Road Communication Standards :

Message Set Standard (Ver.1.05)

February 2006

National Institute for Land and
Infrastructure Management
Ministry of Land, Infrastructure and Transport

Contents

1 Objective

This standard describes message sets which can be used in the case that road agencies procure ITS systems. By applying this standard to many ITS systems, interoperability between different systems can be achieved, and then the intelligent operation and efficient procurement of ITS systems are expected.

The message sets described in this standard represent collections of messages that are exchanged between ITS systems. These message sets are defined for respective component that corresponds to a communication processing function common to various ITS systems. Combination of several message sets could facilitate the system design of diversified ITS systems.

These message sets are not expressed in a conventional fixed format, but in a flexible way as follows.

1) Associate each element contained in a message with mandatory/optional. It allows the reduction of transmission data volume in the case of bandwidth-limited link.
2) Define a group of data elements (which are repeatedly used in ITS systems) as a data set. It widens the applicability of message sets through the selection of appropriate data sets or the addition of new data sets.
3) Define some of data elements in a data set as CHOICE type. It facilitates a selection of data collection conditions, for example. That is, a period or accuracy of data measurement can be selected among several alternatives.

2 Scope

The scope of this message set standard is described in this chapter. Firstly, the relation between this standard and other standards of Road Communication Standards (i.e. data dictionary and protocol standards) is explained. Then, interface points and characteristics of message sets are described.

2.1 Relation to Other Standards

2.1.1 Relation to the Protocol Standard

This standard is related to the ASDU (Application Service Data Unit) within the APDU (Application Protocol Data Unit) shown in the figure below.

The ASDU is typically composed of an application control section and a data section. This standard provides for the application control section as the common message header in Section 5. As for the data section, message sets of publication and subscription types are defined in Section 6.

One ASDU may include plural pairs of the Application Control Part and Data Part as shown in the following figure. It means that one message may be divided into plural pairs, or plural messages are transmitted in one message sequence.

Service header and control information in the APDU are described in the Protocol Standard of Road Communication Standards.

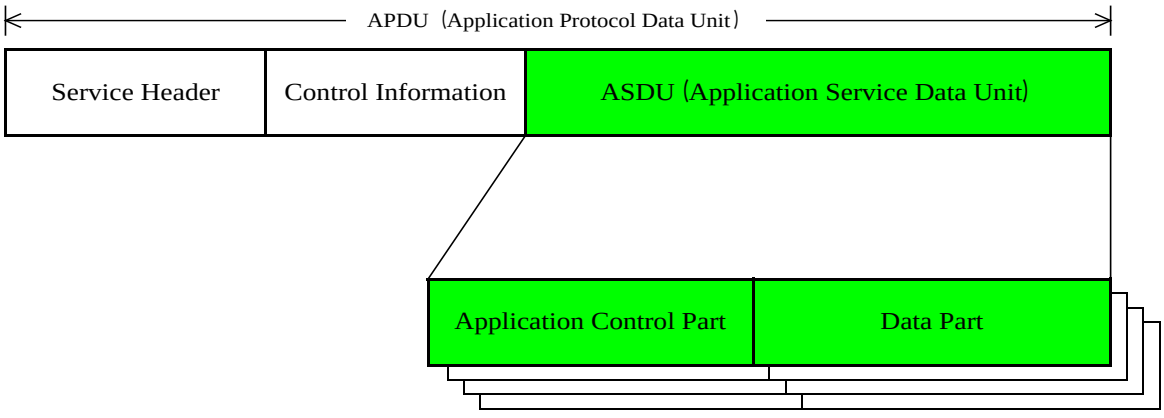


Fig. 2-1 APDU and ASDU

Message Sequence

The message set standard specifies data exchange processing within a message sequence, while message sequence itself is specified in the protocol standard.

Session initiation and termination processes are described in the protocol standard. As for the security issue, user authentication (e.g. log-in message) and user terminal authentication (e.g. authentication message) are also described in the protocol standard.

2.1.2 Relation to the Data Dictionary Standard

This message set standard describes ASN.1 representation of message set. The semantics of data elements that make up message sets are defined in the data dictionary standard.

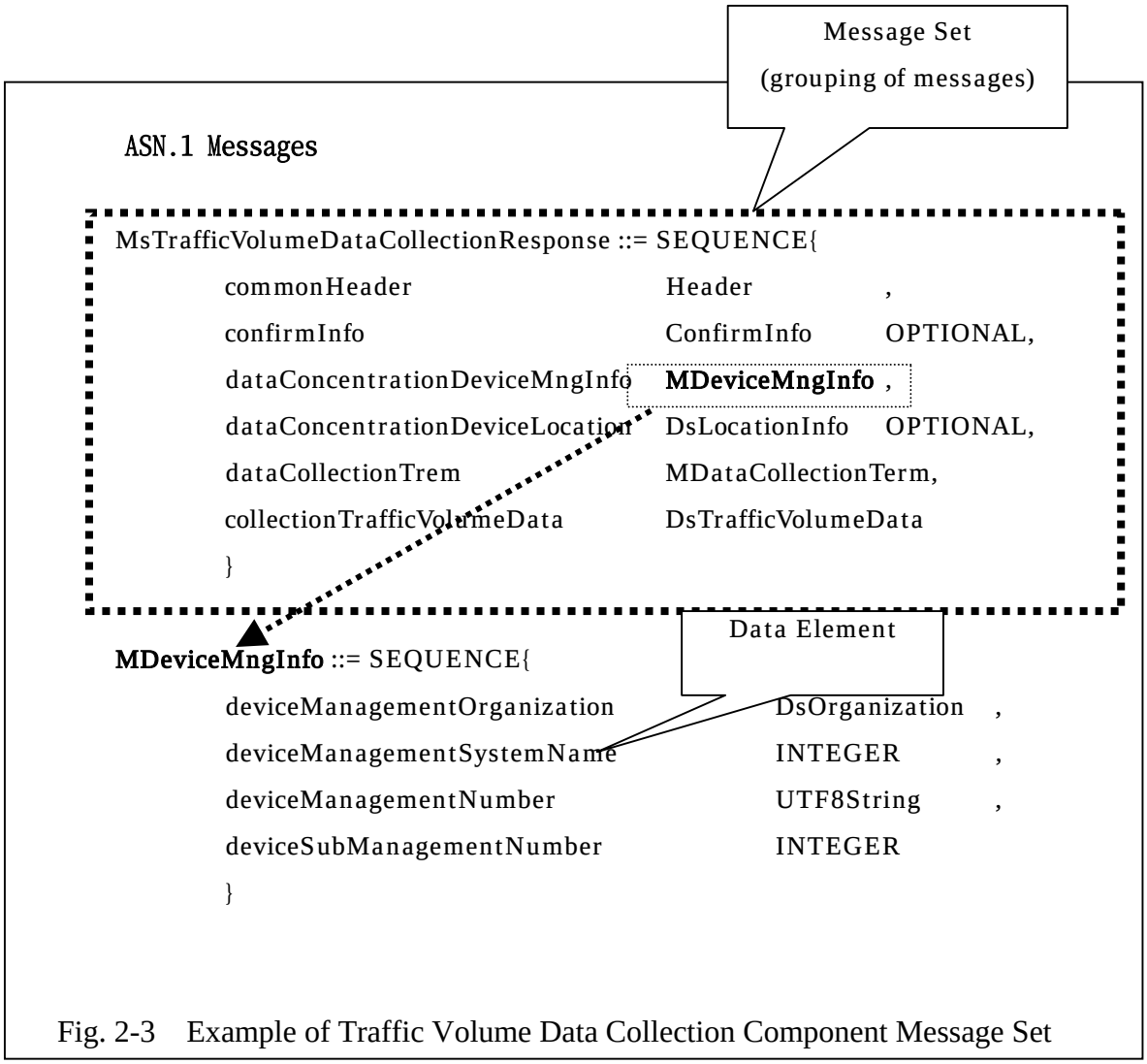


Fig. 2-3 Example of Traffic Volume Data Collection Component Message Set

Data Sets

In this standard, a group of data elements (which are repeatedly used in ITS systems) is defined as a data set, and then universality and expandability are secured.

Let's assume the case that data sets are not defined in this standard. Then, message set structure would not be flexible, and the message set would have to be newly defined whenever service menu or operational conditions of the ITS system are changed. In designing new ITS systems, where data elements not included in the existing message sets are required, new message sets would have to be designed. This situation is described in the upper portion of the following figure.

By using data sets, expandability of message sets is ensured. The lower portion of the figure shows that, if new data elements are required in the message set, new data set is created and it is located in the shaded place of the existing message set.

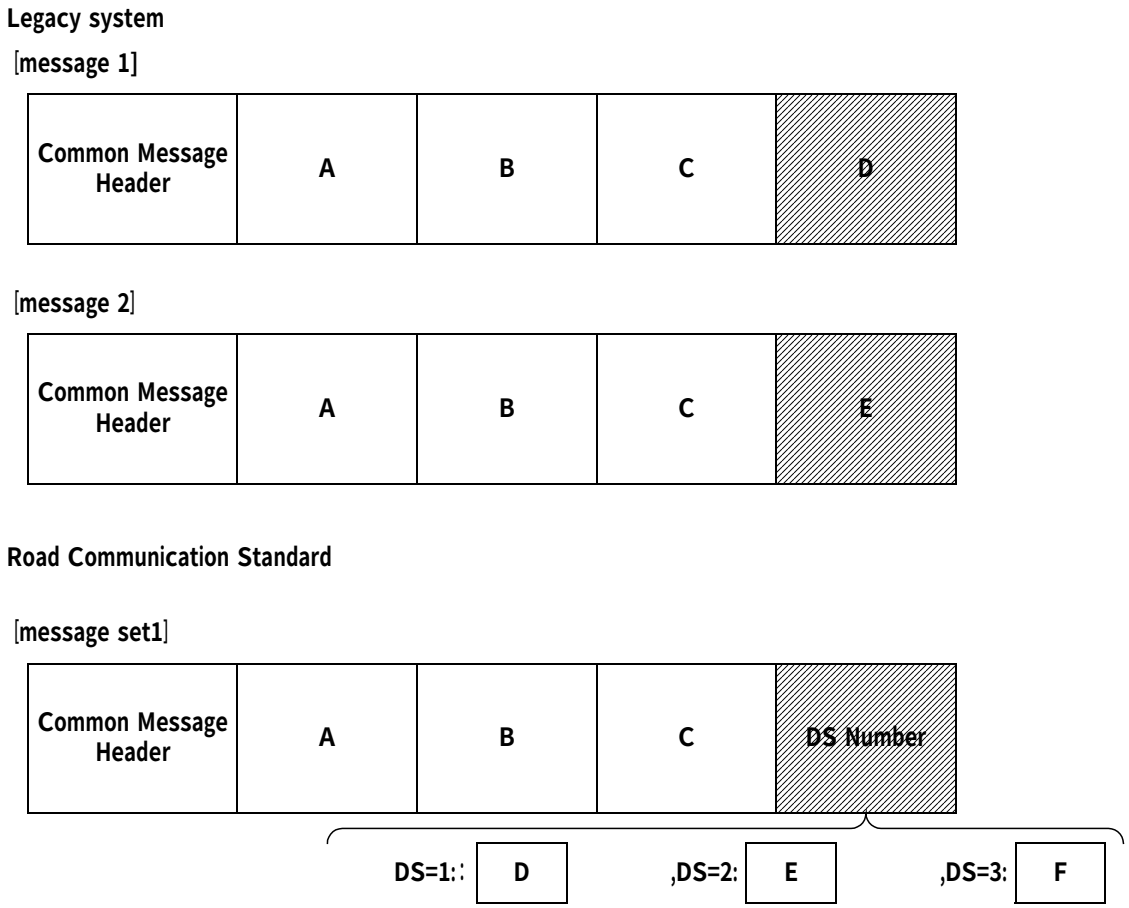


Fig. 2-4 Example of Data Set Application

2.2 Interface Point of Message Set

In this standard, message sets have been designed at the α, β,γ, and δ interface points in the road management systems shown in the figure below.

Interface Point α

Interface Point α is located between the two ITS systems which are managed by different road agencies. Road agencies include regional bureaus of MLIT (Ministry of Land, Infrastructure and Transport), local management offices (i.e. subordinates of regional bureaus), local governments, and highway agencies. As a rule, the local management office is connected to another road agency or to the adjacent office, via the regional bureau. However, in the case of direct communications without a relay by the regional bureau, the same message sets as at interface point α are to be applied.

Interface Point β

Interface Point β is located between the local management office and the device which is positioned outside the office. Specifically, this corresponds to the communications (excepting Internet Protocol) between the center and the roadside equipment (e.g., sensor, message board, and servers at road rest areas), and between the center and the other traffic information provision equipment (e.g., computers at office and home).

If the RCS is applied to roadside-to-roadside communication in future, the same message sets as at interface point β are to be applied.

Interface Point γ

Interface Point γ is located between the inside and outside of the road communications network.

Interface Point δ

Interface Point δ is located between the road communications network and external organizations. Messages exchanged at this interface point are used as fundamental data for judging the traffic forecast or control. If these data are to be provided to the general user, terms and conditions of such an information provision should be settled in advance in each road agency. It is necessary because the road agency has to handle the troubles caused by the provision of the data which have been given by other agencies.

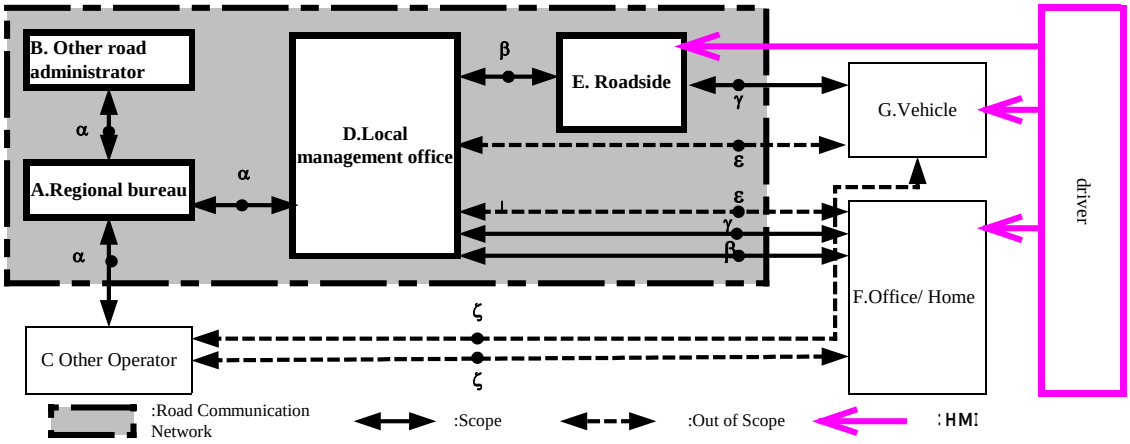


Fig. 2-5 Reference Model and Interface Point

Interface Point ε

Interface Point ε is located between the road communications network and the on-board equipment, and between the center and the server owned by road agencies, where http is used as an application-level protocol. At this interface point, XML or HTML files are used, and message sets are not defined.

Interface Point ζ

Interface Point ζ shows interfaces in which data other than road-related information would be obtained from provider locations other than on-board equipment or roadside. Note that interface point ζ is out of scope.

Treatment of RCS in the highway public corporations

The above descriptions are based on the assumption that this standard is used by the MLIT. Where it is used by other road agencies, system configurations specific to the agencies should be considered.

For the highway public corporations in Japan, the same center conducts the collection and provision of information, as well as the management of information. That is, the center is identical to the integration of A and D in the figure 2-5.

Table 2.2-1 The use of component in each Interface Point

		Component	α	β	γ	δ
(1) Separate ITS system	Collection	Traffic volume data collection		√		
		Environment data collection		√		
		Road Structure Monitoring		√		
		Vehicle weight data collection		√		
		Road event data provision		√		
		Road/Vehicle communications data collection			√	
	Provision	Road event data provision	√	√	√	√
		Road/Vehicle communications data provision			√	
	exchange	Data exchange for between road administrators	√			
		Data exchange for other operators				√
		Data exchange for commercial vehicle operator				√
(2)Device management system		Collection unit		√		
		Provision unit		√		
(3)Image processing system		Data exchange for video image		√		

2.3 Characteristics of Message Sets

2.3.1 Message for Function Allocation

Function Allocation Model

Figure 2-6 describes transitions of function allocations. In the legacy system (a), the judgment function is managed by the center. As the roadside equipment becomes intelligent, a part of the judgment function is moved from the center to the roadside equipment as shown in (b). This message set standard assumes the case (b).

Here, the term judgment function means the function which generates the information such as congestion or disasters, based on a variety of collected data.

The message sets of "Road information signboard" , "CCTV equipment" and "Environmental observation device" in the communication between center and road sides were arranged by references. These message sets assume equipment without the judgement function on the road side. Therefore, that are not a message sets examined for equipment that has the judgement function on the road side that an original road communication standard assumes.

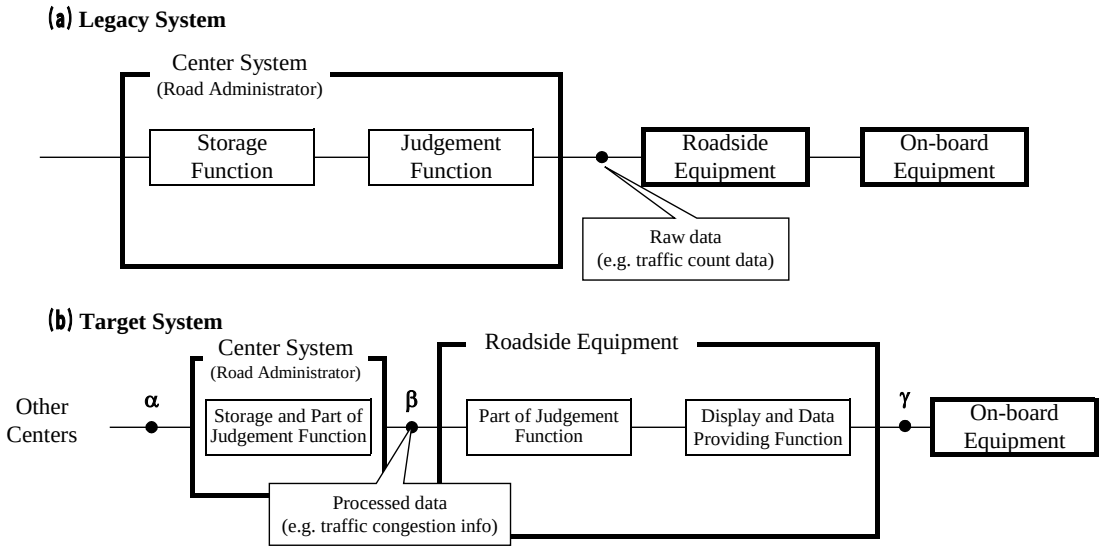


Fig. 2-6 Function Allocation Reference Model

Distribution of Functions to Roadside Equipment

1) Control of roadside equipment by the network

In the legacy system, the intermediate processing functions of the center control all the roadside equipment. As the number of roadside equipment increases, processing load in the center increases. This standard, however, assumes that some of control functions are handled by the network, and then the load on intermediate processing devices of the center is reduced.

2) Intermediate processing functions in the roadside equipment

This message set standard assumes the system configuration in which some of the intermediate processing functions are shifted from the center to the roadside. Then, the capacity of processing functions of a roadside equipment can be smaller than that of the center, and cost reduction of the total system is expected.

3) Enhanced expandability

By distributing intermediate processing functions from the center to the roadside equipment, partial improvements or expansions of the ITS system do not have an effect on the overall network. Thus, the enhancement of expandability can be expected.

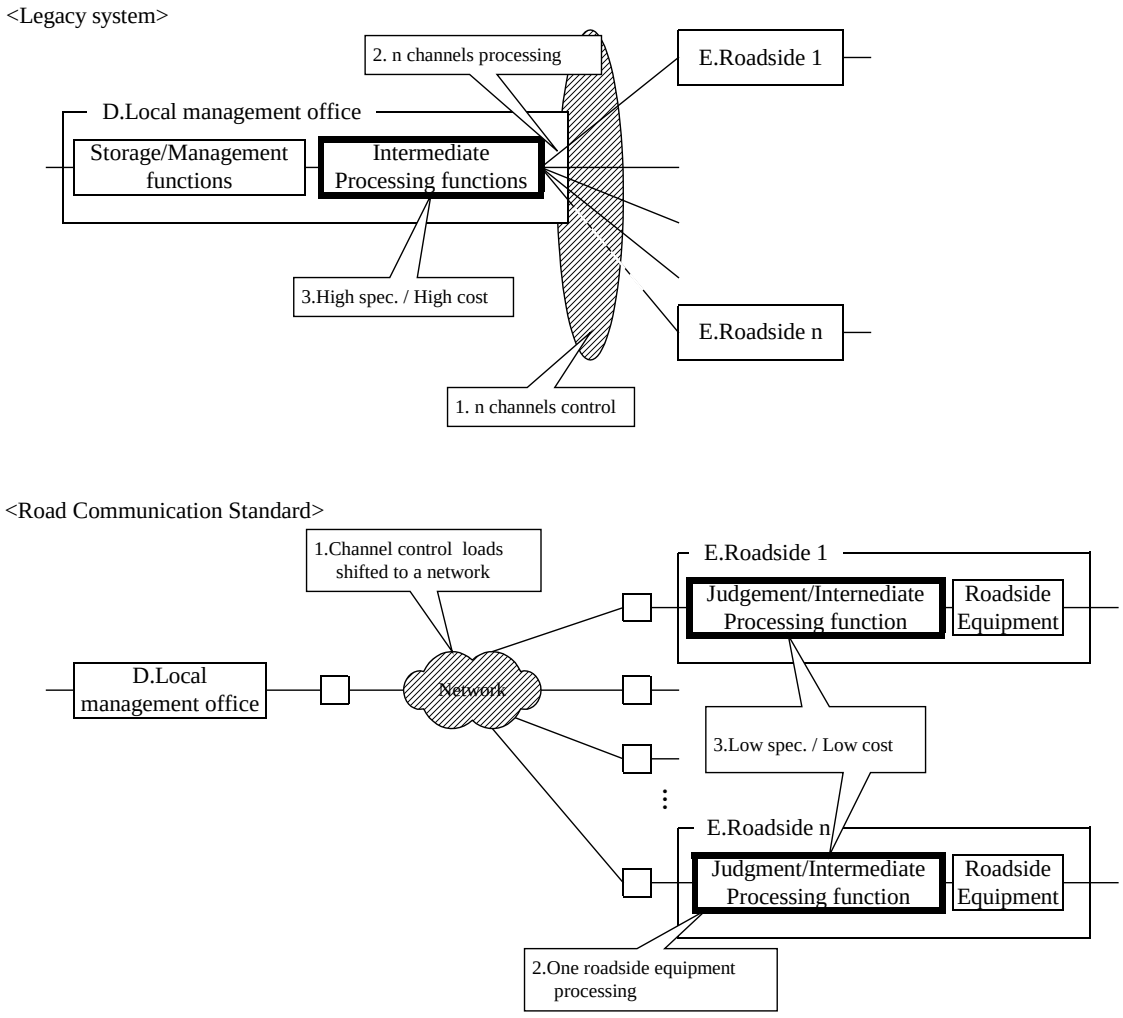


Fig. 2-7　Function Allocation to Roadside equipment

3 Reference

1) Road Communication Standards : Protocol Standard
2) Road Communication Standards : Data Dictionary Standard

4 Definitions of Terms and Abbreviations

4.1 Terms

Item	Term	Definition
1	Component	Collection of functions used in common by large numbers of systems.
2	Data Element	The smallest unit of data that constitutes a message. Individual data elements are defined in a data dictionary.
3	DS: Data Set	The smallest unit of information that constitute a message. Data set is a grouping of data elements.
4	DD: Data Dictionary	Organized and constructed (electronic data base) compilation of descriptions of data elements that provides a consistent means for documenting, storing and retrieving the syntactical form (i.e. representational form) and the meaning and connotation of each data element.
5	Message	Transmission unit for information exchanged between components at an interface point.
6	Message Sequence Diagram	A diagram used to capture the behaviour of several objects within a single use case. This modeling diagram is used to show objects and the messages that flow between these objects in a single use case. Provided as part of the Protocol Standard.
7	MS: Message Set	Collection of messages used at interface points between components. Message sets are composed of the following elements: -Data sets -Data items
8	Message Group	Collection of data sets exchanged between center systems, in which the data sets are divided into individual collections depending on the exchange frequency, so as to increase transmission efficiency.
9	Message Header	Information on identifiers for the message sets exchanged at interface points.
10	PT: Protocol	A formal set of rules governing communications between peers. A protocol may define or negotiate elements such as acknowledgment algorithms, addressing, MTU, etc.

4.2 Abbreviations

(1) APDU: Application Protocol Data Unit
(2) ASDU： Application Service Data Unit
(3) BIT： Bitstring
(4) DD： Data Dictionary
(5) DS： Data Set
(6) ENU： Enumerated
(7) INT： Integer
(8) MS： Message Set
(9) PT： Protocol
(10) SA： System Architecture
(11) SA / PA： Service Area / Parking Area
(12) WWW： World Wide Web
(13) XML： eXtensible Markup Language

5 Common Message Header

5.1 Header Structure

In the message set standard, the common message header is provided for the receive side to interpret the received message sets. The message set ID and the time stamp are treated as mandatory elements because the receiver should know which message set was used and when it was sent.

