event_query_update_and_delete	Query a single video and image event.	DSS	IVM	8.2.14.2	GET
13	a_single_video_image_event_query_update_and_delete_object_information	Query object information of video and image event.	DSS	IVM	8.2.14.3	GET
14	batch_disposition_create_query_update_and_delete	Query disposition task.	DSS	IVM	8.2.15.1	GET
15	disposition_notification	Query notification record.	DSS	IVM	8.2.16	GET

Table 15 – Comparative table of feature attributes and their combination query with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
16	batch_subscription_create_query_update_and_delete	Query subscription task.	DSS	IVM	8.2.17.1	GET
17	subscription_notification	Query notification record.	DSS	IVM	8.2.18	GET
18	create_and_delete_target_retrieval_by_image	Query to retrieve the target result by image.	DSS	IVM	8.2.10.3	GET

- It is required to support the disposition and notification of video and image information objects and the retrieval of targets by image between IVM and DSS. Interfaces as shown in Table 16.

Table 16 – Comparative table of disposition and notification with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_disposition_create_query_update_and_delete	Disposition in batch.	IVM	DSS	8.2.15.1	POST
2	batch_disposition_create_query_update_and_delete	Update and delete disposition tasks in batch.	IVM	DSS	8.2.15.1	PUT, DELETE
3	withdraw_disposition	Withdraw disposition.	IVM	DSS	8.2.15.2	PUT
4	disposition_notification	Notification.	DSS	IVM	8.2.16	POST
5	disposition_notification	Delete notification record.	DSS	IVM	8.2.16	DELETE
6	synchronously_target_query_by_image	Create synchronization for image retrieval target task.	IVM	DSS	8.2.10.1	POST
7	synchronously_create_and_delete_target_query_by_image	Create and delete asynchronous target tasks for image retrieval.	IVM	DSS	8.2.10.2	POST, DELETE
8	create_and_delete_target_retrieval_by_image	Upload the target results for image retrieval.	DSS	IVM	8.2.10.3	POST

- It is required to support subscriptions and notifications of video and image information objects, acquisition device / system directories, etc. between IVM and DSS. Interfaces as shown in Table 17.

Table 17 – Comparative table of subscription and notification with interface message

No.	Interface name	Description	Interface provider	Interface caller	Interface message clause	Method
1	batch_subscription_create_query_update_and_delete	Subscription in batch.	DSS	IVM	8.2.17.1	POST
2	batch_subscription_create_query_update_and_delete	Update and delete subscription tasks in batch.	DSS	IVM	8.2.17.1	PUT, DELETE
3	unsubscription	Unsubscription.	DSS	IVM	8.2.17.2	POST
4	subscription_notification	Notification.	IVM	DSS	8.2.18	POST
5	subscription_notification	Delete notification record.	IVM	DSS	8.2.18	DELETE

8 Message flows and relevant protocols

8.1 Message flow

8.1.1 Create resource message interaction process

The interface caller creates the video and image information object for the interface provider. The interface caller sends an HTTP POST request to the interface provider according to the interface access form, and the interface provider returns a response of whether the creation is successful or not to the interface caller. The interactive process is shown in Figure 2.

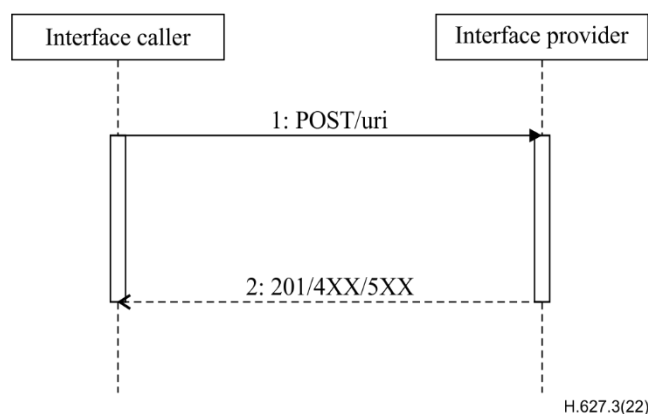


Figure 2 – Creation flow

The message process is described as follows:

- 1) The interface caller sends an HTTP POST request to the interface provider for /uri corresponding resource creation.
- 2) The interface provider returns a response to the interface caller regarding whether the creation is successful or not. Response code 201 is returned for success; Response code 4XX is returned for failure caused by the interface caller, and the message body carries the corresponding failure reason; Response code 5XX is returned for failure caused by the interface provider, and the message body carries the corresponding failure reason.