Table 4.1-1 Common Message Header Elements

	Field Name	Description	Data Type	Required/ Optional	ASN.1 Name
1	Application Identifier (Application ID)	Not necessary with protocols that uniquely identify the applications in a session. Otherwise, applications are identified in this field.	INT	Optional	applicationID
2	Message Set Identifier (Message Set ID)	Message set identifier (at component level).	INT	Required	messageSetID
3	Time Stamp	Date/Time when a message has been sent.	INT	Required	messageTimeStamp
4	Message Set Version	Version info of the message set. Not necessary if a message set version info is transmitted in other ways.	INT	Optional	messageSetVersion
5	Check	Data to verify the integrity of a block of data.	BIT (SIZE (8))	Optional	messageCheck

5.2 Application Identifier

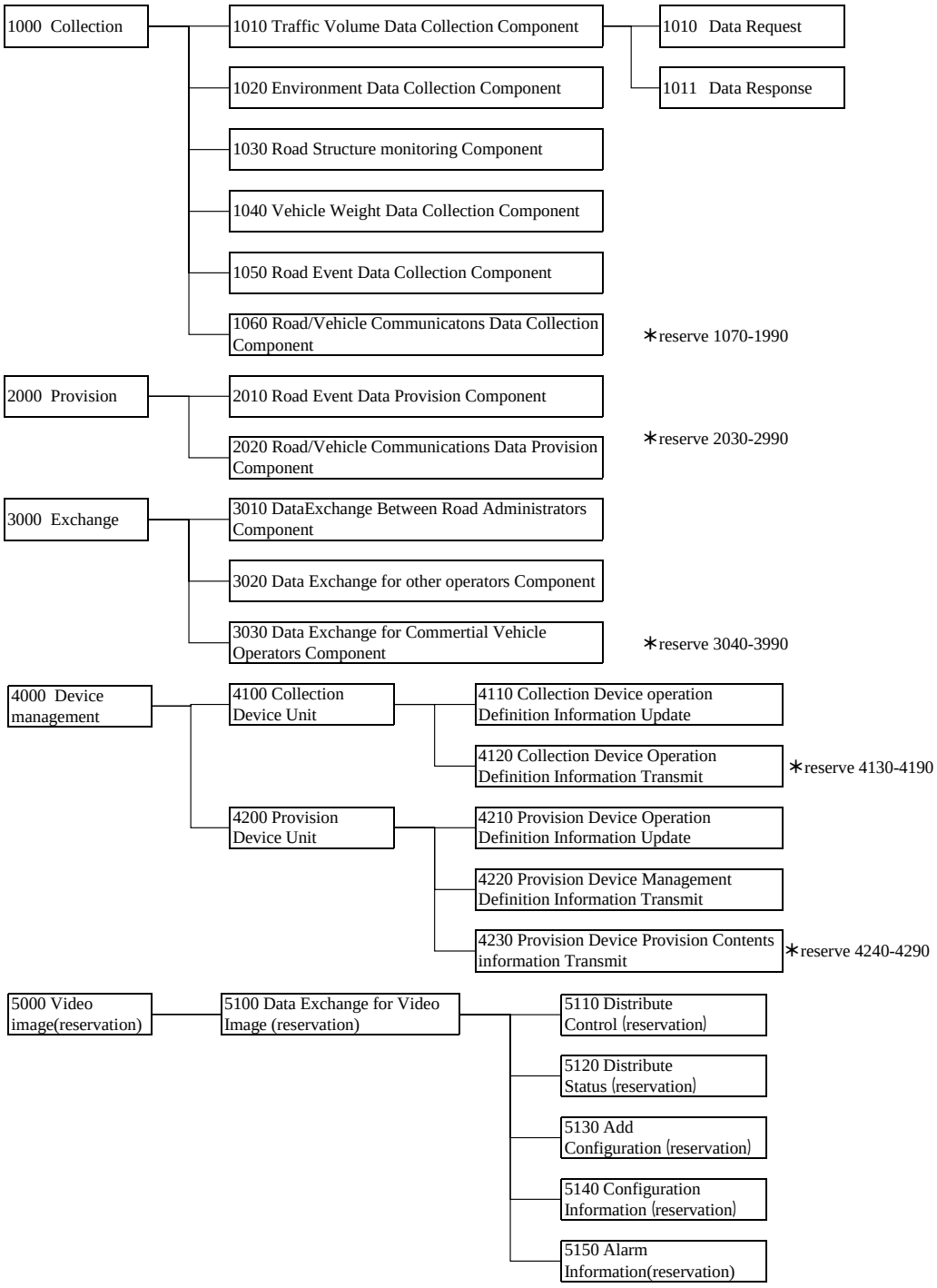
An application Identifier distinguishes the application area represented by the message sets. This ID is not required when it can be registered as application ID with the application interface of L7 in a protocol. Then, it is regarded as optional.

Table 4.1-1 Application Identifier Elements

	Data Name	ASN.1	Note
1	Organization Code	organizationCode	An organization code and an agency code are usually used in the Organization DS.
2	Application Code	messageApplicationId	Application code must be unique in each organization.

5.3 Message Set Identifier

The message set ID is data for identifying the message set defined at the component level. The numbering system is as follows. Four digits are assigned to each message set. For the message sets which will be newly added in future, the number above 5000 will be assigned.



6 Message Sets

6.1 Message Set for Each Component

6.1.1 Message Set for exchanging version information

There is no version negotiation scheme in DATEX-ASN (ISO 14827-2) which our protocol standard adopts as an application layer protocol. Therefore, the message for exchanging version ID has been prepared in this message set standard. This is based on ISO 8823-1 (Connection-oriented presentation protocol: protocol specification). The initiating side provides a list of versions that it is capable of supporting. The responding side also indicates a list of versions that it is capable of supporting. Then, the highest common version number is used in the session.

Table 6.1-1 Message Set for Data Request

	Message	Content	Note
1	commonHeader	MessagesetID(0000)	
2	version	Version ID	

Table 6.1-2 Message Set for Data Response

	Message	Content	Note
1	commonHeader	MessagesetID(0001)	
2	version	VersionID	optional
3	result	Exchange result	optional

6.1.2 Message Set for Traffic Volume Data Collection

The traffic volume data collection component is used to collect the basic data for determining the occurrence of traffic congestion, and for conducting a traffic census.

The role of constituents within the message set is illustrated in the following figure. Three kinds of data sets are defined for traffic volume. These are traffic volume data by vehicle type and by lane within a certain period, passing vehicle data by lane, and AHS (Advanced Cruise-Assist Highway Systems) related data.

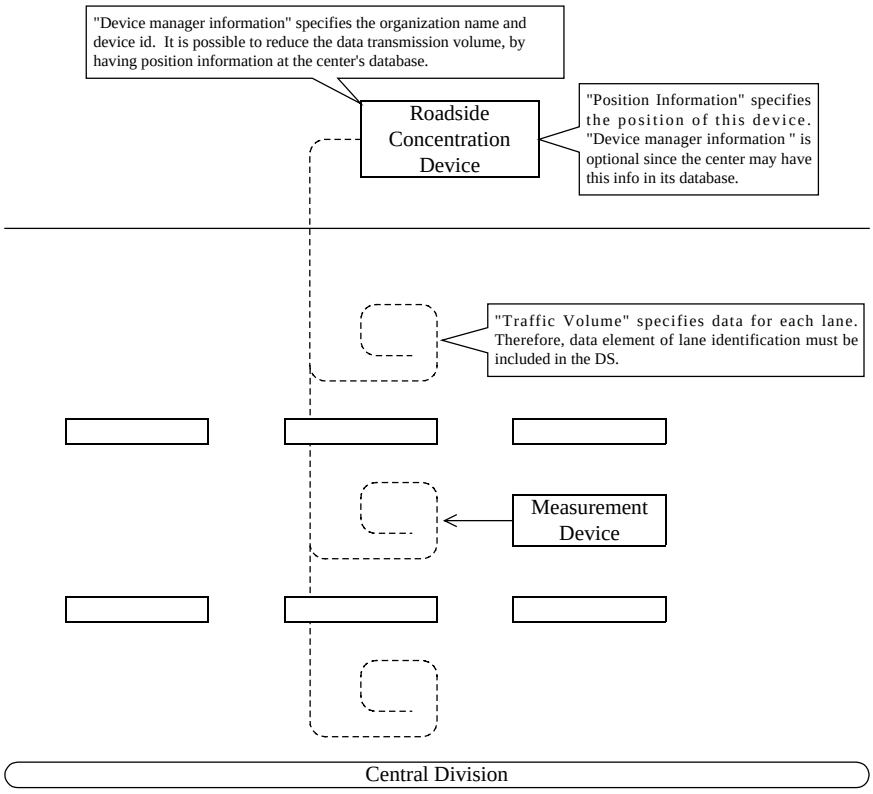


Fig. 6.1-3 Explanation of Message Set

Table 6.1-4 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1010)	
2	requestArea	Request Area	optional
3	requestTerm	Request Term	optional
4	requestDataSetNumber	Request DS # of Traffic Volume Data DS	optional

Table 6.1-5 requestTerm Message

	data element	data type
1	requestStartDateTime	DsDateTime
2	requestEndDateTime	DsDateTime

Table 6.1-6 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetId(1011)	
2	deviceControlAnswerConfirmInfo	Response Status	optional
3	dataConcentrationDeviceMngInfo	Management Information for organizing device	
4	dataConcentrationDeviceLocation	Location of organizing device	optional
5	dataCollectionTerm	Term of collecting data	
6	collectionTrafficVolumeData	Response data of Traffic Volume Data DS	

Table 6.1-7 deviceMngInfo Message

	Data Element	Data Type	Note
1	deviceManagementOrganization	DsOrganization (SEQ)	
2	investDefinitionSystemCode	INT	
3	deviceManagementNumber	UTF8String	
4	deviceSubMagagementNumber	UTF8String	optional

Table 6.1-8 dataCollectTerm Message

	Data Element	Data Type	Note
1	collectionStartDateTime	DsDateTime	
2	collectionEndDateTime	DsDateTime	optional

6.1.3 Message Set for Environmental Data Collection

The environmental data collection component is used to collect the basic data for detecting abnormal environmental conditions at a particular location in the road space, as well as data that is useful in assisting with city planning and road development planning.

The role of constituents within the message set is illustrated in the following figure. The environmental monitoring data DS is used to collect data from the relevant devices.

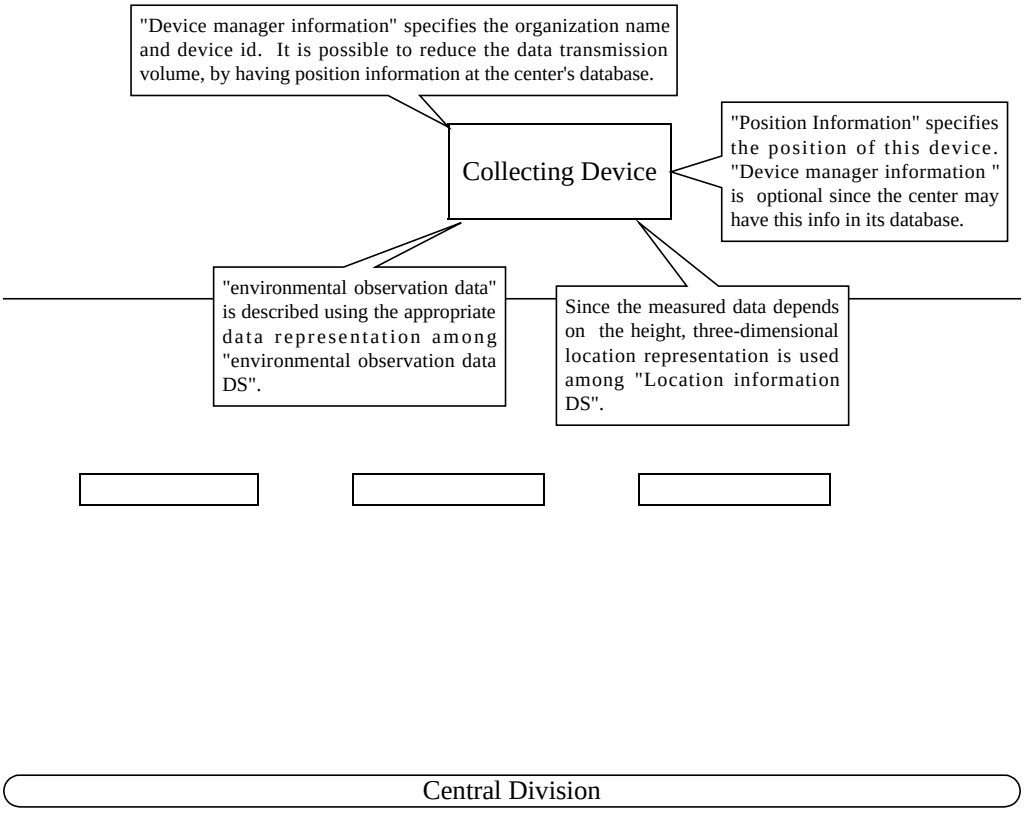


Fig. 6.1-9 Explanation of Message Set

Table 6.1-10 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1020)	
2	requestArea	Request Area	optional
3	requestTerm	Request Term	optional
4	requestDataSetNumber	Request DS # of Environment Monitoring Data DS	optional

Table 6.1-11 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(1021)	
2	deviceControlAnswerConfirmInfo	Response Status	optional
3	dataCollectionDeviceMngInfo	Management Information for organizing device	
4	dataCollectionDeviceLocation	Location of organizing device	optional
5	dataCollectionTerm	Term of collecting data	
6	collectionEnvironmentMonitoringData	Response data of Environment Monitoring Data DS	

6.1.4 Message Set for Road Structure Monitoring

The road structure monitoring component is used to collect basic data for detecting abnormalities in road structures.

The role of constituents within the message set is illustrated in the following figure. Appropriate data set for the related device is selected among the road structure monitoring data DS.

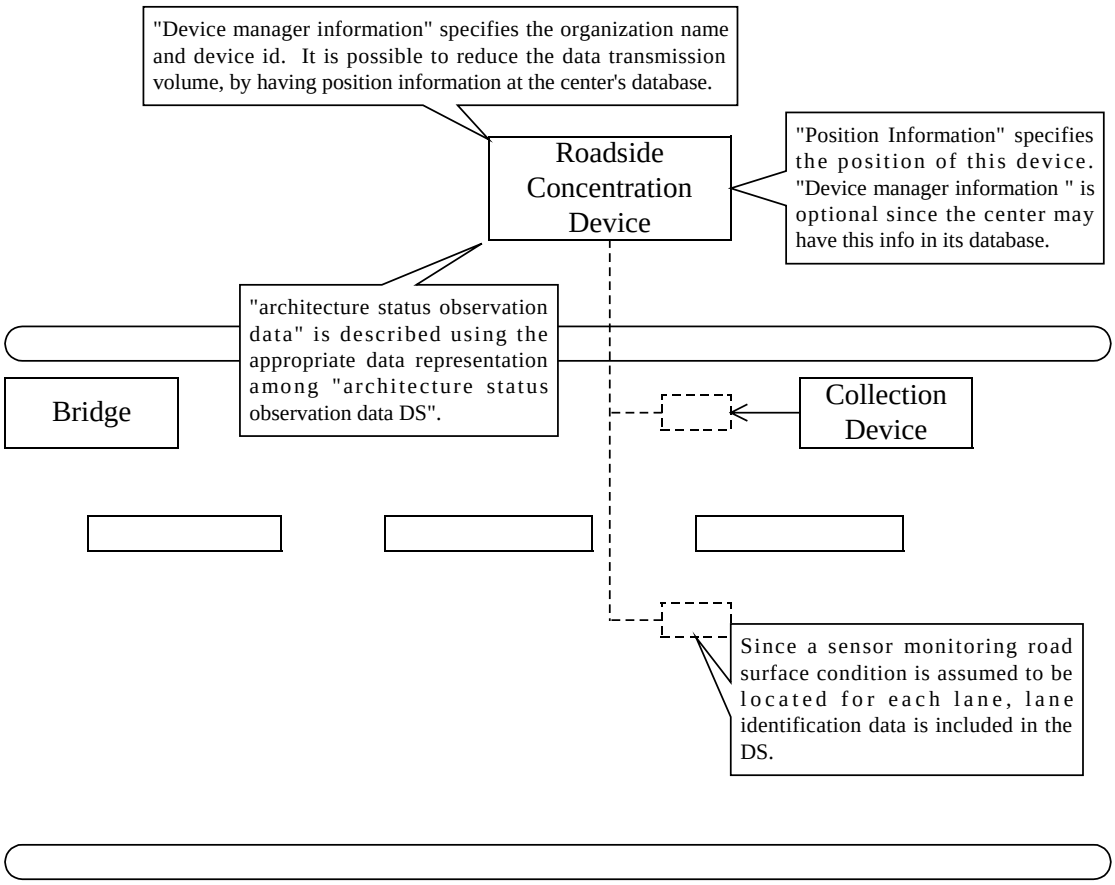


Fig. 6.1-12 Explanation of Message Set

(Example of Road Surface Condition Monitoring)

Table 6.1-13 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1030)	
2	requestArea	Request Area	optional
3	requestTerm	Request Term	optional
4	requestDataSetNumber	Request DS # of Road Structure Monitoring Data DS	optional

Table 6.1-14 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(1031)	
2	deviceControlAnswer ConfirmInfo	Response Status	optional
3	dataConcentrationDe viceMngInfo	Management Information for organizing device	
4	dataConcentrationDe viceLocation	Location of organizing device	optional
5	dataCollectTerm	Term of collecting data	
6	collectionRoadStruct ureMonitoringData	Response data of Road Structure Monitoring Data DS	

6.1.5 Message Set for Vehicle Weight Data Collection

The vehicle weight data collection component is a communication function of sending the vehicle weight information, which of passing vehicle is measured at roadside, to Center systems.

The role of constituents within the message set is illustrated below.

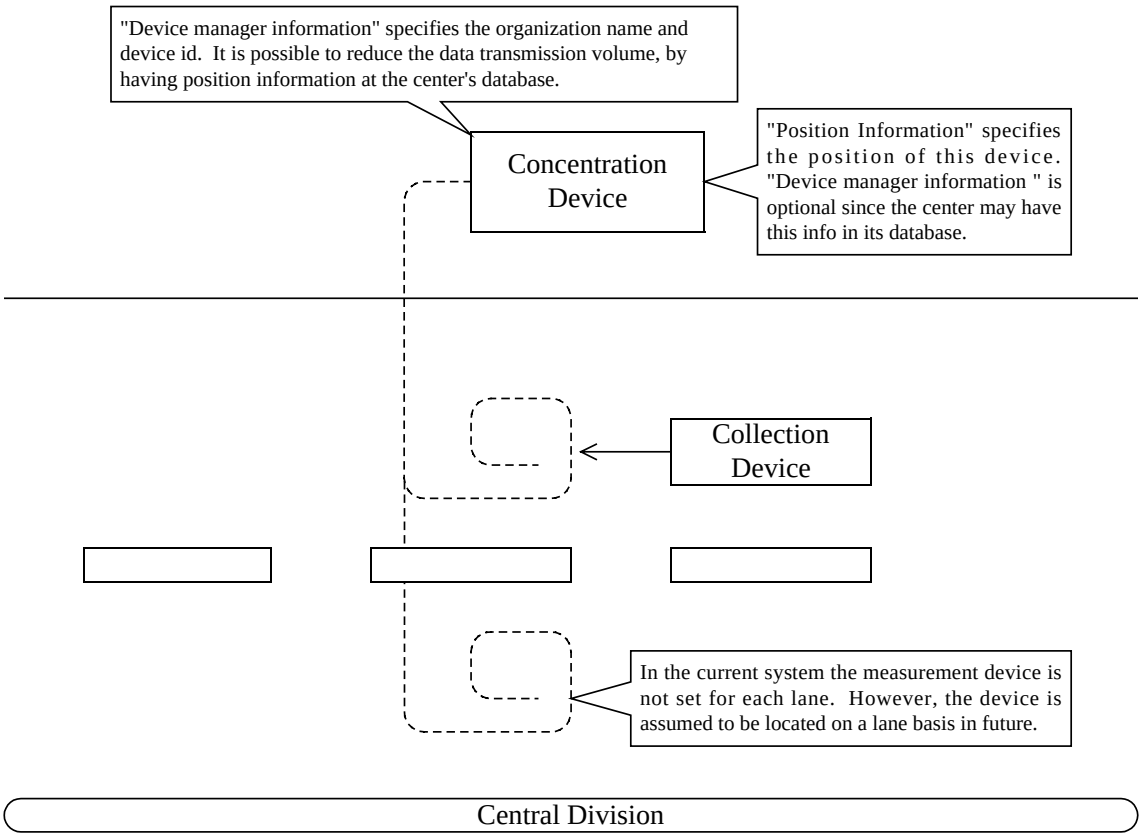


Fig. 6.1-15 Explanation of Message Set

Table 6.1-16 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1040)	
2	requestArea	Request Area	optional
3	requestTerm	Request Term	optional

Table 6.1-17 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(1041)	
2	deviceControlAnswerConfirmInfo	Response Status	optional
3	dataConcentrationDeviceMngInfo	Management Information for organizing device	
4	dataConcentrationDeviceLocation	Location of organizing device	optional
5	dataCollectionTerm	Term of collecting data	
6	collectionWeightData	Response data	Collect Weight Data Message

6.1.6 Message Set for Road Event Data Collection

The road event data collection component is used to collect road and traffic information from roadside. Specifically, this component is applied (1) when the road and traffic information generated at the roadside equipment based on the data collected there is sent, and (2) when the road and traffic information generated at the on-board equipment based on the vehicle sensor data is sent to the center via the roadside equipment.

In the data request, the data set numbers which the request side needs to obtain are specified among the road and traffic information data sets, and in the data response, a chain of requested data sets are sent to the request side.

In such a data set, a data element of information provider name is included. It indicates who has provided the event data. For example, it can be recognized whether the event data came from the ordinary vehicle or from the data collection device which a road agency owns. By this information, it can be judged whether this data can be made open to the public or not.

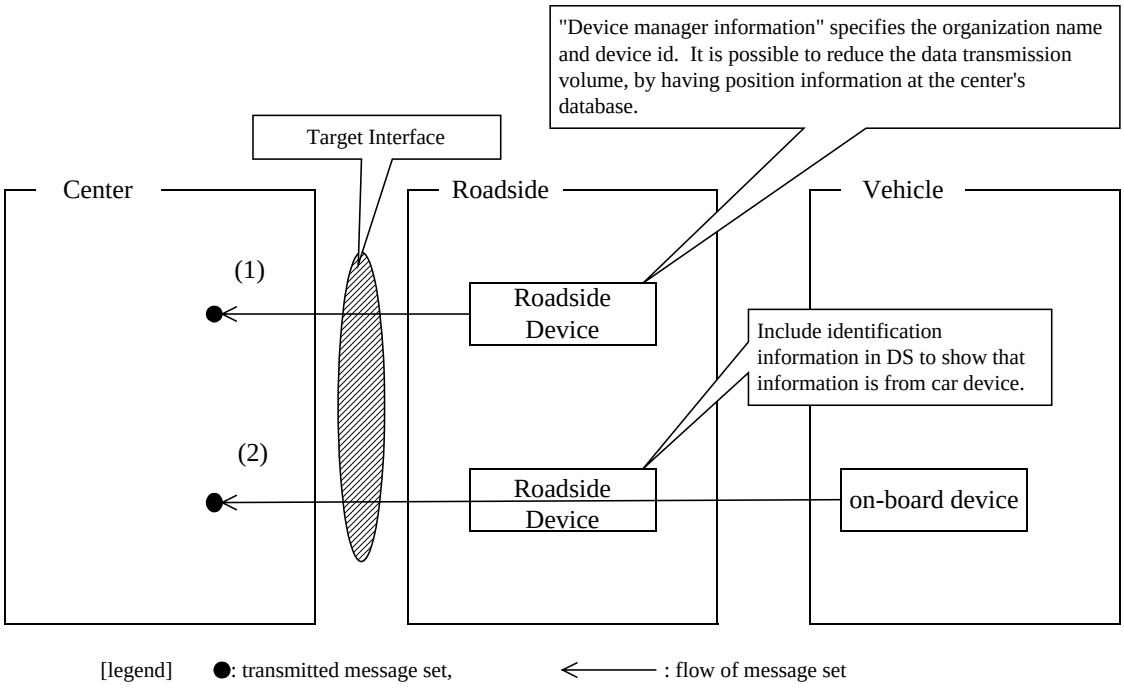


Fig. 6.1-18 Explanation of Message Set

Table 6.1-19 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1050)	
2	requestArea	Request Area	optional
3	requestTerm	Request Term	optional
4	requestDataSetList	Request DS # List of Road Event Information DS	optional

Table 6.1-20 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(1051)	
2	deviceControlAnswer ConfirmInfo	Response Status	Optional
3	dataConcentrationDe viceMngInfo	Management Information for organizing device	
4	dataConcentrationDe viceLocation	Location of organizing device	Optional
5	collectionRoadEventD ata	Sequence of response data of Road Event Information DS	

6.1.7 Message Set for Roadside-to-Vehicle Data Collection

The roadside-to-vehicle data collection component is used to issue requests from roadside to on-board equipment, and to obtain data from the on-board equipment.

The collector information is included in the data transmitted from the on-board equipment to the roadside equipment so that the vehicle (e.g. specially permitted vehicle or road agency's patrol vehicle) which collected the data can be identified.

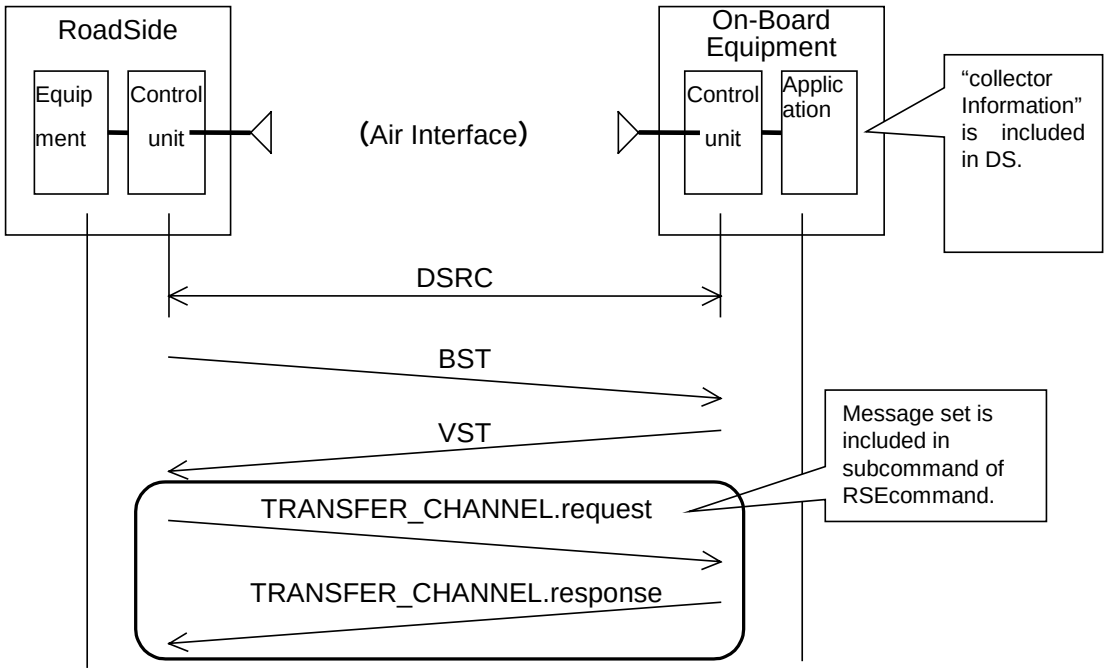


Fig. 6.1-21 Explanation of Message Set

Table 6.1-212 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(1060)	optional
2	deviceSecurityInfo	Security Information	optional
3	requestArea	Request Area	optional
4	requestTerm	Request Term	optional
5	requestDataSetList	Request DS # List of Road Event Information DS	

Table 6.1-223 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(1061)	optional
2	deviceSecurityInfo	Security Information	optional
3	deviceControlAnswerConfirmInfo	Response Status	optional
4	collectionRoadEventData	Sequence of response data of Road Event Information DS	

6.1.8 Message Set for Road Event Data Provision

The road event data provision component is used to provide event information to the public. It is applied when the event data is sent to the message board or to the data server located at the service areas or parking areas of highways.