8.1.2 Query resource message interaction process

The interface caller can query the video and image information object, acquisition device and other information from the interface provider. The interface caller sends an HTTP GET request to the interface provider according to the interface access form, and the interface provider returns the reading result to the interface caller. The interactive process is shown in Figure 3.

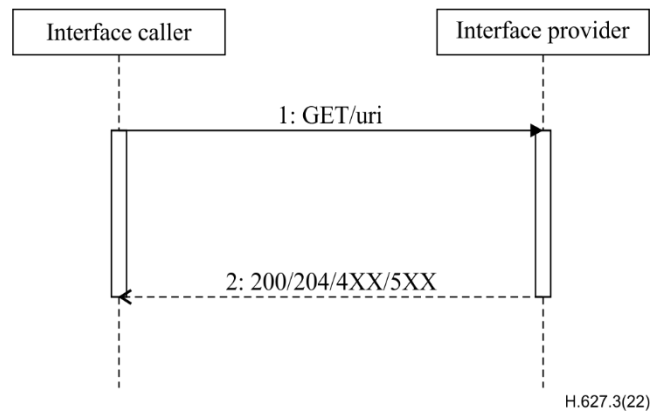


Figure 3 – Query flow

The message process is described as follows:

- 1) The interface caller sends an HTTP GET request to the interface provider for /uri corresponding resources. If pagination is required, specify it in the query condition.
- 2) The interface provider processes the combined query condition and returns the results that meet the conditions. Successfully processed and found a qualified result, response code 200 is returned, and the message body carries the result; the successful processing and no qualified result is found, response code 204 is returned, and the message body is empty; the failure caused by the interface caller returns response code 4XX, and the message body carries the corresponding failure reason; the failure caused by the interface provider returns response code 5XX, and the message body carries the corresponding failure reason.

8.1.3 Update resource message interaction process

The interface caller updates the video and image information object, disposition and announcement object, subscription and notification object, etc. to the interface provider; the interface caller sends an HTTP PUT request to the interface provider according to the interface access form, and the interface provider returns the update result to the interface caller. The interactive process is shown in Figure 4.

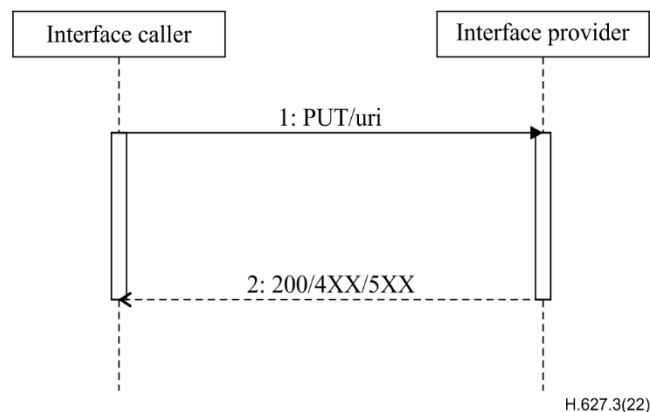


Figure 4 – Update flow

The message process is described as follows:

- 1) The interface caller sends an HTTP PUT update request to the interface provider for /uri corresponding resources.
- 2) The interface provider returns a response of whether the update is successful or not to the interface caller. Response code 200 is returned for success; Response code 4XX is returned for failures caused by the interface caller, and the message body carries the corresponding failure reason; Response code 5XX is returned for failures caused by the interface provider, and the message body carries the corresponding failure reason.

8.1.4 Delete resource message interaction process

The interface caller deletes the video and image information object, disposition and announcement object, the subscription and notification object, etc. to the interface provider. The interface caller sends an HTTP DELETE request to the interface provider according to the interface access form, and the interface provider returns the delete result to the interface caller. The interactive process is shown in Figure 5.