In the data request, the data set numbers which the request side needs to obtain are specified among the data sets for road event data provision, and in the data response, a chain of requested data sets are sent to the request side.

Provider information is included in the data set so that it can be identified who sent the information.

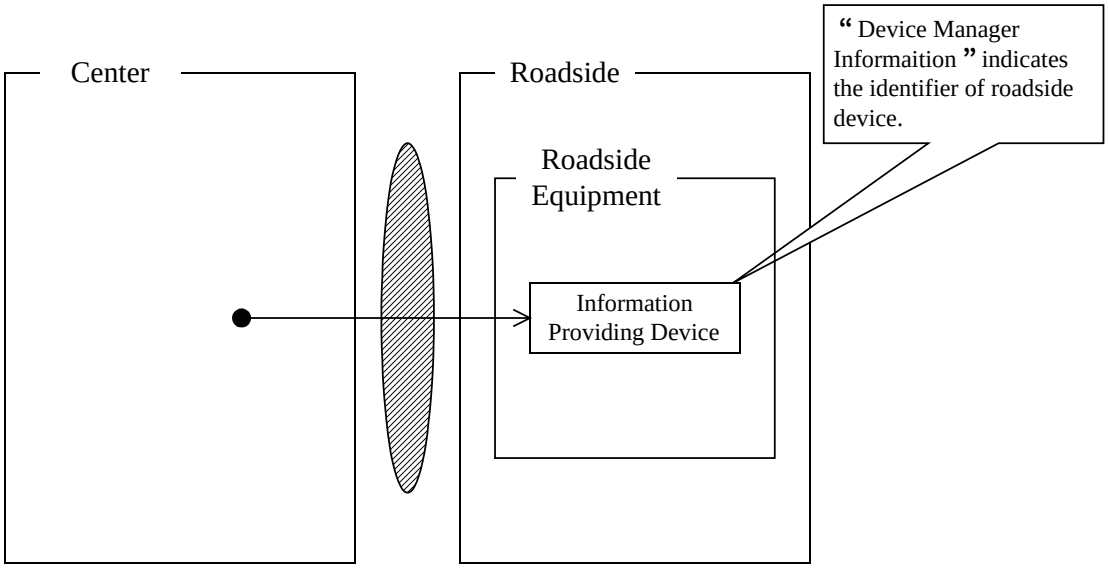


Fig. 6.1-24 Explanation of Message Set

Table 6.1-235 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(2010)	
2	deviceMngInfo	Management Information for organizing device	
3	requestArea	Request Area	optional
4	requestTerm	Request Term	optional
5	requestDataSetList	Request DS # List of Road Event	

Table 6.1-246 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(2011)	
2	deviceControlAnswerConfirmInfo	Response Status	optional
3	provisionDeviceMngInfo	Management Information for organizing device	optional
4	provisionRoadEventData	Sequence of response data of Road Event Information DS	

6.1.9 Message Set for Roadside-to-Vehicle Data Provision

The roadside-to-vehicle data provision component is used to issue requests from on-board equipment to roadside equipment, and to obtain data from the roadside.

The provider information is included in the data transmitted from the roadside equipment to the on-board equipment so that the information provider's name can be identified.

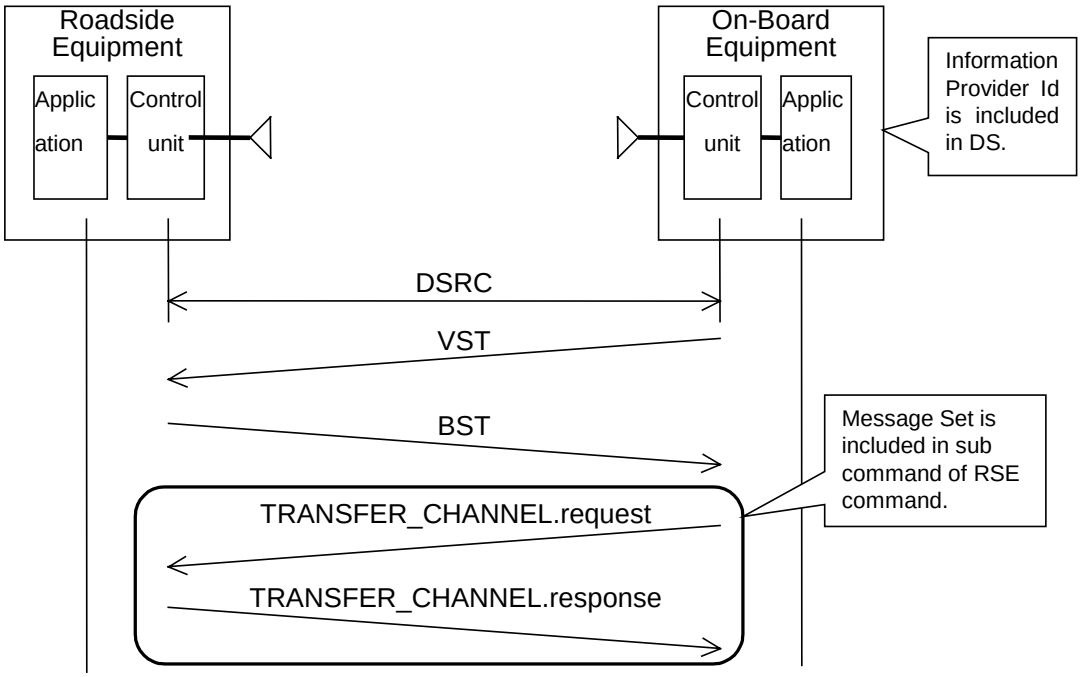


Fig. 6.1-27 Explanation of Message Set

Table 6.1-258 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(2020)	optional
2	deviceSecurityInfo	Security Information	optional
3	messageUserInfo	User Identifier	optional
4	requestArea	Request Area	optional
5	requestTerm	Request Term	optional
6	requestDataSetList	Request DS # List of Road Event Information DS	

Table 6.1-269 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(2021)	optional
2	deviceSecurityInfo	Security Information	optional
3	deviceControlAnswerConfirmInfo	Response Status	optional
4	provisionRoadEventData	Sequence of response data of Road Event Information DS	

6.1.10 Message Set for Data Exchange Between Road Administrators

The component of data exchange between road administrators is used to exchange data which are owned at each other's sides.

It is assumed that, (i)in the case that the data size is very large and the transmission frequency is not so high, a database file is sent instead of this message set, and (ii)otherwise, the message set is used. In using this message set, data can be sent in sequences of data sets.

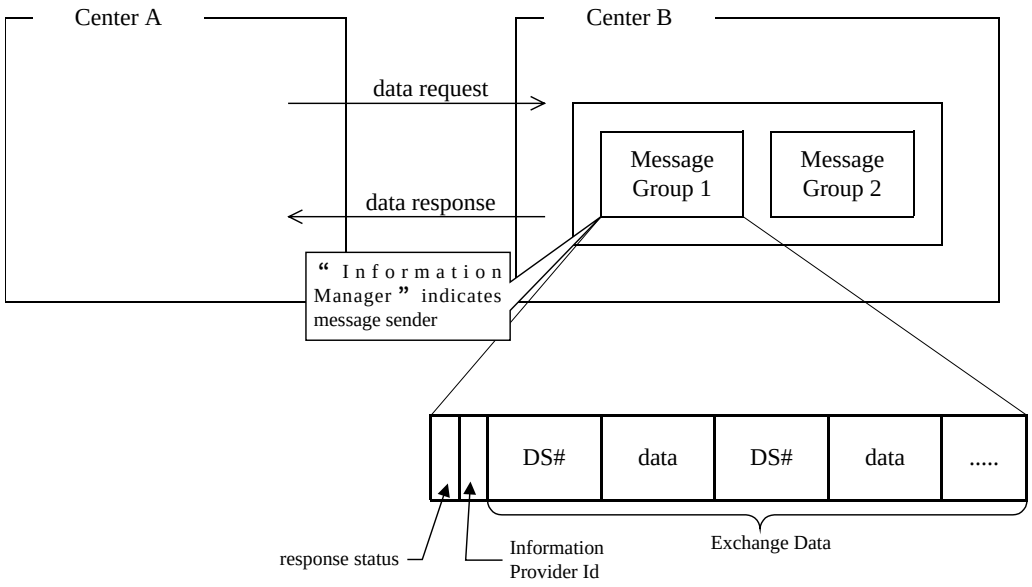


Fig. 6.1-30 Explanation of Message Set

Table 6.1-31 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(3010)	
2	messageUserInfo	User Identifier	
3	requestArea	Request Area	optional
4	requestTerm	Request Term	optional
5	requestDataSetList	Request DS # List	optional

Table 6.1-32 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(3011)	
2	deviceControlAnswerConfirmInfo	Response Status	optional
3	dataMngInfo	Data provider Information	
4	responseDataSet	Sequence of response data	

Table 6.1-33 dataMngInfo Message

	Data Element	Data Type	Note
1	informationManagementOrganization	DsOrganization (SEQ)	optional
2	investDefinitionSystemCode	INT	optional
3	deviceInformationManagementNumber	INT	

6.1.11 Message Set for Data Exchange with other operators

The data exchange for other operators component is used to exchange road and traffic information between road administrators and other external organizations.

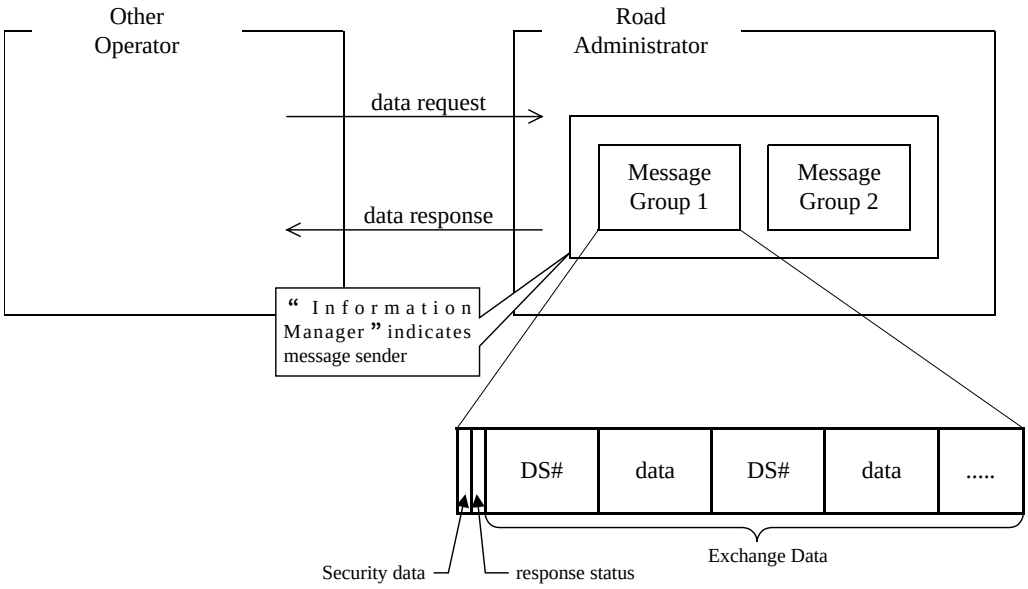


Fig. 6.1-34 Explanation of Message Set

Table 6.1-35 Message Set for Data Request

	Constituent	Content	Note
1	commonHeader	MessagesetID(3020)	
2	deviceSecurityInfo	Security Information	
3	messageUserInfo	User Identifier	
4	requestArea	Request Area	optional
5	requestTerm	Request Term	optional
6	requestDataSetList	Request DS # List	

Table 6.1-36 Message Set for Data Response

	Constituent	Content	Note
1	commonHeader	MessagesetID(3021)	
2	deviceSecurityInfo	Security Information	
3	deviceControlAnswerConfirmInfo	Response Status	optional
4	responseDataSet	Sequence of response data	

6.1.12 Message Set for Data Exchange with Commercial Vehicle Center Operators

The component of data exchange between road agencies and commercial vehicle operators is used to exchange information on the applications and approvals for specially permitted vehicles, the road conditions, and so on.

Since the commercial vehicle operator accesses data from the outside of the road communications network, the message set may include security information.

The application form is sent from commercial vehicle centers to road agencies, and the approval form is from road agencies to commercial vehicle centers. Since the approval form is not issued instantly after the submission of application form, this component is not both-way but one-way communication. Then, this message set is of event-driven type.

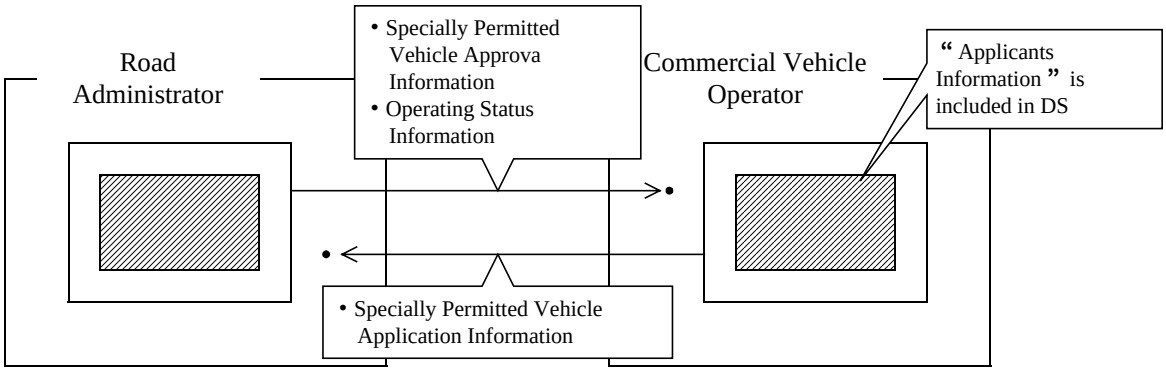


Fig. 6.1-37 Explanation of Message Set

Table 6.1-38 Message Set for sending data

	Constituent	Content	Note
1	commonHeader	MessagesetID(3031)	
2	deviceSecurityInfo	Security Information	optional
3	commercialVehicleOperatorsData	Response data of Commercial Vehicle Operator Information DS	

6.2 Data Sets

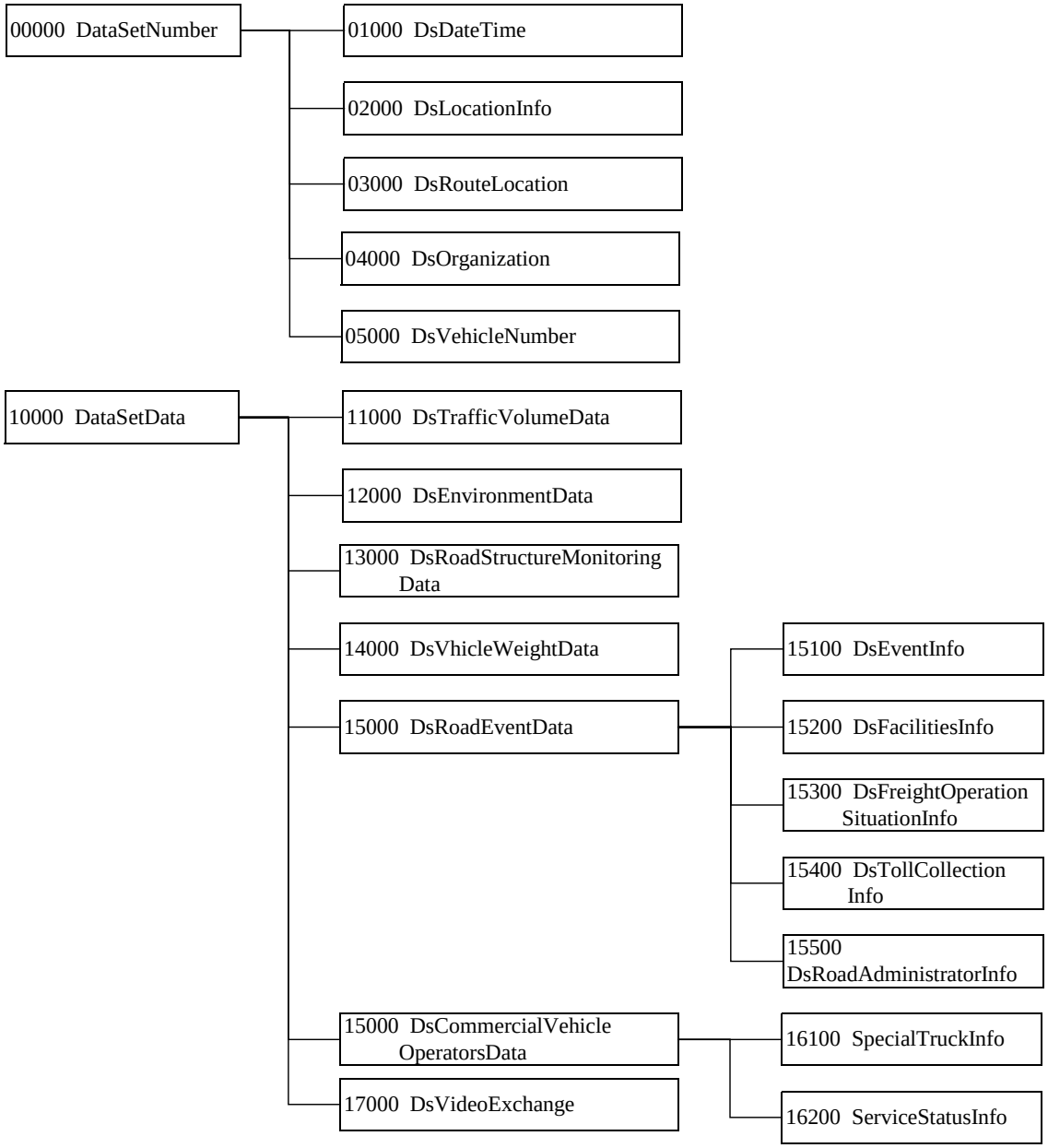


Fig. 6.2-1 Data Sets

6.2.1 Date and Time DS

Table 6.2-1 Date and Time DS

NAME		TYPE
Data Set	Data	
01000 dsDateTime	datesDateOfYear	DatesDateOfYear(Op)
	datesUnitOfTime	DatesUnitOfTime(Op)
	datesDayOfTheWeek	ENUMERATED(Op)
	relationLinkageIdentifier	INTEGER(Op)

6.2.2 Location Information DS

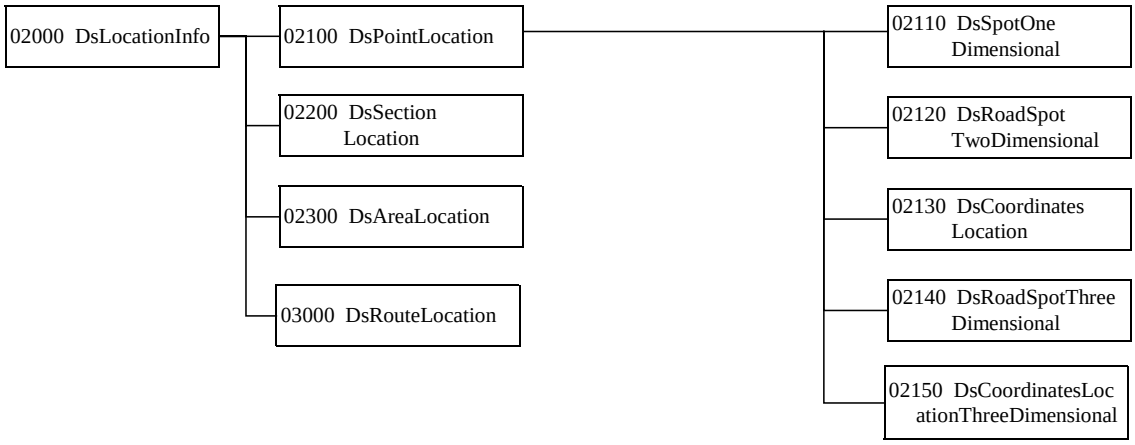


Fig. 6.2-2 Location Information DS

Table 6.2-2 Location Information DS

NAME				TYPE
Data Set			Data	
02100 dsPoint Location	02110 dsLocationRoadPosition	02111 dsExpresswaySpotKp	roadTollRoadRouteCode	UTF8String
			roadTollRoadRouteSubCode	ENUMERATED(Op)
			roadOrganization	DsOrganization
			roadRouteName	UTF8String(Op)
			roadRoadType	ENUMERATED
			roadRouteDirectionCode	ENUMERATED
			locationTollRoadSpotKp	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
		02112 dsNationalRoadSpotKp	roadNationalwayRouteNumber	INTEGER
			roadNationalwayRouteSubCode	INTEGER(Op)
			roadRouteName	UTF8String(Op)
			roadRoadType	ENUMERATED
			roadRouteDirectionCode	ENUMERATED
			locationNationalRoadSpotKp	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
			roadGeneralNationalHighwayClassification	ENUMERATED(Op)
		02113 dsRoadLinkSpot	dsRoadLinkSection	DsRoadLinkSection(SEQUENCE)
			locationLinkEndDistance	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
		02114 dsLocationCode	roadIcCode	UTF8String
			facilitySapaAreaLocationNumber	UTF8String
			roadIntersectionNumber	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
		02115 dsBaseRoadLinkSpot	dsBaseRoadLinkSection	DsBaseRoadLinkSection (SEQUENCE)
			locationLinkEndDistance	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
		02116 dsLocalRoadSpotKp	roadNationalwayRouteNumber	INTEGER(Op)
			roadNationalwayRouteSubCode	INTEGER(Op)
			roadOrganization	DsOrganization(Op)
			roadRouteName	UTF8String(Op)
			roadRoadType	ENUMERATED
			roadRouteDirectionCode	ENUMARATED
			locationLocalRoadSpotKp	INTEGER(Op)
			relationLinkageIdentifier	INTEGER(Op)
			roadGeneralNationalHighwayClassification	ENUMARATED(Op)
	02120 dsRoadSpotTwoDimensional		dsLocationRoadPosition	DsLocationRoadPosition (CHOICE)
			roadLaneType	ENUMERATED
			locationOffset	INTEGER
			relationLinkageIdentifier	INTEGER(Op)

NAME				TYPE
Data Set			Data	
(02100)	02130 dsCoordinatesLocation	02131 dsNormalCoordinates	locationXYNormalCoordinates	LocationXYNormalCoordinates
			relationLinkageIdentifier	INTEGER(Op)
		02132 dsLatitudeLongitude	locationLatitudeDegree	INTEGER
			locationLatitudeMinute	INTEGER
			locationLatitudeSecond	INTEGER
			locationLongitudeDegree	INTEGER
			locationLongitudeMinute	INTEGER
			locationLongitudeSecond	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
			locationLatitudeLongitudeDegree	LocationLatitudeLongitudeDegree
	02140 dsRoadSpotThreeDimensional		dsRoadSpotTwoDimensional	DsRoadSpotTwoDeminsional(SEQUENCE)
			locationAltitudeGround	INTEGER
			locationAltitudeSea	INTEGER
			relationLinkageIdentifier	INTEGET(Op)
	02150 dsCoordinatesLocationThreeDimensional		dsCoordinatesLocation	DsCoordinatesLocation(CHOICE)
			locationAltitudeGround	INTEGER
			locationAltitudeSea	INTEGER
			relationLinkageIdentifier	INTEGER(Op)
	02160 dsAddress		locationMunicipalityCode	INTEGER
			locationHouseNumber	UTF8String
			relationLinkageIdentifier	INTEGER(Op)

NAME			TYPE
Data Set		Data	
02200 dsSection Location	02201 dsStartEndSection	startPointLocation	DsPointLocation(CHOICE)
		endPointLocation	DsPointLocation(CHOICE)
		relationLinkageIdentifier	INTEGER(Op)
	02202 dsRoadLinkSection	locationSecondaryCoordinatesCode	INTEGER
		locationLinkLayer	ENUMERATED
		locationLinkEntry	ENUMERATED
		locationLinkNumber	INTEGER
		locationLinkVersion	INT(Op)
		relationLinkageIdentifier	INTEGER(Op)
	02203 dsSpanSection	locationSpanCode	INTEGER
		locationCourseDistance	INTEGER
		locationSpanDistance	INTEGER
		roadRouteName	UTF8String
		relationLinkageIdentifier	INTEGER(Op)
	02204 dsBaseRoadLinkSection	roadRouteDirectionCode	ENUMERATED
		locationSecondaryCoordinatesCode	INTEGER
		locationBaseRoadLinkNumber	INTEGER
		locationBaseRoadLinkVersion	INT(Op)
		relationLinkageIdentifier	INTEGER(Op)
02300 dsAreaLocation	02301 locationRegionCode0	locationRegionCode	ENUMERATED
		relationLinkageIdentifier	INTEGER(Op)
	02302 locationLifeAreaCode0	locationLifeAreaCode	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	02303 locationMunicipalityCode0	locationMunicipalityCode	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	02304 locationBZoneCode0	locationBZoneCode	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	02305 locationCZoneCode0	locationCZoneCode	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	02306 locationWideArea0	locationWideArea	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
03000 dsRouteLocation		dsRoutelocation	DsRouteLocation(CHOICE)

6.2.3 Route Location Information DS

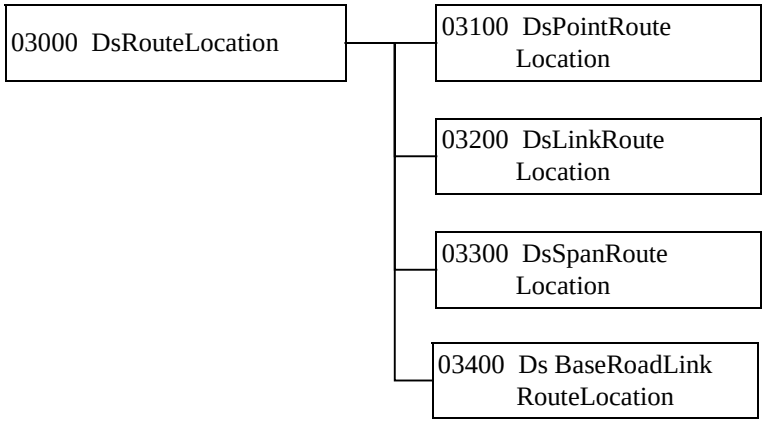


Fig. 6.2-3 Route Location Information DS

Table 6.2-3 Route Location Information DS

NAME			TYPE
Data Set		Data	
03000 dsRouteLocation	03100 dsPointRouteLocation	startPointLocation	DsPointLocation
		detourPointLocation	DsPointLocation
		endPointLocation	DsPointLocation
		routeCourseDistance	INTEGER
		routeNumber	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	03200 dsLinkRouteLocation	startLink	DsRoadLinkSection
		endLink	DsRoadLinkSection
		detourLink	DsRoadLinkSection
		routeLinkFigure	INTEGER
		locationCourseDistance	INTEGER
		relationLinkageIdentifier	INTEGER(Op)
	03300 dsSpanRouteLocation	startSpan	DsRoadLinkSection
		endSpan	DsRoadLinkSection
		locationSpanDistance	INTEGER
		locationSpanNumber	OCTET STRING
		relationLinkageIdentifier	INTEGER(Op)
	03400 dsBaseRoadLink RouteLocation	startBaseRoadLink	DsBaseRoadLinkSection
		EndBaseRoadLink	DsBaseRoadLinkSection
		detourBaseRoadLink	DsBaseRoadLinkSection
		routeLinkFigure	INTEGER
		locationCourseDistance	INTEGER
		relationLinkageIdentifier	INTEGER(Op)

6.2.4 Organization DS

Table 6.2-4 Organization DS

NAME		TYPE
Data Set	Data	
04000 dsOrganization	organizationAgencyCode	ENUMERATED(Op)
	organizationAgencyName	UTF8String(Op)
	organizationOrganizationCode	UTF8String(Op)
	organizationOrganizationName	UTF8String(Op)
	organizationBureauCode	UTF8String(Op)
	organizationBureauName	UTF8String(Op)
	organizationDivisionCode	INTEGER(Op)
	organizationDivisionName	UTF8String(Op)
	organizationPersonName	UTF8String(Op)
	organizationAddress	UTF8String(Op)
	organizationTelNum	UTF8String(Op)
	organizationFaxNum	UTF8String(Op)
	organizationEmail	OCTET STRING(Op)
	organizationVehicleBureauCode	ENUMERATED(Op)
	organizationWeatherOrganizationCode	ENUMERATED(Op)
	locationRegionCode	ENUMERATED(Op)
	locationMunicipalityCode	INTEGER(Op)
	organizationRoadAdministrationInsideIdentifierNum	INTEGER(Op)
	relationLinkageIdentifier	INTEGER(Op)

6.2.5 Vehicle Number DS

Table 6.2-5 Vehicle Number DS

NAME		TYPE
Data Set	Data	
05001 dsVehicleNumber	organizationVehicleBureauCode	ENUMERATED
	movableBodyCategoryNumber	INTEGER
	movableBodyKanaCharacter	UTF8String
	movableBodySequentialNumber	INTEGER
	relationLinkageIdentifier	INTEGER(Op)

6.2.6 Traffic Volume Data DS

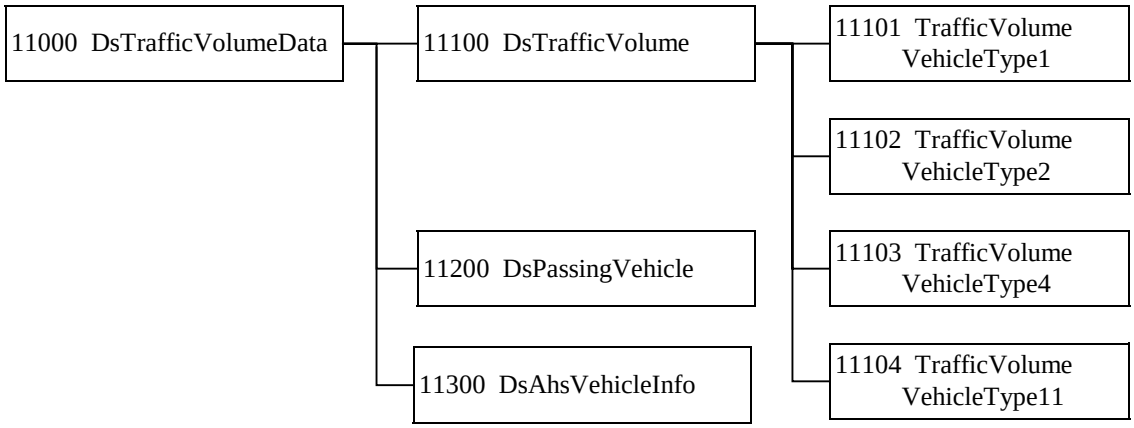


Fig. 6.2-4 Traffic Volume Data DS

Table 6.2-6 Traffic Volume Data DS

NAME		TYPE
Data Set	Data	
11100 dsTraffic Volume	deviceCollectedDataStatus	ENUMERATED(Op)
	deviceLocation	DsLocationInfo
	calculationTotalTrafficVolume	CalculationTrafficVolume
	typeTrafficVolume	DsTypeTrafficVolume
	calculationSpotAverageSpeed	INTEGER
	calculationOccupancy	INTEGER
	deviceStatus	ENUMERATED(Op)
	relationLinkageIdentifier	INTEGER(Op)
	dateTime	DsDateTime(SET)
	confirmTotalTrafficVolume	DeviceControlAnswerConfirmInfo(Op)
	confirmSpotAverageSpeed	DeviceControlAnswerConfirmInfo(Op)
	confirmOccupancy	DeviceControlAnswerConfirmInfo(Op)
11200 dsPassingVehicle	calculationPassingVehicleIndex	INTEGER
	deviceLocation	DsLocationInfo
	dateTime	DsDateTime
	eventDeterminationOfVehicleHeight	ENUMERATED(Op)
	calculationVehicleHeightMeasurementResult	INTEGER(Op)
	eventDeterminationOfVehicleLength	ENUMERATED(Op)
	calculationVehicleLengthMeasurementResult	INTEGER(Op)
	eventDeterminationOfVehicleLengthBetweenGroundAndBottom	ENUMERATED(Op)
	calculationVehiclewidthMeasurementResult	INTEGER(Op)
	calculationVehicleForm	BIT String(Op)
	calculationVehicleNetWeightMeasurementResult	INTEGER(Op)
	calculationAxisNumberMeasurementResult	INTEGER(Op)
	vehicleNumber	DsVehicleNumber(Op)
	calculationCarTypeMeasurementResult	ENUMERATED
	calculationSpotSpeed	INTEGER(Op)
	calculationRelativeSpeed	INTEGER(Op)
	calculationVehicularGap	INTEGER(Op)
	deviceStatus	ENUMERATED(Op)
	relationLinkageIdentifier	INTEGER(Op)
11300 dsAhsVehicleInfo	vehicleLocation	DsLocationInfo
	dateTime	DsDateTime
	calculationAhsVehicleSpeed	INTEGER
	calculationAhsLocationTrafficLanes	INTEGER

NAME		TYPE
Data Set	Data	
	calculationAhsVehicleLength	INTEGER
	calculationAhsVehicleType	ENUMERATED
	calculationVehicleIDMeasurementResult	INTEGER
	relationLinkageIdentifier	INTEGER(Op)

Table 6.2-7 Traffic Volume of Vehicle Type

NAME		TYPE
Data Set	Data	
11101 trafficVolumeVehicleType1	trafficVolumeVehicleType1	CalculationTrafficVolume
	relationLinkageIdentifier	INTEGER(Op)
11102 trafficVolumeVehicleType2	trafficVolumeVehicleType2Large	CalculationTrafficVolume
	trafficVolumeVehicleType2Small	CalculationTrafficVolume
	trafficVolumeVehicleType2Unknown	CalculationTrafficVolume
	relationLinkageIdentifier	INTEGER(Op)
11103 trafficVolumeVehicleType4	trafficVolumeVehicleType4LargeCargo	CalculationTrafficVolume
	trafficVolumeVehicleType4Bus	CalculationTrafficVolume
	trafficVolumeVehicleType4SmallCargo	CalculationTrafficVolume
	trafficVolumeVehicleType4PassengerCar	CalculationTrafficVolume
	trafficVolumeVehicleType4Unknown	CalculationTrafficVolume
	relationLinkageIdentifier	INTEGER(Op)
11104 trafficVolumeVehicleType11	trafficVolumeVehicleType11Pedestrian	CalculationTrafficVolume
	trafficVolumeVehicleType11Bicycle	CalculationTrafficVolume
	trafficVolumeVehicleType11AutoBike	CalculationTrafficVolume
	trafficVolumeVehicleType11LightPassengerCar	CalculationTrafficVolume
	trafficVolumeVehicleType11PassengerCar	CalculationTrafficVolume
	trafficVolumeVehicleType11Bus	CalculationTrafficVolume
	trafficVolumeVehicleType11LightCargo	CalculationTrafficVolume
	trafficVolumeVehicleType11SmallCargo	CalculationTrafficVolume
	trafficVolumeVehicleType11CargoPassengerCar	CalculationTrafficVolume
	trafficVolumeVehicleType11NormalCargo	CalculationTrafficVolume
	trafficVolumeVehicleType11SpecialTruck	CalculationTrafficVolume
	trafficVolumeVehicleType11Unknown	CalculationTrafficVolume
	relationLinkageIdentifier	INTEGER(Op)

Table 6.2-8 Traffic Volume of Vehicle Type Detail

NAME		TYPE
Data Set	Data	
calculationTrafficVolume	calculationTrafficVolume1min	CalculationTrafficVolume1m
	calculationTrafficVolume1m	INTEGER
	dateTimeStart	DsDateTime(Op)
	dateTimeEnd	DsDateTime(Op)
	calculationTrafficVolume5min	CalculationTrafficVolume5m

NAME		TYPE
Data Set	Data	
	calculationTrafficVolume5m	INTEGER
	dateTimeStart	DsDateTime(Op)
	dateTimeEnd	DsDateTime(Op)
	calculationTrafficVolume10min	CalculationTrafficVolume10m
	CalculationTrafficVolume10m	INTEGER
	dateTimeStart	DsDateTime(Op)
	dateTimeEnd	DsDateTime(Op)
	calculationTrafficVolume60min	CalculationTrafficVolume60m
	CalculationTrafficVolume60m	INTEGER
	dateTimeStart	DsDateTime(Op)
	dateTimeEnd	DsDateTime(Op)
	calculationTrafficVolumeTotal	CalculationTotalTrafficVolume
	CalculationTrafficVolumeTotal	INTEGER
	dateTimeStart	DsDateTime(Op)
	dateTimeEnd	DsDateTime(Op)

6.2.7 Environment Monitoring Data DS

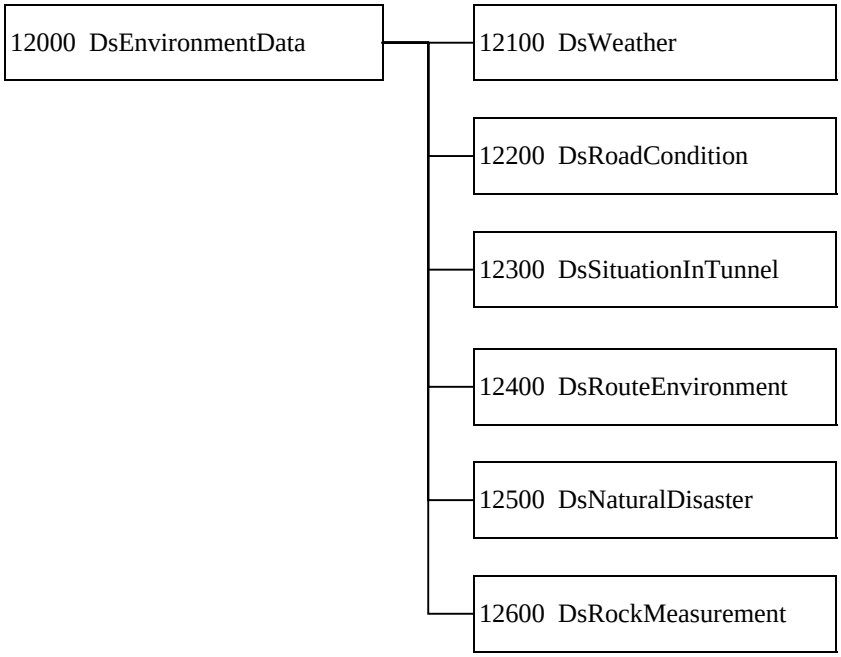


Fig. 6.2-5 Environment Monitoring Data DS

Table 6.2-9 Environment Monitoring Data DS

NAME			TYPE
Data Set		Data	
12100 dsWeather	12101 temperature	temperature0	CHOICE
		relationLinkageIdentifier	INTEGER(Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12102 humidity	calculationHumidity	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmHumidity	DeviceControlAnswerConfirmInfo(Op)
	12103 rainfall	rainfallAmount0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmRainfall	DeviceControlAnswerConfirmInfo(Op)
	12104 snowfall	snowfallAmount0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmSnowfall	DeviceControlAnswerConfirmInfo(Op)
	12105 precipitation	precipitation0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmPrecipitation	DeviceControlAnswerConfirmInfo(Op)
	12106 snowAmount	snowAmount0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmSnowAmount	DeviceControlAnswerConfirmInfo(Op)
	12107 ashfallAmount	calculationAshfallAmount	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmAshfallAmount	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12100)	12108 wind	calculation16WindDirections	ENUMERATED
		instantaneousWindSpeed	CHOICE(Op)
		calculationMaxInstantaneousWindSpeed	INTEGER(Op)
		averageWindSpeed	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmWind	DeviceControlAnswerConfirmInfo(Op)
	12109 earthquake	calculationAcceleration	INTEGER
		calculationSiValue	INTEGER
		calculationMaximumHorizontalAcceleration	INTEGER
		calculationMaximumVerticalAcceleration	INTEGER
		calculationAccelerationSpeedResponseValue	INTEGER
		eventMeasuredScaleOfAnEarthquake	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmEarthquake	DeviceControlAnswerConfirmInfo(Op)
	12110 seaLevel	calculationSeaLevel	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmSeaLevel	DeviceControlAnswerConfirmInfo(Op)
	12111 waveHeight	calculationWaveHeight	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmWaveHight	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12100)	12112 visibility	calculationSnowfallFogDensity	INTEGER(Op)
		calculationVisibility	INTEGER
		calculationTransmittance	INTEGER(Op)
		calculationTransmissivity	INTEGER(Op)
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmVisibility	DeviceControlAnswerConfirmInfo(Op)
		confirmSnowfallFogDensity	DeviceControlAnswerConfirmInfo(Op)
		confirmTransmittance	DeviceControlAnswerConfirmInfo(Op)
		confirmTransmissivity	DeviceControlAnswerConfirmInfo(Op)
	12113 radiation	calculationSolarRadiationPenetration	INTEGER
		calculationSolarRadiationTotal	INTEGER(Op)
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmRadiation	DeviceControlAnswerConfirmInfo(Op)
	12114 dayLightTime	calculationSunshineHour	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmDayLightTime	DeviceControlAnswerConfirmInfo(Op)
	12115 netRadiation	calculationNetRadiation	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmNetRadiation	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
12200 dsRoadSituation	12201 roadSurfaceCondition	roadSurfaceCondition0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12202 roadTemperature	roadTemperature0	CHOICE
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12203 roadMoisture	calculationRoadMoisture	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12204 roadSurfaceReflectionCoefficient	calculationRoadSurfaceReflectionCoefficient	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12205 waterLevel	calculationWaterLevel	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
12300 dsSituationInTunnel	12301 airTransmissivityInsideTheTunnel	calculationAirTransmissivityInsideTheTunnel	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12302 windSpeedInsideTheTunnel	calculationWindSpeedInsideTheTunnel	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12303 tunnelFireDetection	eventTunnelFireDetection	ENU
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12304 brightness	calculationBrightness	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12305 tunnelAnchorLoad	calculationAnchorLoad	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12306 tunnelSlopeOfGround	calculationSlopeOfGround	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12300)	12307 tunnelStressDisplacement	calculationStressDisplacement	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12308 tunnelCrackedDisplacement	calculationCrackedDisplacement	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12309 tunnelSlopeOfPit	calculationSlopeOfPit	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12310 tunnelVerticalElastic	calculationVerticalElastic	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12311 tunnelVerticalGround	calculationVerticalGround	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12300)	12312 tunnelTemperature	calculationTemperatureOfMeasurePoint	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12313 tunnelBedrockMutation	calculationBedrockMutation	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12314 tunnelCo	calculationCo	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
12400 dsRouteEnvironment	12401 co	calculationCo	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12402 nox	calculationNOx	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12403 so2	calculationSo2	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12404 ch	calculationCh	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12405 supendedParticulateMatter	calculationSpm	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12406 photochemicalOxident	calculationPo	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12400)	12407 noise	calculationNoise	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12408 vibration	calculationVibration	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12409 noiseLevel	calculationNoiseLevelL5	INTEGER (Op)
		calculationNoiseLevelL10	INTEGER (Op)
		calculationNoiseLevelL50	INTEGER (Op)
		calculationNoiseLevelL90	INTEGER (Op)
		calculationNoiseLevelL95	INTEGER (Op)
		calculationEquivalentSoundLevel	INTEGER (Op)
		calculationMaxSoundLevel	INTEGER (Op)
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12410 no	calculationNO	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12411 no2	calculationNO2	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
12500 dsNatural Disaster	12501 disasterDetection	eventDisasterDetection	ENUMERATED
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12502 mudSlideGroundMoisture	calculationMudSlideGroundMoisture	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
12600 dsRockMeasurement	12601 rockElasticCrack	calculationElasticCrack	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12602 rockInclination	calculationInclination	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12603 rockBedrockMutation	calculationBedrockMutation	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12604 rockTemperature	calculationTemperatureBySensor	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

NAME			TYPE
Data Set		Data	
(12600)	12605 undergroundTemperature	calculationUndergroundTemperature	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12606 rockAcousticEmissionSensor	calculationAESensor	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12607 rockRainfall	calculationRainfallAmount	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)
	12608 rockCrackedMutation	calculationCrackedMutation	INTEGER
		relationLinkageIdentifier	INTEGER (Op)
		pointLocation	DsPointLocation(Op)
		dateTime	DsDateTime(Op)
		sensorID	DsSensorID(Op)
		confirmTemperature	DeviceControlAnswerConfirmInfo(Op)

6.2.8 Structure Monitoring Data DS

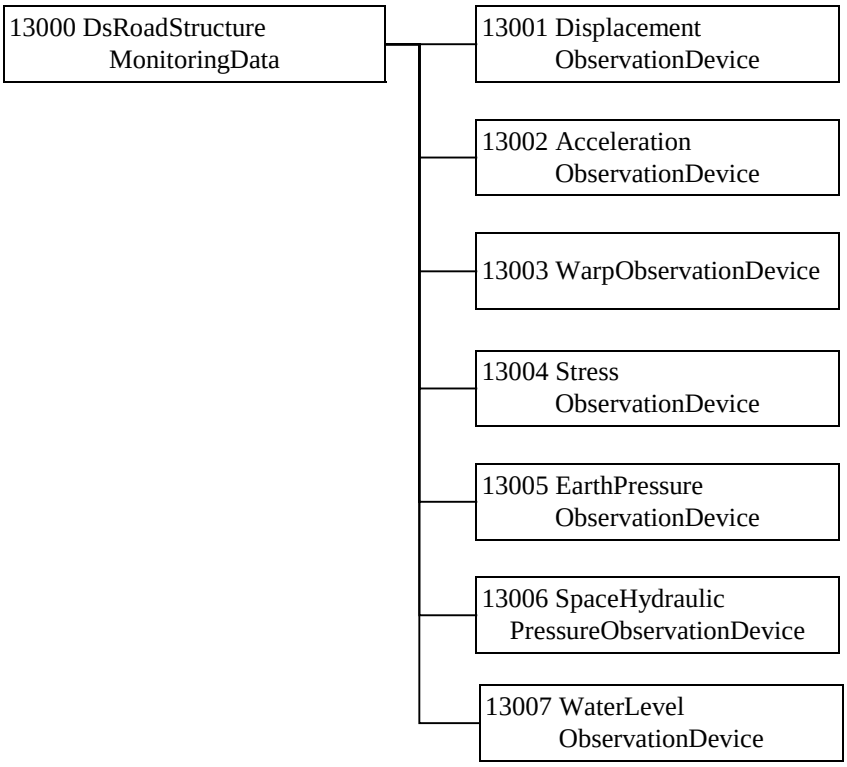


Fig. 6.2-6 Structure Monitoring Data DS

Table 6.2-10 Structure Monitoring Data DS

NAME		TYPE
Data Set	Data	
13001 displacementObservationDevice	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationDisplacement	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13002 accelerationObservationDevice	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationAcceleration	INTEGER
	relationLinkageIdentifier	INTEGER(Op)

NAME		TYPE
Data Set	Data	
13003 warpObservationDevice	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationStrain	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13004 stressObservation Device	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationStressValue	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13005 earthPressureObservationDevice	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationEarthPressure	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13006 spaceHydraulicPressureObservation Device	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationPoreWaterPressure	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13007 waterLevelObservationDevice	collectionPosition	DsLocationInfo(CHOICE)
	facilityStructureType	ENUMERATED
	facilityStructureManagementNumber	OCTET STRING
	calculationWaterLevel	INTEGER
	relationLinkageIdentifier	INTEGER(Op)
13008- For Future Use	-	-

6.2.9 Vehicle Weight Data DS

Table 6.2-11 Vehicle Weight Data DS

NAME		TYPE
Data Set	Data	
14001 dsVehicleWeight	deviceLocation	DsLocationInfo(CHOICE)(Op)
	vehicleNumber	DsVehicleNumber(SEQUENCE)(Op)
	calculationCarTypeMeasurementResult	ENUMERATED(Op)
	eventDataViolationRegulation	ENUMERATED(Op)
	calculationBodyVehicleGrossWeightMeasurementResult	INTEGER
	calculationMaximumAxialLoadMeasurementResult	INTEGER
	calculationAdjointingAxialLoadMeasurementResult	INTEGER
	relationLinkageIdentifier	INTEGER(Op)

6.2.10 Road Event Information DS

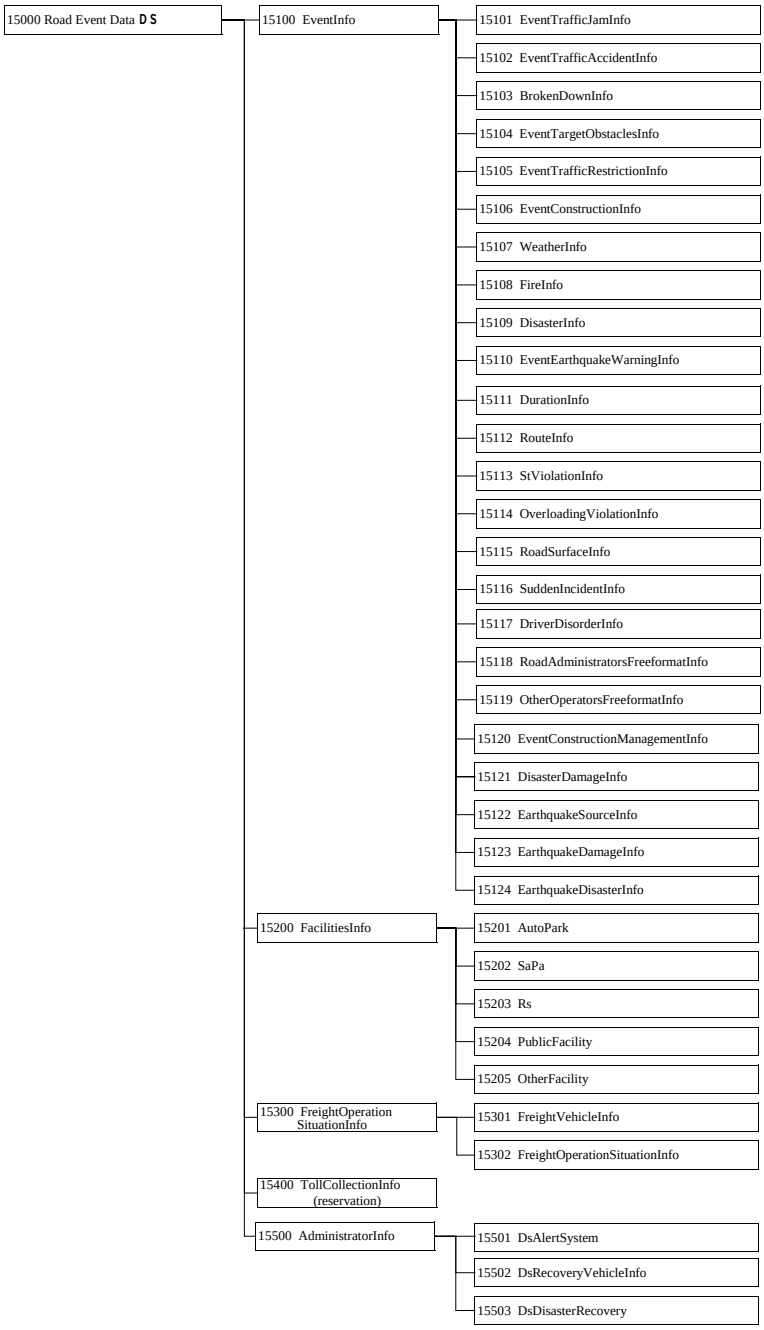


Fig. 6.2-7 Road Event Information DS

Table 6.2-12 Road Event Information DS

NAME		TYPE
Data Set	Data	
15101 eventTrafficJamInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventCauseOfTheTrafficJam	ENU
	eventCauseOfTheTrafficJamDetail	ENU(Op)
	eventTrafficJamLesStatus	ENU
	roadLaneType	ENU
	eventTrafficJamForecast	ENU(Op)
	eventTrafficJamTransitTime	INT
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventDetourInfo	DsRouteLocation(CHOICE) (Op)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	eventCauseInfo	DsEventInfo (CHOICE) (Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15102 eventTrafficAccidentInfo	informationCollectInfo	DsOrganization(SEQUENCE) (Op)
	informerInfo	DsOrganization(SEQUENCE) (Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventAccidentObject	ENU
	eventAccidentPattern	ENU
	eventAccidentHandlingConditions	ENU
	objectVehicleInfo	DsTypeTrafficVolume(CHOICE)(Op)
	personalDamageInfo	PersonalDamageInfo(SEQUENCE)(Op)
	eventNumbersOfFatalInjuriesDue	INTEGER
	eventNumbersOfSeriousInjuriesDue	INTEGER
	eventNumbersOfMediumInjuriesDue	INTEGER
	eventNumbersOfSlightInjuriesDue	INTEGER
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventDetourInfo	DsRouteLocation(CHOICE) (Op)
	eventRelationInfo	DsEventInfo(CHOICE)(Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15103 brokenDownInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventPatternOfBrokenDownCar	ENU
	locationInfo	(DS)
	eventRemovalStatusOfBrokenDownCar	ENU
	eventTypeOfBrokenDownVehicle	DsTypeTrafficVolume(CHOICE)
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15104 eventTargetObstaclesInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventTargetObstacles	ENU
	eventObstaclesForm	ENU
	eventObstaclesStatus	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRelationInfo	DsEventInfo(CHOICE)(Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15105 eventTrafficRestrictionInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventRestrictionContent	ENU
	eventRestrictionContentDetails	ENU(Op)
	eventRestrictionCause	ENU
	eventCauseDetails	ENU(Op)
	roadLaneType	ENU
	eventTrafficRestrictionVehicle	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventDetourInfo	DsRouteLocation(CHOICE)(Op)
	eventRelationInfo	DsEventInfo(CHOICE)(Op)
	relationRelationType	ENU(Op)
	eventCauseInfo	DsEventInfo(CHOICE)(Op)
	relationLinkageIdentifier	INT(Op)
	eventTrackingNumber	OCTET STRING(Op)
	eventTrafficRestrictionTotalLanes	INTEGER(Op)
	routeCourseDistance	INTEGER (Op)