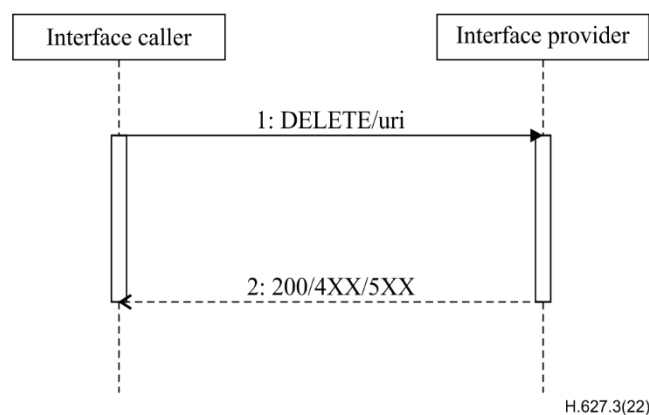


Figure 5 – Delete flow

The message process is described as follows:

- 1) The interface caller sends an HTTP DELETE to delete request to the interface provider for /uri corresponding resources, and the resource identifier to be deleted is specified in the query string.
- 2) The interface provider returns a response of whether the deletion is successful or not to the interface caller. Response code 200 is returned for success; response code 4XX is returned for failures caused by the interface caller, and the message body carries the corresponding failure reason; response code 5XX is returned for failures caused by the interface provider, and the message body carries the corresponding failure reason.

8.1.5 Asynchronous operation message interaction process

The interface caller performs asynchronous operations such as disposition and announcement, subscription and notification, and image retrieval of the target to the interface provider. The interface caller sends an HTTP POST request to the interface provider according to the interface access form, and the interface provider asynchronously returns the result to the interface caller after a period of processing. The interactive process is shown in Figure 6.

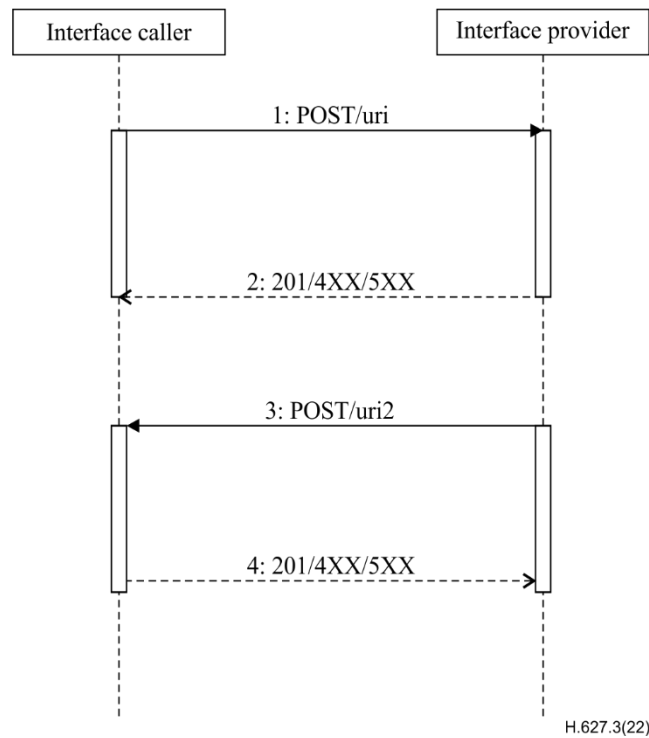


Figure 6 – Asynchronous operation flow

The message process is described as follows:

- 1) The interface caller sends an HTTP POST request to the interface provider to perform asynchronous operations for /uri corresponding resources.
- 2) The interface provider returns a response to the interface caller regarding whether the reception is successful or not. Response code 201 is returned for success; response code 4XX is returned for failure caused by the interface caller, and the message body carries the corresponding failure reason; Response code 5XX is returned for failure caused by the interface provider, and the message body carries the corresponding failure reason.
- 3) The interface provider processes the asynchronous request and sends the processing result to /uri2 specified by the interface caller via HTTP POST.
- 4) The interface caller returns a response to the interface provider regarding whether the reception is successful or not. Response code 201 is returned for success; response code 4XX is returned for failure caused by the interface caller, and the message body carries the corresponding failure reason; response code 5XX is returned for failure caused by the interface provider, and the message body carries the corresponding failure reason.

8.2 Interface definition

The interface definition description is shown in Table 18. Among them, the URI item describes the interface resource request URI; the function item comprehensively describes the functional behaviour; the method item lists the HTTP methods involved in the resource, including POST, GET, PUT, and DELETE; the query string item describes the key-value pair of the resource; the message body item describes the message body of the HTTP request; the returned result item describes the response message body of the HTTP request; use "<object tag>" to point the object that constitutes the message body and the returned result; the comment item gives a special description of the interface message. The interface message body and response results are described by the JSON Schema shown in Annex A.