NAME		TYPE
Data Set	Data	
15106 eventConstructionInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventConstructionOperationConditions	ENU
	eventWeatherConditions	ENU
	roadLaneType	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	restrictionDateTime	DsDateTime
	restrictionUpdateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventConstructionType	ENU
	eventOperationContent	ENU
	eventDetourInfo	DsRouteLocation(CHOICE) (Op)
	eventRelationInfo	DsEventInfo(CHOICE)(Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME			TYPE
Data Set	Data		
	eventConstructionName		UTF8String(Op)
	eventConstructionPurpose		UTF8String(Op)
	eventConstructionRelationInfo	eventConstructionOwnerURL	OCTET STRING(Op)
		eventProjectEvaluationURL	OCTET STRING(Op)
	eventConstructionOwnerOrganization		DsOrganization(Op)

NAME		TYPE
Data Set	Data	
15107 weatherInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventWeatherPattern	ENU
	eventWeatherConditionForecast	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRainfallStatus	ENU(Op)
	eventSnowfallStatus	ENU(Op)
	eventWindSpeedConditions	ENU(Op)
	eventWaveCondition	ENU(Op)
	eventTransmittanceDecrease	ENU(Op)
	eventVisibility	ENU(Op)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	eventPrecipitationType	ENU(Op)
	eventWeather	ENU
	eventAttentionType	ENU
	eventAttentionAndWarningContents	UTF8String
	relationLinkageIdentifier	INT(Op)
	eventMeasuredScaleOfAnEarthquake	INT(Op)
	calculationEarthquakeScale	INT(Op)

NAME		TYPE
Data Set	Data	
15108 fireInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventFireType	DsTypeTrafficVolume(CHOICE)
	eventConditionOfFire	ENU
	eventHandlingConditions	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	eventTunnelFireDetection	ENU
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15109 disasterInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDisasterType	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	eventProjectSectionNameOfDisaster Occurrence	UTF8String
	eventDisasterOccurrenceFacilityType	ENU
	eventDisasterOccurrenceFacilityQua ntity	INT
	eventCauseDetails	ENU(Op)
	eventDisasterDetection	ENU
	eventDisasterOutline	UTF8String
	eventDisasterDetail	UTF8String(Op)
	eventRestorationOutline	UTF8String
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15110 eventEarthqu akeWarningIn fo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventEarthquakeWarningAnnouncem entPlace	ENU
	dateTime	DsDateTime
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15111 durationInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	startLocationInfo	DsLocationInfo(CHOICE)
	endLocationInfo	DsLocationInfo(CHOICE)
	routeSectionUnitDuration	INT
	routeSectionAverageSpeed	INT
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	routeSectionStaticCourseDuration	INT
	routeStaticTravelSpeed	INT
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15112 routeInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	routeAttribute	ENU
	startLocationInfo	DsLocationInfo(CHOICE)
	byLocationInfo	DsLocationInfo(CHOICE)
	endLocationInfo	DsLocationInfo(CHOICE)
	routeCourseDistance	INT
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15113 stViolationInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDataViolationRegulation	ENU
	dateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15114 overloadingViolationInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDataViolationRegulation	ENU
	dateTime	DsDateTime
	locationInfo	DsPointLocation(CHOICE)
	vehicleNumber	DsVehileNumber(SEQUENCE)
	calculationBodyVehicleGrossWeightMeasurementResult	INT
	calculationMaximumAxialLoadMeasurementResult	INT
	calculationAdjointingAxialLoadMeasurementResult	INT
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15115 roadSurfaceInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	dateTime	DsDateTime
	pointLocation	DsLocationInfo(CHOICE)
	eventRoadSurfaceConditions	ENU
	eventAhsRoadSurfaceConditions13Detail	ENU
	eventAhsRoadSurfaceConditions7	ENU
	eventAhsRoadSurfaceConditions5	ENU
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15116 suddenIncidentInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	dateTime	DsDateTime
	pointLocation	DsLocationInfo(CHOICE)
	eventSuddenIncidentDetect	ENU
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15117 driverDisorderInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventAhsConfirmationUnusualBreakOutToDriver	ENU
	driverDisorderDateTime	DsDateTime
	driverDisorderLocation	DsLocationInfo(CHOICE)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15118 roadAdministratorsFreeformatInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventOtherInfomation	UTF8String
	relationLinkageIdentifier	INT(Op)
15119 otherOperatorsFreeformatInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventAnotherSubjectInfo	UTF8String
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15120 eventConstructionManagementInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventConstructionContent	ENU
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	eventProgressLengthOfConstructionLink	INT
	eventCurrentRateOfConstruction	INT
	eventNamesOfOperationalConstructionEquipmentForRestoration	ENU
	eventNumbersOfOperationalConstructionEquipmentForRestoration	INT
	eventRoadStructureToBeConstucted	ENU
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15121 disasterDamageInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo(CHOICE)
	eventRelationInfo	DsEventInfo(CHOICE) (Op)
	relationRelationType	ENU(Op)
	dsRoadSurfaceDamage	DsRoadSurfaceDamage
	dsShoulderDamage	DsShoulderDamage
	dsSlopeFaceDamage	DsSlopeFaceDamage
	dsBridgeDamage	DsBridgeDamage
	dsRetainingWallRevetmentDamage	DsRetainingWallRevetmentDamage
	dsCrossingFacilityDamage	DsCcrossingFacilityDamage
	dsTunnelDamage	DsTunnelDamage
	dsRoadAccessoriesDamage	DsRoadAccessoriesDamage
	dsCommonDuctDamage	DsCommonDuctDamage
	dsRoadBoardDamageConditions	DsRoadBoardDamageConditions
	dsBankDamage	DsBankDamage
	eventRoadDamageLength	INT
	eventRoadDamageArea	INT
	eventRoadDamageSoilVolume	INT
	eventRestorationStatus	ENU
	eventEmergencyRestorationConditions	ENU
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15122 earthquakeSourceInfo	informationCollectInfo	DsOrganization(Op)
	informerInfo	DsOrganization(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo
	calculationCentrumDepth	INT
	calculationEarthquakeScale	INT
	eventRelationInfo	DsEventInfo(Op)
	relationRelationType	ENU(Op)
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15123 earthquakeDamageInfo	informationCollectInfo	DsOrganization(Op)
	informerInfo	DsOrganization(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo
	eventRelationInfo	DsEventInfo(Op)
	relationRelationType	ENU(Op)
	dsOrganization	DsOrganization
	dsRoadObstruction	DsRoadObstruction
	dsRoadSideFire	DsRoadSideFire
	dsRoadSurfaceDamage	DsRoadSurfaceDamage
	dsShoulderDamage	DsShoulderDamage
	dsSlopeFaceDamage	DsSlopeFaceDamage
	dsBridgeDamage	DsBridgeDamage
	dsRetainingWallRevetmentDamage	DsRetainingWallRevetmentDamage
	dsCrossingFacilityDamage	DsCrossingFacilityDamage
	dsTunnelDamage	DsTunnelDamage
	dsRoadAccessoriesDamage	DsRoadAccessoriesDamage
	dsCommonDuctDamage	DsCommonDuctDamage
	dsRoadBoardDamageConditions	DsRoadBoardDamageConditions
	dsBankDamage	DsBankDamage
	eventRoadDamageLength	INT
	eventRoadDamageArea	INT
	EventRoadDamageSoilVolume	INT
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15124 earthquake DisasterInfo	informationCollectInfo	DsOrganization(Op)
	informerInfo	DsOrganization(Op)
	relationEventIdentifier	INT(Op)
	eventStatusCode	ENU
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	locationInfo	DsLocationInfo
	eventRelationInfo	DsEventInfo(Op)
	relationRelationType	ENU(Op)
	eventRestorationStatus	ENU
	eventEmergencyRestorationConditions	ENU
	eventrestorationOutline	UTF8String
	relationLinkageIdentifier	INT(Op)
15125-99	For Future Use	-

NAME		TYPE
Data Set	Data	
15201 autoPark	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	dsDataTime	DsDateTime
	facilityAutoParkName	UTF8String
	facilityAutoParkTelephoneNum	UTF8String
	facilityFacilityCode	OCTET STRING
	entranceLocationInfo	DsLocationInfo(CHOICE)
	locationInfo	DsLocationInfo(CHOICE)
	facilityAutoParkAddress	UTF8String
	facilityAutoParkType	UTF8String
	facilityAutoParkStructure	UTF8String
	facilityAmountVehiclesToBeAccommodated	ENU
	facilityAutoParkCapacity	INT
	facilityHeightRestriction	ENU
	facilityVehicleTypeRestriction	ENU
	facilityAutoParkLengthRestriction	UTF8String
	eventDateTime	DsDateTime
	updateTime	DsDateTime
	businessDay	DsDateTime
	facilityParkingFee	INT
	facilityChargeUnit	ENU
	facilityDiscountedFee	ENU
	facilityAutoParkFee	INT
	facilityCongestionStatus	ENU
	facilityParkingTurnoverRate	INT
	facilityWaitingTime	INT
	facilityNumberOfParkingVehicles	INT
	facilityCongestionDegreeForecast	ENU
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15202 saPa	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	dsDateTime	DsDateTime
	facilitySaPaName	UTF8String
	facilityFacilityCode	OCTET STRING
	locationInfo	DsLocationInfo(CHOICE)
	facilityFacilityContent	UTF8String
	startDateTime	DsDateTime
	endDateTime	DsDateTime
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15203 rs	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	dsDateTime	DsDateTime
	facilityRsName	UTF8String
	facilityFacilityCode	OCTET STRING
	locationInfo	DsLocationInfo(CHOICE)
	facilityFacilityContent	UTF8String
	startDateTime	DsDateTime
	endDateTime	DsDateTime
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15204 publicFacility	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	dsDateTime	DsDateTime
	facilityNameOfFacility	UTF8String
	facilityTypeOfFacility	UTF8String
	facilityAddressOfFacility	UTF8String
	facilityTelephoneNumOfFacility	UTF8String
	facilityOpenDay	UTF8String
	openHoursOpen	DsDateTime
	openHoursClose	DsDateTime
	facilityChargeOfFacility	UTF8String
	facilityInformationOfEvent	UTF8String(Op)
	facilityCapacityOfFacility	INT
	facilityNumberOfBooking	INT
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15205 otherFacility	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	dsDateTime	DsDateTime
	facilityOtherFacilityName	UTF8String
	locationInfo	DsLocationInfo(CHOICE)
	facilityFacilityCode	OCTET STRING
	facilityFacilityContent	UTF8String
	startDateHour	DsDateTime
	endDateHour	DsDateTime
	relationLinkageIdentifier	INT(Op)
15206-99	For Future Use	-

NAME		TYPE
Data Set	Data	
15301 freightVehicleInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	movableOnboardDeviceVehicleCode	ENU
	movableVehicleLicencePlateNumber	OCTET STRING
	movableVehicleLengthOverall	INT
	movableVehicleHeightOverall	INT
	movableVehicleWidth	INT
	movableBodyCapacityCargoLoad	INT
	movableVehicleWeight	INT
	movableBodyVehicleGrossWeight	INT
	calculationVehicleSpecificCharacteristics	UTF8String
	statisticsBodyCapacityPassengers	INT
	movableTotalPistonDisplacement	INT
	movableBodyFuelType	ENU
	movableAxleLoadBeforehand	INT
	movableLongitudinalAxisMultiple	INT
	movablePostfrontAxleMultiple	INT
	movableAxleLoadFuture	INT
	movableVehicleIdentificationNumber	INT
	calculationCarTypeMeasurementResult	ENU
	calculationAxisNumberMeasurementResult	INT
	calculationLicencePlateNumberMeasurementResult	OCTET STRING
	calculationAxleLoadMeasurementResult	INT
	calculationVehicleHeightMeasurementResult	INT
	relationLinkageIdentifier	INT(Op)
15302 freightOperationSituationInfo	movableFreightOperationSituationInfo	UTF8String
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15501 dsAlertSystem	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	eventStatusCode	ENU
	startDateTime	DsDateTime
	inputDateTime	DsDateTime
	eventOfficeAlertSystemCode	ENU
	eventOfficeSystemName	UTF8String
	regionalConstructionBureauDateTime	DsDateTime
	regionalConstructionBureauInputTime	DsDateTime
	eventRegionalConstructionBureauSystemCode	ENU
	eventRegionalConstructionBureauSystemName	UTF8String
	relationLinkageIdentifier	INT(Op)
15502 dsRecoveryVehicleInfo	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	eventStatusCode	ENU
	movableVehicleCode	INT
	eventConstructionEquipmentNameOwnedByConstructionOffice	ENU
	movableOnboardDeviceID	INT
	organizationOrganizationCode	DsOrganization(SEQUENCE)
	relationLinkageIdentifier	INT(Op)
15503 dsDisasterRecovery	informationCollectInfo	DsOrganization(SEQUENCE)(Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	eventStatusCode	ENU
	pointLocation	DsLocationInfo(CHOICE)
	collectionDateTime	DsDateTime
	eventMaterialTypeLoadedIntoRestorationVehicle	ENU
	eventMaterialAmountLoadedIntoRestorationVehicle	INT
	eventNameOfDisasterMeasureConstructionEquipmentAccordingToDeploymentPlan	ENU
	eventNumberOfDisasterMeasureConstructionEquipmentAccordingToDeploymentPlan	INT
	eventConstructionEquipmentNameOwnedByConstructionOffice	ENU
	eventNumberOfConstructionEquipmentsOwnedByConstructionOffice	INT
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
15504 dsInformationBoardProvidingInfo	informationCollectInfo	DsOrganization(SEQUENCE) (Op)
	informerInfo	DsOrganization(SEQUENCE)(Op)
	provisionStartTime	DsDateTime(SET)（Op）
	provisionUpdateEndTime	DsDateTime(SET) （Op）
	provisionDeviceMngInfo	MDeviceMngInfo
	provisionDeviceLocationInfo	DsLocationInfo
	provisionDeviceStatus	DeviceControlAnswerConfirmInfo
	provisionContents	SEQUENCE OF ProvisionContents
	relationLinkageIdentifier	INT(Op)
15505-99	For Future Use	-

NAME		TYPE
Data Set	Data	
ProvisionContents	watchInfoProvisionInfoDef	INT(Op)
	watchInfoProvisionInfoDefSub	INT(Op)
	watchInfoDisplayCharacterContent	UTF8String(Op)
	imageContents	ImageInfo(Op)
	imageDataFormat	ENU
	imageDataBody	OCTET STRING

6.2.11 Commercial Vehicle Operator Information DS

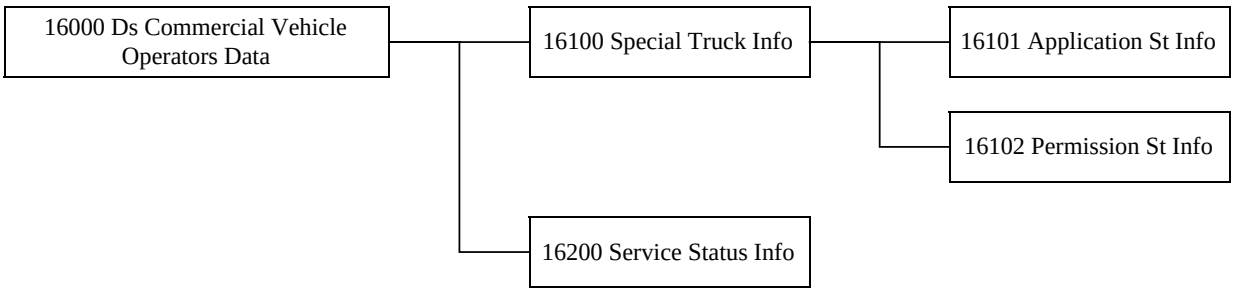


Figure 6.2-8 Commercial Vehicle Operator Information DS

Table 6.2-13 Commercial Vehicle Operator Information DS

NAME		TYPE
Data Set	Data	
16101 applicationStInfo	specialtruckApplicantPersonName	OCTET STRING
	dateDates	DsDateTime
	frontVehicleNumber	DsVehicleNumber(SEQUENCE)
	rearVehicleNumber	DsVehicleNumber(SEQUENCE)
	specialtruckVehicleTypeApplication	ENU
	specialtruckVehicleWidthApplication	INT
	specialtruckVehicleHeightApplication	INT
	specialtruckVehicleLengthApplication	INT
	specialtruckGrossWeightApplication	INT
	specialtruckMaximumAxialWeightApplication	INT
	specialtruckAdjointingAxialWeightApplication	INT
	specialtruckCargoNameApplication	ENU
	specialtruckCargoWidthApplication	INT
	specialtruckCargoHeightApplication	INT
	specialtruckCargoLengthApplication	INT
	movableTruckCargoLoad	INT
	roadRoute	DsRouteLocation(CHOICE)
	specialtruckApplicationContent	UTF8String
	relationLinkageIdentifier	INT(Op)

NAME		TYPE
Data Set	Data	
16102 PermissionStInfo	specialtruckPermissionPerson	OCTET STRING
	dateDatesTimeStart	DsDateTime
	dateDatesTimeEnd	DsDateTime
	specialtruckPermissionCondition	UTF8String
	specialtruckPermissionNumber	INT
	frontVehicleNumber	DsVehicleNumber(SEQUENCE)
	rearVehicleNumber	DsVehicleNumber(SEQUENCE)
	specialtruckVehicleTypePermission	ENU
	specialtruckVehicleWidthPermission	INT
	specialtruckVehicleHeightPermission	INT
	specialtruckVehicleLengthPermission	INT
	specialtruckGrossWeightPermission	INT
	specialtruckMaximumAxialWeightPermission	INT
	specialtruckAdjointingAxialWeightPermission	INT
	specialtruckCargoNamePermission	ENU
	specialtruckCargoWidthPermission	INT
	specialtruckCargoHeightPermission	INT
	specialtruckCargoLengthPermission	INT
	movableTruckCargoLoad	INT
	roadRoute	DsRouteLocation(CHOICE)
	specialtruckApplicationContent	UTF8String
	relationLinkageIdentifier	INT(Op)
16103 -16199	For Future Use	-

6.3 ASN.1 Module

<ASN.1 Description>

```
RCS-message-set
DEFINITIONS   AUTOMATIC TAGS   EXTENSIBILITY IMPLIED ::=
BEGIN
EXPORTS RCS-Message;
IMPORTS DatesDateOfYear, DatesUnitOfTime, LocationXYNormalCoordinates,
        LocationLatitudeLongitudeDegree FROM RCS-data-dictionary;

Header ::= SEQUENCE{
        applicationID                   ApplicationID           OPTIONAL,
        messageSetID                    MessageSetID            ,
        messageTimeStamp                DsDateTime              ,
        messageSetVersion               MessageSetVersion  OPTIONAL,
        messageCheck                    MessageCheck            OPTIONAL
        }
MessageSetVersion::=INTEGER
MessageCheck::= BIT STRING

ApplicationID ::= SEQUENCE{
        organizationCode                DsOrganization          ,
        messageApplicationId            MessageApplicationId
        }
MessageApplicationId ::= INTEGER
--{      Not Defined      }

MessageSetID::= INTEGER{
        initialRequest (0)                                      ,
        initialResponse (1)                                     ,
                        --Traffic Volume Data Collection Component
        msTrafficVolumeDataCollectionRequest (1010)             ,
        msTrafficVolumeDataCollectionResponse (1011)            ,
                        --Environment Data Collection Component
        msEnvironmentDataCollectionRequest (1020)               ,
        msEnvironmentDataCollectionResponse (1021)              ,
```

```
                --Road Structure Monitering Data Component
msRoadStructureMoniteringRequest (1030)                 ,
msRoadStructureMoniteringResponse (1031)                ,
                --Vehicle Weight Data Collection Component
msVehicleWeightDataCollectionRequest (1040)             ,
msVehicleWeightDataCollectionResponse (1041)            ,
                --Road Event Data Collection Component
msRoadEventDataCollectionRequest (1050)                 ,
msRoadEventDataCollectionResponse (1051)                ,
                --Road/Vehicle Communication Data Collection Component
msRoadVehicleCommunicationDataCollectionRequest (1060)  ,
msRoadVehicleCommunicationDataCollectionResponse (1061) ,
                --Road Event Data Provision Component
msRoadEventDataProvisionRequest (2010)                  ,
msRoadEventDataProvisionResponse (2011)                 ,
                --Road/Vehicle Communication Data Provision Component
msRoadVehicleCommunicationDataProvisionRequest (2020)   ,
msRoadVehicleCommunicationDataProvisionResponse (2021)  ,
                --Data Exchange Between Road Administrators Component
msDataExchangeBetweenRoadAdministratorsRequest (3010)   ,
msDataExchangeBetweenRoadAdministratorsResponse (3011)  ,
                --Data Exchange Between Other Operators Component
msDataExchangeBetweenOtherOperatorsRequest (3020)       ,
msDataExchangeBetweenOtherOperatorsResponse (3021)      ,
                --Data Exchange Between Commercial Vehicle Operators Component
msExchangeDataBetweenCommercialVehicleOperators (3031)  ,
                --Collection Device Operation Component
msCollectionDeviceOperationDefineInfo (4111)            ,
msCollectionDeviceOperationRequest (4120)               ,
msCollectionDeviceOperationResponse (4121)              ,
                --Provision Device Operation Component
msProvisionDeviceOperationDefineInfo (4211)             ,
msProvisionDeviceOperationRequest (4220)                ,
msProvisionDeviceOperationResponse (4221)               ,
                --Provision Device Information Comfirmation
msProvisionDeviceOperationInInfo (4231)
}
```

```
RCS-Message::=CHOICE{
        initialRequest                          InitialRequest                  ,
        initialReaponse                         InitialResponse                 ,
        msTrafficVolumeDataCollectionRequest    MsTrafficVolumeDataCollectionRequest,
        msTrafficVolumeDataCollectionResponse
                                                MsTrafficVolumeDataCollectionResponse,
        msEnvironmentDataCollectionRequest      MsEnvironmentDataCollectionRequest,
        msEnvironmentDataCollectionResponse     MsEnvironmentDataCollectionResponse,
        msRoadStructureMonitoringRequest        MsRoadStructureMonitoringRequest,
        msRoadStructureMonitoringResponse       MsRoadStructureMonitoringResponse,
        msVehicleWeightDataCollectionRequest
                                        MsVehicleWeightDataCollectionRequest ,
        msVehicleWeightDataCollectionResponse
                                        MsVehicleWeightDataCollectionResponse,
        msRoadEventDataCollectionRequest
                                        MsRoadEventDataCollectionRequest     ,
        msRoadEventDataCollectionResponse
                                        MsRoadEventDataCollectionResponse    ,
        msRoadVehicleCommunicationDataCollectionRequest
                                        MsRoadVehicleCommunicationDataCollectionRequest,
        msRoadVehicleCommunicationDataCollectionResponse
                                        MsRoadVehicleCommunicationDataCollectionResponse,
        msRoadEventDataProvisionRequest         MsRoadEventDataProvisionRequest,
        msRoadEventDataProvisionResponse        MsRoadEventDataProvisionResponse,
        msRoadVehicleCommunicationDataProvisionRequest
                                        MsRoadVehicleCommunicationDataProvisionRequest,
        msRoadVehicleCommunicationDataProvisionResponse
                                        MsRoadVehicleCommunicationDataProvisionResponse,
        msDataExchangeBetweenRoadAdministratorsRequest
                                        MsDataExchangeBetweenRoadAdministratorsRequest,
        msDataExchangeBetweenRoadAdministratorsResponse
                                        MsDataExchangeBetweenRoadAdministratorsResponse,
        msDataExchangeBetweenOtherOperatorsRequest
                                        MsDataExchangeBetweenOtherOperatorsRequest,
        msDataExchangeBetweenOtherOperatorsResponse
                                        MsDataExchangeBetweenOtherOperatorsResponse,
```

```
        msExchangeDataBetweenCommercialVehicleOperators
                                MsExchangeDataBetweenCommercialVehicleOperators
        }

InitialRequest::=SEQUENCE{
        commonHeader                Header              ,
        version                     Version
        }

InitialResponse::=SEQUENCE{
        commonHeader                Header              ,
        version                     Version             ,
        result                      BOOLEAN
        --reject=FALSE,Accept=TRUE
        }

Version::=SEQUENCE OF VersionID
VersionID::=INTEGER(6)

MsTrafficVolumeDataCollectionRequest ::= SEQUENCE{
        commonHeader                Header              ,
        requestArea                 DsLocationInfo      OPTIONAL,
        requestTerm                 MRequestTerm        OPTIONAL,
        requestDataSetNumber        DataSetNumber       OPTIONAL
        }

MRequestTerm ::= SEQUENCE{
        requestStartDateTime        DsDateTime          ,
        requestEndDateTime          DsDateTime
        }

MsTrafficVolumeDataCollectionResponse ::= SEQUENCE{
        commonHeader                Header              ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                        OPTIONAL,
        dataConcentrationDeviceMngInfo  MDeviceMngInfo  ,
        dataConcentrationDeviceLocation DsLocationInfo  OPTIONAL,
```

```
        dataCollectionTerm              MDataCollectionTerm     ,
        collectionTrafficVolumeData     DsTrafficVolumeData
        }
DeviceControlAnswerConfirmInfo::=ENUMERATED{
        normal(0),abnormal(1),uncollected(2),invalidData(9)
        }

MDeviceMngInfo ::= SEQUENCE{
        deviceManagementOrganization    DsOrganization          ,
        investDefinitionSystemCode      INTEGER                 ,
        deviceManagementNumber          UTF8String              ,
        deviceSubManagementNumber       UTF8String              OPTIONAL
        }

MDataCollectionTerm::=SEQUENCE{
        collectionStartDateTime         DsDateTime              ,
        collectionEndDateTime           DsDateTime              OPTIONAL
        }

MsEnvironmentDataCollectionRequest ::= SEQUENCE{
        commonHeader                    Header                  ,
        requestArea                     DsLocationInfo          OPTIONAL,
        requestTerm                     MRequestTerm            OPTIONAL,
        requestDataSetNumber            DataSetNumber           OPTIONAL
        }

MsEnvironmentDataCollectionResponse ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        dataCollectionDeviceMngInfo     MDeviceMngInfo          ,
        dataCollectionDeviceLocation    DsLocationInfo          OPTIONAL,
        dataCollectionTerm              MDataCollectionTerm     ,
        collectionEnvironmentMonitoringData SET OF DsEnvironmentMonitoringData
        }

MsRoadStructureMonitoringRequest   ::= SEQUENCE{
```

```
        commonHeader                    Header                  ,
        requestArea                     DsLocationInfo          OPTIONAL,
        requestTerm                     MRequestTerm            OPTIONAL,
        requestDataSetNumber            DataSetNumber           OPTIONAL
        }

MsRoadStructureMonitoringResponse ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        dataConcentrationDeviceMngInfo  MDeviceMngInfo          ,
        dataConcentrationDeviceLocation DsLocationInfo          OPTIONAL,
        dataCollectionTerm              MDataCollectionTerm     ,
        collectionRoadStructureMonitoringData DsRoadStructureMonitoringData
        }

MsVehicleWeightDataCollectionRequest   ::= SEQUENCE{
        commonHeader                    Header                  ,
        requestArea                     DsLocationInfo          OPTIONAL,
        requestTerm                     MRequestTerm            OPTIONAL
        }

MsVehicleWeightDataCollectionResponse ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        dataConcentrationDeviceMngInfo  MDeviceMngInfo          ,
        dataConcentrationDeviceLocation DsLocationInfo          OPTIONAL,
        dataCollectionTerm              MDataCollectionTerm     ,
        collectionWeightData            DsVehicleWeightData
        }

MsRoadEventDataCollectionRequest ::= SEQUENCE{
        commonHeader                    Header                  ,
        requestArea                     DsLocationInfo          ,
        requestTerm                     MRequestTerm            ,
        requestDataSetList              SET OF DataSetNumber
```

```
        }

MsRoadEventDataCollectionResponse ::= SEQUENCE{
        commonHeader                    Header                          ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        dataConcentrationDeviceMngInfo  MDeviceMngInfo                  ,
        dataConcentrationDeviceLocation DsLocationInfo                  OPTIONAL,
        collectionRoadEventData         SET OF DsRoadEventData
        }

MsRoadVehicleCommunicationDataCollectionRequest ::= SEQUENCE{
        commonHeader                    Header                          OPTIONAL,
        deviceSecurityInfo              DeviceSecurityInfo              OPTIONAL,
        requestArea                     DsLocationInfo                  OPTIONAL,
        requestTerm                     MRequestTerm                    OPTIONAL,
        requestDataSetList              SET OF DataSetNumber
        }
DeviceSecurityInfo::=BIT STRING

MsRoadVehicleCommunicationDataCollectionResponse   ::= SEQUENCE{
        commonHeader                    Header                          OPTIONAL,
        deviceSecurityInfo              DeviceSecurityInfo              OPTIONAL,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        collectionRoadEventData         SET OF DsRoadEventData
        }

MsRoadEventDataProvisionRequest ::= SEQUENCE{
        commonHeader                    Header                          ,
        deviceMngInfo                   MDeviceMngInfo                  ,
        requestArea                     DsLocationInfo                  OPTIONAL,
        requestTerm                     MRequestTerm                    OPTIONAL,
        requestDataSetList              SET OF DataSetNumber
        }

MsRoadEventDataProvisionResponse ::= SEQUENCE{
```

```
        commonHeader                    Header                          ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        provisionDeviceMngInfo          MDeviceMngInfo                  OPTIONAL,
        provisionRoadEventData          SET OF DsRoadEventData
        }

MsRoadVehicleCommunicationDataProvisionRequest ::= SEQUENCE{
        commonHeader                    Header                          OPTIONAL,
        deviceSecurityInfo              DeviceSecurityInfo              OPTIONAL,
        messageUserInfo                 MessageUserInfo                 OPTIONAL,
        requestArea                     DsLocationInfo                  OPTIONAL,
        requestTerm                     MRequestTerm                    OPTIONAL,
        requestDataSetList              SET OF DataSetNumber
        }
MessageUserInfo::=INTEGER

MsRoadVehicleCommunicationDataProvisionResponse ::= SEQUENCE{
        commonHeader                    Header                          OPTIONAL,
        deviceSecurityInfo              DeviceSecurityInfo              OPTIONAL,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        provisionRoadEventData          SET OF DsRoadEventData
        }

MsDataExchangeBetweenRoadAdministratorsRequest ::= SEQUENCE{
        commonHeader                    Header                          ,
        messageUserInfo                 MessageUserInfo                 ,
        requestArea                     DsLocationInfo                  OPTIONAL,
        requestTerm                     MRequestTerm                    OPTIONAL,
        requestDataSetList              SET OF DataSetNumber            OPTIONAL
        }

MsDataExchangeBetweenRoadAdministratorsResponse ::= SEQUENCE{
        commonHeader                    Header                          ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
```

```
        dataMngInfo                     MDataMngInfo            OPTIONAL,
        responseDataSet                 SET OF DataSetData
        }

MDataMngInfo ::= SEQUENCE{
        informationManagementOrganization   DsOrganization      OPTIONAL,
        investDefinitionSystemCode          INTEGER             OPTIONAL,
        deviceInformationManagementNumber   INTEGER             OPTIONAL
        }

MsDataExchangeBetweenOtherOperatorsRequest ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceSecurityInfo              DeviceSecurityInfo      ,
        messageUserInfo                 MessageUserInfo         ,
        requestArea                     DsLocationInfo          OPTIONAL,
        requestTerm                     MRequestTerm            OPTIONAL,
        requestDataSetList              SET OF DataSetNumber
        }

MsDataExchangeBetweenOtherOperatorsResponse ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceSecurityInfo              DeviceSecurityInfo      ,
        deviceControlAnswerConfirmInfo  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        responseDataSet                 SET OF DataSetData
        }

MsExchangeDataBetweenCommercialVehicleOperators ::= SEQUENCE{
        commonHeader                    Header                  ,
        deviceSecurityInfo              DeviceSecurityInfo      ,
        commercialVehicleOperatorsData  DsCommercialVehicleOperatorsData
        }

DataSetNumber::=INTEGER{
        dsDateTime(1000)                ,
        dsLocationInfo(2000)            ,
        dsPointLocation(2100)           ,
```

dsLocationRoadPosition(2110) ,
dsExpresswaySpotKp(2111) ,
dsNationalRoadSpotKp(2112) ,
dsRoadLinkSpot(2113) ,
dsLocationCode(2114) ,
dsBaseRoadLinkSpot(2115) ,
dsRoadSpotTwoDimensional(2120) ,
dsCoordinatesLocation(2130) ,
dsNormalCoordinates(2131) ,
dsLatitudeLongitude(2132) ,
dsLatitudeLongitudeType2(2133) ,
dsLatitudeLongitudeType3(2134) ,
dsRoadSpotThreeDimensional(2140) ,
dsCoordinatesLocationThreeDimensional(2150),
dsAddress(2160) ,
dsSectionLocation(2200) ,
dsStartEndSection(2201) ,
dsRoadLinkSection(2202) ,
dsSpanSection(2203) ,
dsBaseRoadLinkSection(2204) ,
dsAreaLocation(2300) ,
dsRouteLocation(3000) ,
dsPointRouteLocation(3100) ,
dsLinkRouteLocation(3200) ,
dsSpanRouteLocation(3300) ,
dsBaseRoadLinkRouteLocation(3400) ,
dsOrganization(4000) ,
dsMovableBodyInfo(5000) ,
dsVehicleNumber(5001) ,
dsTrafficVolumeData(11000) ,
dsTrafficVolume (11100) ,
trafficVolumeVehicleType1 (11101) ,
trafficVolumeVehicleType2 (11102) ,
trafficVolumeVehicleType4 (11103) ,
trafficVolumeVehicleType11 (11104) ,
dsPassingVehicle (11200) ,
dsAhsVehicleInfo (11300) ,

dsEnvironmentMonitoringData(12000) ,
dsWeather(12100) ,
temperature(12101) ,
humidity(12102) ,
rainfall(12103) ,
snowfal(12104) ,
precipitation(12105) ,
snowAmount(12106) ,
ashfallAmount(12107) ,
wind(12108) ,
earthquake(12109) ,
seaLevel(12110) ,
waveHight(12111) ,
visibility(12112) ,
radiation(12113) ,
dayLightTime(12114) ,
netRadiation(12115) ,
dsRoadSituation(12200) ,
roadSurfaceCondition(12201) ,
roadTemperature(12202) ,
roadMoisture(12203) ,
roadSurfaceReflectionCoefficient(12204),
waterLevel(12205) ,
dsSituationInTunnel(12300) ,
airTransmissivityInsideTheTunnel(12301),
windSpeedInsideTheTunnel(12302) ,
tunnelFireDetection(12303) ,
brightness(12304) ,
tunnelAnchorLoad(12305) ,
tunnelSlopeOfGround(12306) ,
tunnelStressDisplacement(12307) ,
tunnelCrackedDisplacement(12308) ,
tunnelSlopeOfPit(12309) ,
tunnelVerticalElastic(12310) ,
tunnelVerticalGround(12311) ,
tunnelTemperature(12312) ,
tunnelBedrockMutation(12313) ,

tunnelCo(12314) ,
dsRouteEnvironment(12400) ,
co(12401) ,
nox(12402) ,
so2(12403) ,
ch(12404) ,
supendedParticulateMatter(12405) ,
photochemicalOxident(12406) ,
noise(12407) ,
vibration(12408) ,
noiseLevel(12409) ,
no(12410) ,
no2(12411) ,
dsNaturalDisaster(12500) ,
disasterDetection(12501) ,
mudSlideGroundMoisture(12502) ,
dsRockMeasurement(12600) ,
rockElasticCrack(12601) ,
rockInclination(12602) ,
rockBedrockMutation(12603) ,
rockTemperature(12604) ,
undergroundTemperature(12605) ,
rockAcousticEmissionSensor(12606) ,
rockRainfall(12607) ,
rockCrackedMutation(12608) ,
dsRoadStructureMonitoringData(13000) ,
displacementObservationDevice(13001) ,
accelerationObservationDevice(13002) ,
warpObservationDevice(13003) ,
stressObservationDevice(13004) ,
earthPressureObservationDevice(13005) ,
spaceHydraulicPressureObservationDevice(13006),
waterLevelObservationDevice(13007) ,
dsVehicleWeightData(14000) ,
dsVehicleWeight(14001) ,
dsRoadEventData(15000) ,
dsEventInfo(15100) ,

eventTrafficJamInfo(15101) ,
eventTrafficAccidentInfo(15102) ,
brokenDownInfo(15103) ,
eventTargetObstaclesInfo(15104) ,
eventTrafficRestrictionInfo(15105) ,
eventConstructionInfo(15106) ,
weatherInfo(15107) ,
fireInfo(15108) ,
disasterInfo(15109) ,
eventEarthquakeWarningInfo(15110) ,
durationInfo(15111) ,
routeInfo(15112) ,
stViolationInfo(15113) ,
overloadingViolationInfo(15114) ,
roadSurfaceInfo(15115) ,
suddenIncidentInfo(15116) ,
driverDisorderInfo(15117) ,
roadAdministratorsFreeformatInfo(15118),
otherOperatersFreeformatInfo(15119) ,
eventConstructionManagementInfo(15120),
disasterDamageInfo(15121) ,
earthquakeSourceInfo(15122) ,
earthquakeDamageInfo(15123) ,
earthquakeDisasterInfo(15124) ,
dsFacilitiesInfo(15200) ,
autoPark(15201) ,
saPa(15202) ,
rs(15203) ,
publicFacility(15204) ,
otherFacility(15205) ,
dsFreightOperationSituationsInfo(15300),
freightVehicleInfo(15301) ,
freightOperationSituationInfo(15302) ,
dsTollCollectionInfo(15400) ,
dsRoadAdministratorInfo(15500) ,
dsAlertSystem(15501) ,
dsRecoveryVehicleInfo (15502) ,

```
        dsDisasterRecovery(15503)           ,
        dsInformationBoardProvidingInfo(15504),
        dsCommercialVehicleOpreatorsData(16000),
        specialTruckInfo(16100)             ,
        applicationStInfo(16101)            ,
        permissionStInfo(16102)             ,
        serviceStatusInfo(16200)            ,
        dsVideoExchange(17000)              ,
        dsDistributeOption(17100)           ,
        dsDistributeStatus(17200)           ,
        dsConfigurationElement (17300)      ,
        dsAddDeleteEquipment (17301)        ,
        dsAddDeleteRoute (17302)            ,
        dsAddDeleteManagementPlace (17303)  ,
        dsAlarmInformation(17400)           ,
        dsEquipmentID(17500)                ,
        dsDeviceInfo(17501)                 ,
        dsRelatedDeviceInfo(17600)          ,
        dsRelatedVideoSwitcher(17601)       ,
        dsRelatedMultiViewer(17602)
        }

DataSetData::=CHOICE{
        dsTrafficVolumeData             DsTrafficVolumeData         ,
        dsEnvironmentMonitoringData     DsEnvironmentMonitoringData,
        dsRoadStructureMonitoringData   DsRoadStructureMonitoringData,
        dsVehicleWeightData             DsVehicleWeightData         ,
        dsRoadEventData                 DsRoadEventData             ,
        dsCommercialVehicleOperatorsData DsCommercialVehicleOperatorsData
        --dsVideoExchange               DsVideoExchange
        }

DsDateTime ::= SEQUENCE{
        datesDateOfYear                 DatesDateOfYear             OPTIONAL,
        datesUnitOfTime                 DatesUnitOfTime             OPTIONAL,
        datesDayOfTheWeek               ENUMERATED {
                sunday(0),monday(1),tuesday(2),wednesday(3),thursday(4),friday(5),saturday(6),
```

```
                invalidData(9)}                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                    OPTIONAL
        }

DsLocationInfo ::=CHOICE{
        dsPointLocation                 DsPointLocation            ,
        dsSectionLocation               DsSectionLocation          ,
        dsAreaLocation                  DsAreaLocation             ,
        dsRouteLocation                 DsRouteLocation
        }

DsPointLocation ::=SEQUENCE{
        dsLocationRoadPosition          DsLocationRoadPosition     OPTIONAL,
        dsRoadSpotTwoDimensional        DsRoadSpotTwoDimensional   OPTIONAL,
        dsCoordinatesLocation           DsCoordinatesLocation      OPTIONAL,
        dsRoadSpotThreeDimensional      DsRoadSpotThreeDimensional
                                                                   OPTIONAL,
        dsCoordinatesLocationThreeDimensional
                                        DsCoordinatesLocationThreeDimensional
                                                                   OPTIONAL,
        dsAddress                       DsAddress                  OPTIONAL
        }

DsLocationRoadPosition::= SEQUENCE{
        dsExpresswaySpotKp              DsExpresswaySpotKp         OPTIONAL,
        dsNationalRoadSpotKp            DsNationalRoadSpotKp       OPTIONAL,
        dsRoadLinkSpot                  DsRoadLinkSpot             OPTIONAL,
        dsLocationCode                  DsLocationCode             OPTIONAL,
        dsBaseRoadLinkSpot              DsBaseRoadLinkSpot         OPTIONAL,
        dsLocalRoadSpotKp               DsLocalRoadSpotKp          OPTIONAL
        }

DsExpresswaySpotKp ::=SEQUENCE{
        roadTollRoadRouteCode           UTF8String                 ,
        roadTollRoadRouteSubCode        ENUMERATED{mainLine(0),tunnel(1),
                mainLineBarrier(2),tollbooth(10),restArea(11),busStop(12),inflowRamp(20),
                runoffRamp(21),junction(30),other(90),invalidData(99)}  OPTIONAL,
```

```
        roadOrganization                DsOrganization          ,
        roadRouteName                   UTF8String              OPTIONAL,
        roadRoadType                    ENUMERATED{nationalExpressway(1),
                cityHighway(2),generalTollRoad(3),theWayOnlyForCars(4),
                generalNationalHighwaySpecifiedSection(5),
                generalNationalHighwayOutsideSpecifiedSection(6),mainDistrictWay(7),
                generalAllPrefecturesWay(8),municipalRoad(9),other(98),invalidData(99)},
        roadRouteDirectionCode          ENUMERATED{
                none(0), noDirection(1), up(2), down(3), inbound(4), outbound(5),
                upInbound(6), downOutbound(7), updown(8), eastbound(9), westbound(10),
                northbound(11), southbound(12), bothDirections(13), upAnotherLane(14),
                downAnotherLane(15), upLeft(16), downLeft(17), upRight(18), downRight(19),
                upBothRoutes(20), downBothRoutes(21), inboundLeft(22), outboundLeft(23),
                inboundRight(24), outboundRight(25), invalidData(97),other(98), unknown(99)},
        locationTollRoadSpotKp          INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsNationalRoadSpotKp::=SEQUENCE{
        roadNationalwayRouteNumber      INTEGER                 ,
        roadNationalwayRouteSubCode     INTEGER                 OPTIONAL,
        roadRouteName                   UTF8String              OPTIONAL,
        roadRoadType                    ENUMERATED{
                nationalExpressway(1),cityHighway(2),generalTollRoad(3),
                theWayOnlyForCars(4),generalNationalHighwaySpecifiedSection(5),
                generalNationalHighwayOutsideSpecifiedSection(6),mainDistrictWay(7),
                generalAllPrefecturesWay(8),municipalRoad(9),other(98),invalidData(99)},
        roadGeneralNationalHighwayClassification
                                        ENUMERATED{
                unknown(0),curRoad(1),oldRoad(2),newRoad(3),researching(4)}
                                                                OPTIONAL,
        roadRouteDirectionCode          ENUMERATED{
                none(0), noDirection(1), up(2), down(3), inbound(4), outbound(5),
                upInbound(6), downOutbound(7), updown(8), eastbound(9), westbound(10),
                northbound(11), southbound(12), bothDirections(13), upAnotherLane(14),
                downAnotherLane(15), upLeft(16), downLeft(17), upRight(18), downRight(19),
                upBothRoutes(20), downBothRoutes(21), inboundLeft(22), outboundLeft(23),
```

```
                inboundRight(24), outboundRight(25), invalidData(97),other(98), unknown(99)},
        locationNationalRoadSpotKp      INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsRoadLinkSpot::=SEQUENCE{
        dsRoadLinkSection               DsRoadLinkSection       ,
        locationLinkEndDistance         INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsLocationCode::=SEQUENCE{
        dsLocationCode0                 DsLocationCode0         ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }
DsLocationCode0::=CHOICE{
        roadIcCode                      UTF8String              ,
        facilitySapaAreaLocationNumber  UTF8String              ,
        roadIntersectionNumber          INTEGER
        }

DsBaseRoadLinkSpot::=SEQUENCE{
        dsBaseRoadLinkSection           DsBaseRoadLinkSection   ,
        locationLinkEndDistance         INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsLocalRoadSpotKp::=SEQUENCE{
        roadNationalwayRouteNumber      INTEGER                 OPTIONAL,
        roadNationalwayRouteSubCode     INTEGER                 OPTIONAL,
        roadOrganization                DsOrganization          OPTIONAL,
        roadRouteName                   UTF8String              OPTIONAL,
        roadRoadType                    ENUMERATED{
                nationalExpressway(1),cityHighway(2),generalTollRoad(3),
                theWayOnlyForCars(4),generalNationalHighwaySpecifiedSection(5),
                generalNationalHighwayOutsideSpecifiedSection(6),mainDistrictWay(7),
                generalAllPrefecturesWay(8),municipalRoad(9),other(98),invalidData(99)},
```

```
        roadGeneralNationalHighwayClassification
                                        ENUMERATED{
                unknown(0),curRoad(1),oldRoad(2),newRoad(3),researching(4)}
                                                        OPTIONAL,
        roadRouteDirectionCode          ENUMERATED{
                none(0), noDirection(1), up(2), down(3), inbound(4), outbound(5),
                upInbound(6), downOutbound(7), updown(8), eastbound(9), westbound(10),
                northbound(11), southbound(12), bothDirections(13), upAnotherLane(14),
                downAnotherLane(15), upLeft(16), downLeft(17), upRight(18), downRight(19),
                upBothRoutes(20), downBothRoutes(21), inboundLeft(22), outboundLeft(23),
                inboundRight(24), outboundRight(25), invalidData(97),other(98), unknown(99)},
        locationLocalRoadSpotKp         INTEGER         OPTIONAL,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

DsRoadSpotTwoDimensional::=SEQUENCE{
        dsLocationRoadPosition          DsLocationRoadPosition  ,
        roadLaneType                    ENUMERATED{
                berm(0),upRunningLane(1),oneRunningLane(2),twoRunningLane(3),
                threeRunningLane(4),fourRunningLane(5),fiveRunningLane(6),
                sixRunningLane(7),overtakingLane(8),allLanes(9),others(10),spare(11),
                sevenRunningLane(12),eightRunningLane(13),invalidData(99)},
        locationOffset                  INTEGER         ,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

DsCoordinatesLocation::=CHOICE{
        dsNormalCoordinates             DsNormalCoordinates     ,
        dsLatitudeLongitude             DsLatitudeLongitude     ,
        locationLatitudeLongitudeDegree LocationLatitudeLongitudeDegree
        }

DsNormalCoordinates::=SEQUENCE{
        locationXYNormalCoordinates     LocationXYNormalCoordinates,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }
```

```
DsLatitudeLongitude::=SEQUENCE{
        locationLatitudeDegree          INTEGER             ,
        locationLatitudeMinute          INTEGER             ,
        locationLatitudeSecond          INTEGER             ,
        locationLongitudeDegree         INTEGER             ,
        locationLongitudeMinute         INTEGER             ,
        locationLongitudeSecond         INTEGER             ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsRoadSpotThreeDimensional::=SEQUENCE{
        dsRoadSpotTwoDimensional        DsRoadSpotTwoDimensional ,
        locationAltitude                LocationAltitude    ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

LocationAltitude::=CHOICE{
        locationAltitudeGround          INTEGER             ,
        locationAltitudeSea             INTEGER
        }

DsCoordinatesLocationThreeDimensional::=SEQUENCE{
        dsCoordinatesLocation           DsCoordinatesLocation ,
        locationAltitude                LocationAltitude    ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsAddress::=SEQUENCE{
        locationMunicipalityCode        INTEGER             ,
        locationHouseNumber             UTF8String          ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsSectionLocation ::=CHOICE{
        dsStartEndSection               DsStartEndSection   ,
        dsRoadLinkSection               DsRoadLinkSection   ,
        dsSpanSection                   DsSpanSection       ,
```

```
        dsBaseRoadLinkSection           DsBaseRoadLinkSection
        }

DsStartEndSection::=SEQUENCE{
        startPointLocation              DsPointLocation         ,
        endPointLocation                DsPointLocation         ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsRoadLinkSection::=SEQUENCE{
        locationSecondaryCoordinatesCode    INTEGER             ,
        locationLinkLayer               ENUMERATED{
                narrowArea(1),mediumArea(2),wideArea(3),invalidData(9)}  ,
        locationLinkEntry               ENUMERATED{
                expressway(0),urbanExpressway(1),ordinaryRoad(2),others(3),invalidData(9)},
        locationLinkNumber              INTEGER                 ,
        locationLinkVersion             INTEGER                 OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsSpanSection::=SEQUENCE{
        locationSpanCode                INTEGER                 ,
        locationCourseDistance          INTEGER                 ,
        locationSpanDistance            INTEGER                 ,
        roadRouteName                   UTF8String              ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsBaseRoadLinkSection::=SEQUENCE{
        roadRouteDirectionCode          ENUMERATED{
                none(0), noDirection(1), up(2), down(3), inbound(4), outbound(5),
                upInbound(6), downOutbound(7), updown(8), eastbound(9), westbound(10),
                northbound(11), southbound(12), bothDirections(13), upAnotherLane(14),
                downAnotherLane(15), upLeft(16), downLeft(17), upRight(18), downRight(19),
                upBothRoutes(20), downBothRoutes(21), inboundLeft(22), outboundLeft(23),
                inboundRight(24), outboundRight(25), invalidData(97),other(98), unknown(99)},
        locationSecondaryCoordinatesCode    INTEGER             ,
```

```
        locationBaseRoadLinkNumber      INTEGER                 ,
        locationBaseRoadLinkVersion     INTEGER                 OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsAreaLocation ::=CHOICE{
        locationRegionCode0             LocationRegionCode0     ,
        locationLifeAreaCode0           LocationLifeAreaCode0   ,
        locationMunicipalityCode0       LocationMunicipalityCode0 ,
        locationBZoneCode0              LocationBZoneCode0      ,
        locationCZoneCode0              LocationCZoneCode0      ,
        locationWideArea0               LocationWideArea0
        }

LocationRegionCode0::=SEQUENCE{
        locationRegionCode              ENUMERATED{
                hokkaido(1),aomoriPrefecture(2),iwatePrefecture(3),miyagiPrefecture(4),
                akitaPrefecture(5),yamagataPrefecture(6),fukushimaPrefecture(7),
                ibaragiPrefecture(8),tochigiPrefecture(9),gunmaPrefecture(10),
                saitamaPrefecture(11),chibaPrefecture(12),tokyo(13),kanagawaPrefecture(14),
                niigataPrefecture(15),toyamaPrefecture(16),ishikawaPrefecture(17),
                fukuiPrefecture(18),yamanashiPrefecture(19),naganoPrefecture(20),
                gifuPrefecture(21),shizuokaPrefecture(22),aichiPrefecture(23),miePrefecture(24),
                shigaPrefecture(25),kyotoPrefecture(26),osakaPrefecture(27),
                hyogoPrefecture(28),naraPrefecture(29),wakayamaPrefecture(30),
                tottoriPrefecture(31),shimanePrefecture(32),okayamaPrefecture(33),
                hiroshimaPrefecture(34),yamaguchiPrefecture(35),tokushimaPrefecture(36),
                kagawaPrefecture(37),ehimePrefecture(38),kochiPrefecture(39),
                fukuokaPrefecture(40),sagaPrefecture(41),nagasakiPrefecture(42),
                kumamotoPrefecture(43),oitaPrefecture(44),miyazakiPrefecture(45),
                kagoshimaPrefecture(46),okinawaPrefecture(47),invalidData(99)},
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

LocationLifeAreaCode0::=SEQUENCE{
        locationLifeAreaCode            INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
```

```
        }

LocationMunicipalityCode0::=SEQUENCE{
        locationMunicipalityCode        INTEGER         ,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

LocationBZoneCode0::=SEQUENCE{
        locationBZoneCode               INTEGER         ,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

LocationCZoneCode0::=SEQUENCE{
        locationCZoneCode               INTEGER         ,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

LocationWideArea0::=SEQUENCE{
        locationWideArea                INTEGER         ,
        relationLinkageIdentifier       INTEGER         OPTIONAL
        }

DsRouteLocation ::=CHOICE{
        dsPointRouteLocation            DsPointRouteLocation        ,
        dsLinkRouteLocation             DsLinkRouteLocation         ,
        dsSpanRouteLocation             DsSpanRouteLocation         ,
        dsBaseRoadLinkRouteLocation     DsBaseRoadLinkRouteLocation
        }

DsPointRouteLocation ::=SEQUENCE{
        startPointLocation              DsPointLocation             ,
        detourPointLocation             SEQUENCE OF DsPointLocation,
        endPointLocation                DsPointLocation             ,
        routeCourseDistance             INTEGER                     ,
        routeNumber                     INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }
```

```
DsLinkRouteLocation ::=SEQUENCE{
        startLink                       DsRoadLinkSection           ,
        endLink                         DsRoadLinkSection           ,
        detourLink                      SEQUENCE OF DsRoadLinkSection,
        routeLinkFigure                 INTEGER                     ,
        locationCourseDistance          INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsSpanRouteLocation ::=SEQUENCE{
        startSpan                       DsRoadLinkSection           ,
        endSpan                         DsRoadLinkSection           ,
        locationSpanDistance            INTEGER                     ,
        locationSpanNumber              OCTET STRING                ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsBaseRoadLinkRouteLocation ::=SEQUENCE{
        startBaseRoadLink               DsBaseRoadLinkSection       ,
        endBaseRoadLink                 DsBaseRoadLinkSection       ,
        detourBaseRoadLink              SEQUENCE OF DsBaseRoadLinkSection,
        routeLinkFigure                 INTEGER                     ,
        locationCourseDistance          INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsOrganization::=SEQUENCE {
        organizationAgencyCode          ENUMERATED{
                jh(10),jhe(11),jhc(12),jhw(13),mex(20),hex(30),police(40),honshi(50),mlit(60),
                kosha(70),jichitai(80),invalidData(98),other(99)}   OPTIONAL,
        organizationAgencyName          UTF8String                  OPTIONAL,
        organizationOrganizationCode    UTF8String                  OPTIONAL,
        organizationOrganizationName    UTF8String                  OPTIONAL,
        organizationBureauCode          UTF8String                  OPTIONAL,
        organizationBureauName          UTF8String                  OPTIONAL,
        organizationDivisionCode        INTEGER                     OPTIONAL,
        organizationDivisionName        UTF8String                  OPTIONAL,
```

```
organizationPersonName          UTF8String              OPTIONAL,
organizationAddress             UTF8String              OPTIONAL,
organizationTelNum              UTF8String              OPTIONAL,
organizationFaxNum              UTF8String              OPTIONAL,
organizationEmail               OCTET STRING            OPTIONAL,
organizationVehicleBureauCode   ENUMERATED {
        sapporo(1),hakodate(2),muroran(3),obihiro(4),kushiro(5),kitami(6),
        asahikawa(7),miyagi(101),fukushima(102),iwaki(103),iwate(104),aomori(105),
        hachinohe(106),yamagata(107),syounai(108),akita(109),shinagawa(201),
        adachi(202),nerima(203),tama(204),hachioji(205),yokohama(206),sagami(207),
        kawasaki(208),shonan(209),chiba(210),narashino(211),sodegaura(212),
        omiya(213),kumagaya(214),tokorozawa(215),mito(216),tsuchiura(217),
        gunma(218),tochigi(219),yamanashi(220),nagano(221),matsumoto(222),
        kasukabe(223),niigata(301),nagaoka(302),ishikawa(303),toyama(304),
        shizuoka(401),hamamatsu(402),numazu(403),gifu(404),mie(405),nagoya(406),
        mikawa(407),owariKomaki(408),toyohashi(409),fukui(501),osaka(502),
        izumi(503),kyoto(504),kobe(505),himeji(506),shiga(507),nara(508),
        wakayama(509),naniwa(510),hiroshima(601),tottori(602),shimane(603),
        okayama(604),yamaguchi(605),fukuyama(606),kagawa(701),tokushima(702),
        ehime(703),kochi(704),fukuoka(801),kitakyushu(802),kurume(803),
        chikuho(804),saga(805),nagasaki(806),sasebo(807),shimabara(808),
        kumamoto(809),oita(810),miyazaki(811),kagoshima(812),oshima(813),
        naha(901),miyako(902),yaeyama(903),dummyNumber(998),
        invalidData(999)}                               OPTIONAL,
organizationWeatherOrganizationCode  ENUMERATED{
        districtMeteorologicalObservatory(1),marineObservatory(2),
        localMeteorologicalObservatory(3),specifiedLocalMeteorologicalStation(4),
        aviationMeteorologicalStation(5),
        localMeteorologicalStationEstablishedToAviation(6),branchOfficeOfAirPort(7),
        radarObservationStation(8),publicMeteorologicalDivision(9),
        airportMeteorologicalRadar(10),invalidData(99)}  OPTIONAL,
locationRegionCode              ENUMERATED{
        hokkaido(1),aomoriPrefecture(2),iwatePrefecture(3),
        miyagiPrefecture(4),akitaPrefecture(5),yamagataPrefecture(6),
        fukushimaPrefecture(7),ibaragiPrefecture(8),tochigiPrefecture(9),
        gunmaPrefecture(10),saitamaPrefecture(11),chibaPrefecture(12),tokyo(13),
        kanagawaPrefecture(14),niigataPrefecture(15),toyamaPrefecture(16),
```

```
                ishikawaPrefecture(17),fukuiPrefecture(18),yamanashiPrefecture(19),
                naganoPrefecture(20),gifuPrefecture(21),shizuokaPrefecture(22),
                aichiPrefecture(23),miePrefecture(24),shigaPrefecture(25),kyotoPrefecture(26),
                osakaPrefecture(27),hyogoPrefecture(28),naraPrefecture(29),
                wakayamaPrefecture(30),tottoriPrefecture(31),shimanePrefecture(32),
                okayamaPrefecture(33),hiroshimaPrefecture(34),yamaguchiPrefecture(35),
                tokushimaPrefecture(36),kagawaPrefecture(37),ehimePrefecture(38),
                kochiPrefecture(39),fukuokaPrefecture(40),sagaPrefecture(41),
                nagasakiPrefecture(42),kumamotoPrefecture(43),oitaPrefecture(44),
                miyazakiPrefecture(45),kagoshimaPrefecture(46),okinawaPrefecture(47),
                invalidData(99)}                                    OPTIONAL,
        locationMunicipalityCode        INTEGER                     OPTIONAL,
        organizationRoadAdministrationInsideIdentifierNum
                                        INTEGER                     OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsMovableBodyInfo::=DsVehicleNumber

DsVehicleNumber ::= SEQUENCE{
        organizationVehicleBureauCode       ENUMERATED    {
                sapporo(1),hakodate(2),muroran(3),obihiro(4),kushiro(5),kitami(6),
                asahikawa(7),miyagi(101),fukushima(102),iwaki(103),iwate(104),aomori(105),
                hachinohe(106),yamagata(107),syounai(108),akita(109),shinagawa(201),
                adachi(202),nerima(203),tama(204),hachioji(205),yokohama(206),sagami(207),
                kawasaki(208),shonan(209),chiba(210),narashino(211),sodegaura(212),
                omiya(213),kumagaya(214),tokorozawa(215),mito(216),tsuchiura(217),
                gunma(218),tochigi(219),yamanashi(220),nagano(221),matsumoto(222),
                kasukabe(223),niigata(301),nagaoka(302),ishikawa(303),toyama(304),
                shizuoka(401),hamamatsu(402),numazu(403),gifu(404),mie(405),nagoya(406),
                mikawa(407),owariKomaki(408),toyohashi(409),fukui(501),osaka(502),
                izumi(503),kyoto(504),kobe(505),himeji(506),shiga(507),nara(508),
                wakayama(509),naniwa(510),hiroshima(601),tottori(602),shimane(603),
                okayama(604),yamaguchi(605),fukuyama(606),kagawa(701),tokushima(702),
                ehime(703),kochi(704),fukuoka(801),kitakyushu(802),kurume(803),
                chikuho(804),saga(805),nagasaki(806),sasebo(807),shimabara(808),
                kumamoto(809),oita(810),miyazaki(811),kagoshima(812),oshima(813),
```

```
            naha(901),miyako(902),yaeyama(903),dummyNumber(998),invalidData(999)},
        movableBodyCategoryNumber       INTEGER                 ,
        movableBodyKanaCharacter        UTF8String              ,
        movableBodySequentialNumber     INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsTrafficVolumeData ::= CHOICE{
        dsTrafficVolume                 DsTrafficVolume         ,
        dsPassingVehicle                DsPassingVehicle        ,
        dsAhsVehicleInfo                DsAhsVehicleInfo
        }

DsTrafficVolume ::= SEQUENCE OF DsTrafficVolumeByLane
DsTrafficVolumeByLane ::= SEQUENCE{
        deviceCollectedDataStatus       ENUMERATED{
            detailUnknown(0),normal(1),extraordinary(2),invalidData(9)}
                                                                OPTIONAL,
        deviceLocation                  DsLocationInfo          ,
        calculationTotalTrafficVolume   CalculationTrafficVolume ,
        typeTrafficVolume               DsTypeTrafficVolume     ,
        calculationSpotAverageSpeed     INTEGER                 ,
        calculationOccupancy            INTEGER                 ,
        deviceStatus                    ENUMERATED{
            oK(0),ultrasonicAbnormal(1),loopAbnormal(2),laserAbnormal(3),
            imageAbnormal(4),processingUnitAbnormal(5),invalidData(9)}
                                                                OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        dateTime                        DsDateTime              ,
        confirmTotalTrafficVolume       DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        confirmSpotAverageSpeed         DeviceControlAnswerConfirmInfo
                                                                OPTIONAL,
        confirmOccupancy                DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }
```

```
DsPassingVehicle ::= SEQUENCE OF DsPassingVehicleByLane
DsPassingVehicleByLane ::= SEQUENCE{
        calculationPassingVehicleIndex      INTEGER                 ,
        deviceLocation                      DsLocationInfo          ,
        dateTime                            DsDateTime              ,
        eventDeterminationOfVehicleHeight   ENUMERATED{
                outOfScale(0),moreThan2m(1),from1Dot5mTo2Dot0m(2),lessThan1Dot5m(3),
                undeterminable(4),invalidData(9)}                   OPTIONAL,
        calculationVehicleHeightMeasurementResult
                                            INTEGER                 OPTIONAL,
        eventDeterminationOfVehicleLength   ENUMERATED{
                outOfScale(0),moreThan4Dot75m(1),lessThan4Dot75m(2),undeterminable(3),
                invalidData(9)}                                     OPTIONAL,
        calculationVehicleLengthMeasurementResult
                                            INTEGER                 OPTIONAL,
        eventDeterminationOfVehicleLengthBetweenGroundAndBottom
                                            ENUMERATED{
                outOfScale(0),high(1),low(2),undeterminable(3),invalidData(9)}
                                                                    OPTIONAL,
        calculationVehiclewidthMeasurementResult
                                            INTEGER                 OPTIONAL,
        calculationVehicleForm              BIT STRING              OPTIONAL,
        calculationVehicleNetWeightMeasurementResult
                                            INTEGER                 OPTIONAL,
        calculationAxisNumberMeasurementResult
                                            INTEGER                 OPTIONAL,
        vehicleNumber                       DsVehicleNumber         OPTIONAL,
        calculationCarTypeMeasurementResult ENUMERATED{
                minicar(0),standardSizedCar(1),bus(2),lightLoads(3),smallTruck(4),
                goodsAndPassengersVehicle(5),ordinaryTruck(6),specialUsesVehicle(7),
                undeterminable(8),invalidData(99)},
        calculationSpotSpeed                INTEGER                 OPTIONAL,
        calculationRelativeSpeed            INTEGER                 OPTIONAL,
        calculationVehicularGap             INTEGER                 OPTIONAL,
        deviceStatus                        ENUMERATED{
                oK(0),ultrasonicAbnormal(1),loopAbnormal(2),laserAbnormal(3),
                imageAbnormal(4),processingUnitAbnormal(5),invalidData(9)}
```

```
                                                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsAhsVehicleInfo::= SEQUENCE {
        vehicleLocation                 DsLocationInfo              ,
        dateTime                        DsDateTime                  ,
        calculationAhsVehicleSpeed      INTEGER                     ,
        calculationAhsLocationTrafficLanes  INTEGER                 ,
        calculationAhsVehicleLength     INTEGER                     ,
        calculationAhsVehicleType       ENUMERATED{
                largeVehicle(1),ordinaryVehicle(2),workVehicle(3),other(4),invalidData(9)},
        calculationVehicleIDMeasurementResult
                                        INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

DsTypeTrafficVolume ::= CHOICE{
        trafficVolumeVehicleType1       TrafficVolumeVehicleType1   ,
        trafficVolumeVehicleType2       TrafficVolumeVehicleType2   ,
        trafficVolumeVehicleType4       TrafficVolumeVehicleType4   ,
        trafficVolumeVehicleType11      TrafficVolumeVehicleType11
        }

TrafficVolumeVehicleType1::= SEQUENCE{
        trafficVolumeVehicleType1       CalculationTrafficVolume    ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

TrafficVolumeVehicleType2 ::= SEQUENCE{
        trafficVolumeVehicleType2Large      CalculationTrafficVolume    ,
        trafficVolumeVehicleType2Small      CalculationTrafficVolume    ,
        trafficVolumeVehicleType2Unknown    CalculationTrafficVolume    ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

TrafficVolumeVehicleType4 ::= SEQUENCE{
```

```
        trafficVolumeVehicleType4LargeCargo   CalculationTrafficVolume     ,
        trafficVolumeVehicleType4Bus          CalculationTrafficVolume     ,
        trafficVolumeVehicleType4SmallCargo   CalculationTrafficVolume     ,
        trafficVolumeVehicleType4PassengerCar CalculationTrafficVolume     ,
        trafficVolumeVehicleType4Unknown      CalculationTrafficVolume     ,
        relationLinkageIdentifier             INTEGER                      OPTIONAL
        }

TrafficVolumeVehicleType11 ::= SEQUENCE{
        trafficVolumeVehicleType11Pedestrian  CalculationTrafficVolume     ,
        trafficVolumeVehicleType11Bicycle     CalculationTrafficVolume     ,
        trafficVolumeVehicleType11AutoBike    CalculationTrafficVolume     ,
        trafficVolumeVehicleType11LightPassengerCar
                                              CalculationTrafficVolume     ,
        trafficVolumeVehicleType11PassengerCar
                                              CalculationTrafficVolume     ,
        trafficVolumeVehicleType11Bus         CalculationTrafficVolume     ,
        trafficVolumeVehicleType11LightCargo  CalculationTrafficVolume     ,
        trafficVolumeVehicleType11SmallCargo  CalculationTrafficVolume     ,
        trafficVolumeVehicleType11CargoPassengerCar
                                              CalculationTrafficVolume     ,
        trafficVolumeVehicleType11NormalCargo
                                              CalculationTrafficVolume     ,
        trafficVolumeVehicleType11SpecialTruck
                                              CalculationTrafficVolume     ,
        trafficVolumeVehicleType11Unknown     CalculationTrafficVolume     ,
        relationLinkageIdentifier             INTEGER                      OPTIONAL
        }

CalculationTrafficVolume::= CHOICE{
        calculationTrafficVolume1min          CalculationTrafficVolume1m   ,
        calculationTrafficVolume5min          CalculationTrafficVolume5m   ,
        calculationTrafficVolume10min         CalculationTrafficVolume10m  ,
        calculationTrafficVolume60min         CalculationTrafficVolume60m  ,
        calculationTrafficVolumeTotal         CalculationTotalTrafficVolume
        }
```

```
CalculationTrafficVolume1m::=SEQUENCE{
        calculationTrafficVolume1m      INTEGER         ,
        dateTimeStart                   DsDateTime      OPTIONAL,
        dateTimeEnd                     DsDateTime      OPTIONAL
        }

CalculationTrafficVolume5m::=SEQUENCE{
        calculationTrafficVolume5m      INTEGER         ,
        dateTimeStart                   DsDateTime      OPTIONAL,
        dateTimeEnd                     DsDateTime      OPTIONAL
        }

CalculationTrafficVolume10m::=SEQUENCE{
        calculationTrafficVolume10m     INTEGER         ,
        dateTimeStart                   DsDateTime      OPTIONAL,
        dateTimeEnd                     DsDateTime      OPTIONAL
        }

CalculationTrafficVolume60m::=SEQUENCE{
        calculationTrafficVolume60m     INTEGER         ,
        dateTimeStart                   DsDateTime      OPTIONAL,
        dateTimeEnd                     DsDateTime      OPTIONAL
        }

CalculationTotalTrafficVolume::=SEQUENCE{
        calculationTotalTrafficVolume   INTEGER         ,
        dateTimeStart                   DsDateTime      OPTIONAL,
        dateTimeEnd                     DsDateTime      OPTIONAL
        }

DsEnvironmentMonitoringData ::= CHOICE{
        dsWeather                       DsWeather       ,
        dsRoadSituation                 DsRoadSituation ,
        dsSituationInTunnel             DsSituationInTunnel ,
        dsRouteEnvironment              DsRouteEnvironment ,
        dsNaturalDisaster               DsNaturalDisaster ,
        dsRockMeasurement               DsRockMeasurement
```

```
        }

DsWeather ::= CHOICE{
        Temperature                 Temperature                 ,
        humidity                    Humidity                    ,
        rainfall                    Rainfall                    ,
        snowfall                    Snowfall                    ,
        precipitation               Precipitation               ,
        snowAmount                  SnowAmount                  ,
        ashfallAmount               AshfallAmount               ,
        wind                        Wind                        ,
        earthquake                  Earthquake                  ,
        seaLevel                    SeaLevel                    ,
        waveHeight                  WaveHeight                  ,
        visibility                  Visibility                  ,
        radiation                   Radiation                   ,
        dayLightTime                DayLightTime                ,
        netRadiation                NetRadiation
        }

Temperature ::= SEQUENCE{
        temperature0                Temperature0                ,
        relationLinkageIdentifier   INTEGER                     OPTIONAL,
        pointLocation               DsPointLocation             OPTIONAL,
        dateTime                    DsDateTime                  OPTIONAL,
        sensorID                    DsSensorID                  OPTIONAL,
        confirmTemperature          DeviceControlAnswerConfirmInfo
                                                                OPTIONAL

        }

Temperature0 ::= CHOICE{
        calculationTemperature              INTEGER             ,
        calculationTemperatureHighQuality   INTEGER             ,
        calculationAhsTemperature           INTEGER             ,
        calculationHighestTemperature       INTEGER             ,
        calculationLowestTemperature        INTEGER
        }
```

```
DsSensorID ::= SEQUENCE{
        dsOrganization                  DsOrganization          ,
        deviceManagementNumber          UTF8String              ,
        deviceSubManagementNumber       UTF8String              OPTIONAL
        }

Humidity ::= SEQUENCE{
        calculationHumidity             INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmHumidity                 DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

Rainfall ::= SEQUENCE{
        rainfallAmount0                 RainfallAmount0         ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmRainfall                 DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }
RainfallAmount0 ::= CHOICE{
        calculationRainfallAmountIn5Minutes     INTEGER         ,
        calculationRainfallAmountIn10Minutes    INTEGER         ,
        calculationHourlyRainfallAmount         INTEGER         ,
        calculationContinuousRainfallAmount     INTEGER         ,
        calculationEffectiveRainfallAmount      INTEGER         ,
        calculationCurrentDayRainfallAmount     INTEGER         ,
        calculationRainfallPerDay               INTEGER         ,
        calculationContinuousRainfallAmountEveryTenMinutes
                                                INTEGER
        }
```

```
Snowfall ::= SEQUENCE{
        snowfallAmount0                 SnowfallAmount0                 ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmSnowfall                 DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }
SnowfallAmount0 ::= CHOICE{
        calculationSnowfallAmount       INTEGER                         ,
        calculationSnowfallAmountHour   INTEGER                         ,
        calculationSnowfall             INTEGER                         ,
        calculationSnowfallAmountDayFrom21 INTEGER                      ,
        calculationSnowfallAmountDayFrom9 INTEGER                       ,
        calculationCumulativeSnowfall   INTEGER                         ,
        calculationSnowAmount           INTEGER
        }

Precipitation   ::= SEQUENCE{
        precipitation0                  Precipitation0                  ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmPrecipitation            DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

Precipitation0 ::= CHOICE{
        calculationPrecipitation        INTEGER                         ,
        calculationHourlyPrecipitation  INTEGER                         ,
        calculationPrecipitation3Hours  INTEGER                         ,
        calculationPrecipitation6Hours  INTEGER                         ,
        calculationPrecipitation12Hours INTEGER                         ,
        calculationPrecipitation24Hours INTEGER                         ,
```

```
        calculationDailyPrecipitation           INTEGER                 ,
        calculationCumulativePrecipitation      INTEGER                 ,
        calculationProbabilityOfPrecipitation   INTEGER                 ,
        calculationPrecipitation10MinutesHighQuality
                                                INTEGER                 ,
        calculationAhsRainfallAmountIn10Minutes
                                                INTEGER
        }

SnowAmount ::= SEQUENCE{
        snowAmount0                             SnowAmount0             ,
        relationLinkageIdentifier               INTEGER                 OPTIONAL,
        pointLocation                           DsPointLocation         OPTIONAL,
        dateTime                                DsDateTime              OPTIONAL,
        sensorID                                DsSensorID              OPTIONAL,
        confirmSnowAmount                       DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

SnowAmount0 ::= CHOICE{
        calculationSnowAmount                   INTEGER                 ,
        calculationAhsRoadSnowAmount            INTEGER
        }

AshfallAmount::= SEQUENCE{
        calculationAshfallAmount                INTEGER                 ,
        relationLinkageIdentifier               INTEGER                 OPTIONAL,
        pointLocation                           DsPointLocation         OPTIONAL,
        dateTime                                DsDateTime              OPTIONAL,
        sensorID                                DsSensorID              OPTIONAL,
        confirmAshfallAmount                    DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

Wind ::= SEQUENCE{
        calculation16WindDirections             ENUMERATED {
                n(1),nne(2),ne(3),ene(4),e(5),se(6),ese(7),sse(8),s(9),ssw(10),sw(11),wsw(12),w(13),
```

```
                wnw(14),nw(15),nnw(16),calm(17),invalidData(99)},
        instantaneousWindSpeed          InstantaneousWindSpeed      OPTIONAL,
        calculationMaxInstantaneousWindSpeed
                                        INTEGER                     OPTIONAL,
        averageWindSpeed                AverageWindSpeed            ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
        pointLocation                   DsPointLocation             OPTIONAL,
        dateTime                        DsDateTime                  OPTIONAL,
        sensorID                        DsSensorID                  OPTIONAL,
        confirmWind                     DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }
InstantaneousWindSpeed ::= CHOICE{
        calculationInstantaneousWindSpeed5Minutes
                                        INTEGER                     ,
        calculationInstantaneousWindSpeed10Minutes
                                        INTEGER                     ,
        calculationInstantaneousWindSpeed5MinutesHighQuality
                                        INTEGER                     ,
        calculationInstantaneousWindSpeed10MinutesHighQuality
                                        INTEGER                     ,
        calculationInstantaneousWindSpeed1MinutesHighQuality
                                        INTEGER
        }
AverageWindSpeed ::= CHOICE{
        calculationAverageWindSpeed     INTEGER                     ,
        calculationAverageWindSpeedOverAMinute
                                        INTEGER                     ,
        calculationAverageWindSpeedOver10Minutes
                                        INTEGER                     ,
        calculationAhsAverageWindSpeedOver10Minutes
                                        INTEGER
        }

Earthquake ::= SEQUENCE{
        calculationAcceleration         INTEGER                     ,
        calculationSiValue              INTEGER                     ,
```

```
        calculationMaximumHorizontalAcceleration
                                        INTEGER             ,
        calculationMaximumVerticalAcceleration
                                        INTEGER             ,
        calculationAccelerationSpeedResponseValue
                                        INTEGER             ,
        eventMeasuredScaleOfAnEarthquake INTEGER            ,
        relationLinkageIdentifier       INTEGER             OPTIONAL,
        pointLocation                   DsPointLocation     OPTIONAL,
        dateTime                        DsDateTime          OPTIONAL,
        sensorID                        DsSensorID          OPTIONAL,
        confirmEarthquake               DeviceControlAnswerConfirmInfo
                                                            OPTIONAL
        }

SeaLevel::= SEQUENCE{
        calculationSeaLevel             INTEGER             ,
        relationLinkageIdentifier       INTEGER             OPTIONAL,
        pointLocation                   DsPointLocation     OPTIONAL,
        dateTime                        DsDateTime          OPTIONAL,
        sensorID                        DsSensorID          OPTIONAL,
        confirmSeaLevel                 DeviceControlAnswerConfirmInfo
                                                            OPTIONAL
        }

WaveHeight::= SEQUENCE{
        calculationWaveHeight           INTEGER             ,
        relationLinkageIdentifier       INTEGER             OPTIONAL,
        pointLocation                   DsPointLocation     OPTIONAL,
        dateTime                        DsDateTime          OPTIONAL,
        sensorID                        DsSensorID          OPTIONAL,
        confirmWaveHeight               DeviceControlAnswerConfirmInfo
                                                            OPTIONAL
        }

Visibility ::= SEQUENCE{
        calculationSnowfallFogDensity   INTEGER             OPTIONAL,
```

```
        calculationVisibility           INTEGER                         ,
        calculationTransmittance        INTEGER                         OPTIONAL,
        calculationTransmissivity       INTEGER                         OPTIONAL,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmVisibility               DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        confirmSnowfallFogDensity       DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        confirmTransmittance            DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL,
        confirmTransmissivity           DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

Radiation ::= SEQUENCE{
        calculationSolarRadiationPenetration    INTEGER                 ,
        calculationSolarRadiationTotal  INTEGER                         OPTIONAL,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRadiation                DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

DayLightTime ::= SEQUENCE{
        calculationSunshineHour         INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmDayLightTime             DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }
```

```
NetRadiation ::= SEQUENCE{
        calculationNetRadiation         INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmNetRadiation             DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

DsRoadSituation ::= CHOICE{
        roadSurfaceCondition            RoadSurfaceCondition            ,
        roadTemperature                 RoadTemperature                 ,
        roadMoisture                    RoadMoisture                    ,
        roadSurfaceReflectionCoefficient RoadSurfaceReflectionCoefficient,
        waterLevel                      WaterLevel
        }

RoadSurfaceCondition::= SEQUENCE{
        roadSurfaceCondition0           RoadSurfaceCondition0           ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRoadSurfaceCondition     DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

RoadSurfaceCondition0::=CHOICE{
        eventRoadSurfaceConditions      ENUMERATED{
                frozen(1),accumulationOfSnow(2),damp(3),filmOfWater(4),dry(5),invalidData(9)},
        eventAhsRoadSurfaceConditions13Details
                                        ENUMERATED{
                verySlipperyThickSnow(1),verySlipperyIceSheet(2),verySlipperyIceFilm(3),
                iceSheet(4),iceFilm(5),iceSheetUnderPowerSnow(6),
                iceSheetUnderGranularSnow(7),thickSnow(8),powderSnow(9),granularSnow(10),
```

```
                sherbet(11),wet(12),dry(13),invalidData(99)}                ,
        eventAhsRoadSurfaceConditions7      ENUMERATED{
                verySlipperyRoad(1),iceRink(2),thickSnow(3),powerAndGranularSnow(4),
                sherbet(5),wet(6),dry(7),invalidData(9)}                    ,
        eventAhsRoadSurfaceConditions5      ENUMERATED{
                frozen(1),snow(2),damp(3),waterFilm(4),dry(5),invalidData(9)}
        }

RoadTemperature::= SEQUENCE{
        roadTemperature0                    RoadTemperature0        ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL,
        pointLocation                       DsPointLocation         OPTIONAL,
        dateTime                            DsDateTime              OPTIONAL,
        sensorID                            DsSensorID              OPTIONAL,
        confirmRoadTemperature              DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

RoadTemperature0::=CHOICE{
        calculationRoadTemperature          INTEGER                 ,
        calculationAhsRoadTemperature       INTEGER
        }

RoadMoisture::= SEQUENCE{
        calculationRoadMoisture             INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL,
        pointLocation                       DsPointLocation         OPTIONAL,
        dateTime                            DsDateTime              OPTIONAL,
        sensorID                            DsSensorID              OPTIONAL,
        confirmRoadMoisture                 DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

RoadSurfaceReflectionCoefficient::= SEQUENCE{
        calculationRoadSurfaceReflectionCoefficient
                                            INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL,
```

```
        pointLocation                   DsPointLocation             OPTIONAL,
        dateTime                        DsDateTime                  OPTIONAL,
        sensorID                        DsSensorID                  OPTIONAL,
        confirmRoadSurfaceReflectionCoefficient
                                        DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

WaterLevel::= SEQUENCE{
        calculationWaterLevel           INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
        pointLocation                   DsPointLocation             OPTIONAL,
        dateTime                        DsDateTime                  OPTIONAL,
        sensorID                        DsSensorID                  OPTIONAL,
        confirmWaterLevel               DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

DsSituationInTunnel ::= CHOICE{
        airTransmissivityInsideTheTunnel    AirTransmissivityInsideTheTunnel,
        windSpeedInsideTheTunnel        WindSpeedInsideTheTunnel    ,
        tunnelFireDetection             TunnelFireDetection         ,
        brightness                      Brightness                  ,
        tunnelAnchorLoad                TunnelAnchorLoad            ,
        tunnelSlopeOfGround             TunnelSlopeOfGround         ,
        tunnelStressDisplacement        TunnelStressDisplacement    ,
        tunnelCrackedDisplacement       TunnelCrackedDisplacement   ,
        tunnelSlopeOfPit                TunnelSlopeOfPit            ,
        tunnelVerticalElastic           TunnelVerticalElastic       ,
        tunnelVerticalGround            TunnelVerticalGround        ,
        tunnelTemperature               TunnelTemperature           ,
        tunnelBedrockMutation           TunnelBedrockMutation       ,
        tunnelCo                        TunnelCo
        }

AirTransmissivityInsideTheTunnel::=SEQUENCE{
        calculationAirTransmissivityInsideTheTunnel
```

```
                                        INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmAirTransmissivityInsideTheTunnel
                                        DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

WindSpeedInsideTheTunnel::=SEQUENCE{
        calculationWindSpeedInsideTheTunnel INTEGER             ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmWindSpeedInsideTheTunnel DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

TunnelFireDetection::=SEQUENCE{
        eventTunnelFireDetection        ENUMERATED {
                notDetected(0),detected(1),invalidData(9)}      ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmTunnelFireDetection      DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

Brightness::=SEQUENCE{
        calculationBrightness           INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
```

```
        confirmBrightness               DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

TunnelAnchorLoad::=SEQUENCE{
        calculationAnchorLoad           INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmTunnelAnchorLoad         DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

TunnelSlopeOfGround::=SEQUENCE{
        calculationSlopeOfGround        INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmTunnelSlopeOfGround      DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

TunnelStressDisplacement::=SEQUENCE{
        calculationStressDisplacement   INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmTunnelStressDisplacement DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

TunnelCrackedDisplacement::=SEQUENCE{
        calculationCrackedDisplacement  INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
```

```
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmTunnelCrackedDisplacement DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

TunnelSlopeOfPit::=SEQUENCE{
        calculationSlopeOfPit           INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmTunnelSlopeOfPit         DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

TunnelVerticalElastic::=SEQUENCE{
        calculationVerticalElastic      INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmTunnelVerticalElastic    DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

TunnelVerticalGround::=SEQUENCE{
        calculationVerticalGround       INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmTunnelVerticalGround     DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }
```

```
TunnelTemperature::=SEQUENCE{
        calculationTemperatureOfMeasurePoint INTEGER                    ,
        relationLinkageIdentifier            INTEGER                    OPTIONAL,
        pointLocation                        DsPointLocation            OPTIONAL,
        dateTime                             DsDateTime                 OPTIONAL,
        sensorID                             DsSensorID                 OPTIONAL,
        confirmTunnelTemperature             DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

TunnelBedrockMutation::=SEQUENCE{
        calculationBedrockMutation           INTEGER                    ,
        relationLinkageIdentifier            INTEGER                    OPTIONAL,
        pointLocation                        DsPointLocation            OPTIONAL,
        dateTime                             DsDateTime                 OPTIONAL,
        sensorID                             DsSensorID                 OPTIONAL,
        confirmTunnelBedrockMutation         DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

TunnelCo::=SEQUENCE{
        calculationCo                        INTEGER                    ,
        relationLinkageIdentifier            INTEGER                    OPTIONAL,
        pointLocation                        DsPointLocation            OPTIONAL,
        dateTime                             DsDateTime                 OPTIONAL,
        sensorID                             DsSensorID                 OPTIONAL,
        confirmTunnelCo                      DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

DsRouteEnvironment ::= CHOICE{
        co                                   Co                         ,
        nox                                  Nox                        ,
        so2                                  So2                        ,
        ch                                   Ch                         ,
        supendedParticulateMatter            SupendedParticulateMatter  ,
        photochemicalOxident                 PhotochemicalOxident       ,
```

```
        noise                       Noise                       ,
        vibration                   Vibration                   ,
        noiseLevel                  NoiseLevel                  ,
        no                          No                          ,
        no2                         No2
        }

Co::=SEQUENCE{
        calculationCo               INTEGER                     ,
        relationLinkageIdentifier   INTEGER                     OPTIONAL,
        pointLocation               DsPointLocation             OPTIONAL,
        dateTime                    DsDateTime                  OPTIONAL,
        sensorID                    DsSensorID                  OPTIONAL,
        confirmCo                   DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

Nox::=SEQUENCE{
        calculationNOx              INTEGER                     ,
        relationLinkageIdentifier   INTEGER                     OPTIONAL,
        pointLocation               DsPointLocation             OPTIONAL,
        dateTime                    DsDateTime                  OPTIONAL,
        sensorID                    DsSensorID                  OPTIONAL,
        confirmNox                  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

So2::=SEQUENCE{
        calculationSo2              INTEGER                     ,
        relationLinkageIdentifier   INTEGER                     OPTIONAL,
        pointLocation               DsPointLocation             OPTIONAL,
        dateTime                    DsDateTime                  OPTIONAL,
        sensorID                    DsSensorID                  OPTIONAL,
        confirmSo2                  DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }
```

```
Ch::=SEQUENCE{
        calculationCh                   INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmCh                       DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL

        }

SupendedParticulateMatter::=SEQUENCE{
        calculationSpm                  INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmSpm                      DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL

        }

PhotochemicalOxident::=SEQUENCE{
        calculationPo                   INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmPo                       DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL

        }

Noise::=SEQUENCE{
        calculationNoise                INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmNoise                    DeviceControlAnswerConfirmInfo
```

```
                                                        OPTIONAL
        }

Vibration::=SEQUENCE{
        calculationVibration            INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmVibration                DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

NoiseLevel ::=SEQUENCE{
        calculationNoiseLevelL5         INTEGER                 OPTIONAL,
        calculationNoiseLevelL10        INTEGER                 OPTIONAL,
        calculationNoiseLevelL50        INTEGER                 OPTIONAL,
        calculationNoiseLevelL90        INTEGER                 OPTIONAL,
        calculationNoiseLevelL95        INTEGER                 OPTIONAL,
        calculationEquivalentSoundLevel INTEGER                 OPTIONAL,
        calculationMaxSoundLevel        INTEGER                 OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmNoiseLevel               DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
        }

No ::= SEQUENCE{
        calculationNO                   INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL,
        pointLocation                   DsPointLocation         OPTIONAL,
        dateTime                        DsDateTime              OPTIONAL,
        sensorID                        DsSensorID              OPTIONAL,
        confirmNo                       DeviceControlAnswerConfirmInfo
                                                                OPTIONAL
```

```
        }

No2 ::= SEQUENCE{
        calculationNO2                  INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmNo2                      DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

DsNaturalDisaster ::= CHOICE{
        disasterDetection               DisasterDetection               ,
        mudSlideGroundMoisture          MudSlideGroundMoisture
        }

DisasterDetection ::=SEQUENCE{
        eventDisasterDetection          ENUMERATED{
                roadCollapse(1),bridgeDamage(2),highWaves(3),faceOfSlopeDamage(4),
                rockFall(5),landslide(6),snowslide(7),mudSlide(8),highSeas(9),eruption(10),
                roadsideFire(11),tunnelFire(12),invalidData(99)}        ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmDisasterDetection        DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

MudSlideGroundMoisture::=SEQUENCE{
        calculationMudSlideGroundMoisture   INTEGER                     ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmMudSlideGroundMoisture   DeviceControlAnswerConfirmInfo
```

```
                                                        OPTIONAL
        }

DsRockMeasurement::= CHOICE{
        rockElasticCrack                RockElasticCrack            ,
        rockInclination                 RockInclination             ,
        rockBedrockMutation             RockBedrockMutation         ,
        rockTemperature                 RockTemperature             ,
        undergroundTemperature          UndergroundTemperature      ,
        rockAcousticEmissionSensor      RockAcousticEmissionSensor  ,
        rockRainfall                    RockRainfall                ,
        rockCrackedMutation             RockCrackedMutation
        }

RockElasticCrack ::=SEQUENCE{
        calculationElasticCrack         INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
        pointLocation                   DsPointLocation             OPTIONAL,
        dateTime                        DsDateTime                  OPTIONAL,
        sensorID                        DsSensorID                  OPTIONAL,
        confirmRockElasticCrack         DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

RockInclination ::=SEQUENCE{
        calculationInclination          INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
        pointLocation                   DsPointLocation             OPTIONAL,
        dateTime                        DsDateTime                  OPTIONAL,
        sensorID                        DsSensorID                  OPTIONAL,
        confirmRockInclination          DeviceControlAnswerConfirmInfo
                                                                    OPTIONAL
        }

RockBedrockMutation ::=SEQUENCE{
        calculationBedrockMutation      INTEGER                     ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
```

```
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRockBedrockMutation      DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

RockTemperature ::=SEQUENCE{
        calculationTemperatureBySensor  INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRockTemperature          DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

UndergroundTemperature ::=SEQUENCE{
        calculationUndergroundTemperature   INTEGER                     ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmUndergroundTemperature   DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

RockAcousticEmissionSensor ::=SEQUENCE{
        calculationAESensor             INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRockAcousticEmissionSensor   DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }
```

```
RockRainfall ::=SEQUENCE{
        calculationRainfallAmount       INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRockRainfall             DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

RockCrackedMutation ::=SEQUENCE{
        calculationCrackedMutation      INTEGER                         ,
        relationLinkageIdentifier       INTEGER                         OPTIONAL,
        pointLocation                   DsPointLocation                 OPTIONAL,
        dateTime                        DsDateTime                      OPTIONAL,
        sensorID                        DsSensorID                      OPTIONAL,
        confirmRockCrackedMutation      DeviceControlAnswerConfirmInfo
                                                                        OPTIONAL
        }

DsRoadStructureMonitoringData ::= CHOICE{
        displacementObservationDevice       DisplacementObservationDevice,
        accelerationObservationDevice       AccelerationObservationDevice,
        warpObservationDevice               WarpObservationDevice       ,
        stressObservationDevice             StressObservationDevice     ,
        earthPressureObservationDevice      EarthPressureObservationDevice,
        spaceHydraulicPressureObservationDevice
                                            SpaceHydraulicPressureObservationDevice,
        waterLevelObservationDevice         WaterLevelObservationDevice
        }

DisplacementObservationDevice ::= SEQUENCE{
        collectionPosition              DsLocationInfo              ,
        facilityStructureType           ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
```

```
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationDisplacement             INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

AccelerationObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationAcceleration             INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

WarpObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationStrain                   INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

StressObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
```

```
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationStressValue              INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

EarthPressureObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationEarthPressure            INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

SpaceHydraulicPressureObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
        calculationPoreWaterPressure        INTEGER                 ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

WaterLevelObservationDevice ::= SEQUENCE{
        collectionPosition                  DsLocationInfo          ,
        facilityStructureType               ENUMERATED {
                roadSurface(1),shoulder(2),slopeFace(3),bridgeBeam(4),
                retainingWallAndBankProtection(5),crossingFacilities(6),tunnel(7),
                roadAccessory(8),utilityTunnel(9),occupiedProperty(10),other(11),
                invalidData(99)}                                    ,
        facilityStructureManagementNumber   OCTET STRING            ,
```

```
        calculationWaterLevel                   INTEGER                 ,
        relationLinkageIdentifier               INTEGER                 OPTIONAL
        }

DsVehicleWeightData::= DsVehicleWeight

DsVehicleWeight::= SEQUENCE{
        deviceLocation                          DsLocationInfo          OPTIONAL,
        vehicleNumber                           DsVehicleNumber         OPTIONAL,
        calculationCarTypeMeasurementResult     ENUMERATED{
                minicar(0),standardSizedCar(1),bus(2),lightLoads(3),smallTruck(4),
                goodsAndPassengersVehicle(5),ordinaryTruck(6),specialUsesVehicle(7),
                undeterminable(8),invalidData(99)}                      OPTIONAL,
        eventDataViolationRegulation            ENUMERATED{
                vehicleWidthOffence(1),vehicleLengthOffence(2),vehicleHeightOffence(3),
                vehicleWeightOffence(4),vehicleMaximumAxleWeightOffence(5),
                vehicleAdjacentAxleWeightOffence(6),vehicleCargoWidthOffence(7),
                vehicleCargoHeightOffence(8),vehicleCargoLengthOffence(9),
                vehicleCargoWeightOffence(10),trafficConditionOffence(11),
                others(99),invalidData(98)}                             OPTIONAL,
        calculationBodyVehicleGrossWeightMeasurementResult
                                                INTEGER                 ,
        calculationMaximumAxialLoadMeasurementResult
                                                INTEGER                 ,
        calculationAdjointingAxialLoadMeasurementResult
                                                INTEGER                 ,
        relationLinkageIdentifier               INTEGER                 OPTIONAL
        }

DsRoadEventData ::= CHOICE{
        dsEventInfo                             DsEventInfo             ,
        dsFacilitiesInfo                        DsFacilitiesInfo        ,
        dsFreightOperationSituationsInfo        DsFreightOperationSituationsInfo,
        dsTollCollectionInfo                    EXTERNAL                ,
        dsRoadAdministratorInfo                 DsRoadAdministratorInfo
        }
```

```
DsEventInfo ::= CHOICE{
        eventTrafficJamInfo             EventTrafficJamInfo         ,
        eventTrafficAccidentInfo        EventTrafficAccidentInfo    ,
        brokenDownInfo                  BrokenDownInfo              ,
        eventTargetObstaclesInfo        EventTargetObstaclesInfo    ,
        eventTrafficRestrictionInfo     EventTrafficRestrictionInfo ,
        eventConstructionInfo           EventConstructionInfo       ,
        weatherInfo                     WeatherInfo                 ,
        fireInfo                        FireInfo                    ,
        disasterInfo                    DisasterInfo                ,
        eventEarthquakeWarningInfo      EventEarthquakeWarningInfo,
        durationInfo                    DurationInfo                ,
        routeInfo                       RouteInfo                   ,
        stViolationInfo                 StViolationInfo             ,
        overloadingViolationInfo        OverloadingViolationInfo    ,
        roadSurfaceInfo                 RoadSurfaceInfo             ,
        suddenIncidentInfo              SuddenIncidentInfo          ,
        driverDisorderInfo              DriverDisorderInfo          ,
        roadAdministratorsFreeformatInfo RoadAdministratorsFreeformatInfo,
        otherOperatorsFreeformatInfo    OtherOperatorsFreeformatInfo,
        eventConstructionManagementInfo EventConstructionManagementInfo,
        disasterDamageInfo              DisasterDamageInfo          ,
        earthquakeSourceInfo            EarthquakeSourceInfo        ,
        earthquakeDamageInfo            EarthquakeDamageInfo        ,
        earthquakeDisasterInfo          EarthquakeDisasterInfo
        }

EventTrafficJamInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization              OPTIONAL,
        informerInfo                    DsOrganization              OPTIONAL,
        relationEventIdentifier         INTEGER                     OPTIONAL,
        eventStatusCode                 ENUMERATED {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventCauseOfTheTrafficJam       ENUMERATED {
                noClassification(0),intenseTraffic(1),accident(2),construction(3),weatherStatus(4),
                disaster(5),fire(6),brokenDownCar(7),roadObstacle(8),
                viewingLookingSideways(9),trafficRestrictions(10),closedToTraffic(11),
```

```
                operations(12),entertainmentEtc(13),earthquakeWarning(14),obstacles(15),
                others(97),unknown(98),invalidData(99)}                    ,
        eventCauseOfTheTrafficJamDetail     ENUMERATED {
                noDetails(0),nature(1),accident(2),construction(3),weatherStatus(4),disaster(5),
                fire(6),brokenDownCar(7),roadObstacle(8),viewingLookingSideways(9),
                intenseTraffic(10),trafficRestrictions(11),closedToTraffic(12),operations(13),
                entertainmentEtc(14),earthquakeWarning(15),obstacles(16),others(97),
                unknown(98),invalidData(99)}                               OPTIONAL,
        eventTrafficJamLesStatus            ENUMERATED {
                noJam(1),freeFlow(2),illFlow(3),jam(4),crowded(5),continuousJam(6),
                invalidData(9)}                                            ,
        roadLaneType                        ENUMERATED{
                berm(0),upRunningLane(1),oneRunningLane(2),twoRunningLane(3),
                threeRunningLane(4),fourRunningLane(5),fiveRunningLane(6),
                sixRunningLane(7),overtakingLane(8),allLanes(9),others(10),spare(11),
                sevenRunningLane(12),eightRunningLane(13),invalidData(99)},
        eventTrafficJamForecast             ENUMERATED {
                increase(1),decrease(2),invalidData(9)}                    OPTIONAL,
        eventTrafficJamTransitTime          INTEGER                        ,
        eventDateTime                       DsDateTime                     ,
        updateTime                          DsDateTime                     ,
        locationInfo                        DsLocationInfo                 ,
        eventDetourInfo                     DsRouteLocation                OPTIONAL,
        eventRelationInfo                   SET OF DsEventInfo             OPTIONAL,
        relationRelationType                ENUMERATED {
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                           OPTIONAL,
        eventCauseInfo                      SET OF DsEventInfo             OPTIONAL,
        relationLinkageIdentifier           INTEGER                        OPTIONAL
        }

EventTrafficAccidentInfo ::= SEQUENCE{
        informationCollectInfo              DsOrganization                 OPTIONAL,
        informerInfo                        DsOrganization                 OPTIONAL,
        relationEventIdentifier             INTEGER                        OPTIONAL,
        eventStatusCode                     ENUMERATED {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
```

```
eventAccidentObject             ENUMERATED {
        movingVehicle(1),protectiveWall(2),medianStrip(3),slopeSurface(4),tunnelWall(5),
        roadFacility(6),roadSurfaceObstacle(7),outsideRoad(8),pedestrian(9),
        nothingCorresponding(10),slope(11),guardRail(12),protectiveFence(13),
        invalidData(98),others(99)}                                 ,
eventAccidentPattern            ENUMERATED {
        detailUnknown(0),collision(1),rearEndCollision(2),minorCollision(3),
        runningInto(4),runningThrough(5),breakingThrough(6),runningOver(7),
        turningSideways(8),overturning(9),turningSidewaysOverturning(10),
        rearEndCollisionTurningSideways(11),falling(12),stumbling(13),
        collapseOfCargo(14),collidingWithFacility(15),vehicleFire(16),carAccident(17),
        accidentInvolvingInjuryDeath(18),accidentInvolvingDamageToProperty(19),
        reinspection(20),others(98),invalidData(99)}                ,
eventAccidentHandlingConditions ENUMERATED {
        detailUnknown(0),dealingWithTheSituation(1),standingBy(2),inspecting(3),
        reported(4),repairsCompleted(5),beingTowed(6),
        inspectionCompleteAndAccidentVehicleBeingRemoved(7),
        inspectionCompleteAndScatteredArticlesBeingRemoved(8),
        inspectionCompleteAndSpiltOilBeingTreated(9),
        inspectionCompleted(10),accidentVehicleBeingRemoved(11),
        injuredBeingRescued(12),wreckerInOperation(13),
        scatteredArticlesBeingRemoved(14),spiltOilBeingTreated(15),
        restrictionsToBeLiftedSoon(16),fireBeingExtinguished(17),fireSpreading(18),
        reinspecting(19),adjusting(20),others(98),invalidData(99)}  ,
objectVehicleInfo               DsTypeTrafficVolume         OPTIONAL,
personalDamageInfo              PersonalDamageInfo          OPTIONAL,
eventDateTime                   DsDateTime                  ,
updateTime                      DsDateTime                  ,
locationInfo                    DsLocationInfo              ,
eventDetourInfo                 DsRouteLocation             OPTIONAL,
eventRelationInfo               SET OF DsEventInfo          OPTIONAL,
relationRelationType            ENUMERATED {
        detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                            OPTIONAL,
relationLinkageIdentifier       INTEGER                     OPTIONAL
}
```

```
PersonalDamageInfo::=SEQUENCE{
        eventNumbersOfFatalInjuriesDue      INTEGER                 OPTIONAL,
        eventNumbersOfSeriousInjuriesDue    INTEGER                 OPTIONAL,
        eventNumbersOfMediumInjuriesDue     INTEGER                 OPTIONAL,
        eventNumbersOfSlightInjuriesDue     INTEGER                 OPTIONAL
        }

BrokenDownInfo ::= SEQUENCE{
        informationCollectInfo              DsOrganization          OPTIONAL,
        informerInfo                        DsOrganization          OPTIONAL,
        relationEventIdentifier             INTEGER                 OPTIONAL,
        eventStatusCode                     ENUMERATED {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventPatternOfBrokenDownCar         ENUMERATED {
                tirePuncture(61),engineTrouble(62),outOfGas(63),electricalSystemFailure(64),
                gearTrouble(65),breakTrouble(66),acceleratorTrouble(67),clutchTrouble(68),
                radiatorTrouble(69),unknown(70),others(71),invalidData(99)},
        locationInfo                        DsLocationInfo          ,
        eventRemovalStatusOfBrokenDownCar   ENUMERATED {
                undergoingRepairs(1),standingBy(2),repairsCompleted(3),beingTowed(4),
                beingRemoved(6),invalidData(9)}                     ,
        eventTypeOfBrokenDownVehicle        DsTypeTrafficVolume     OPTIONAL,
        eventDateTime                       DsDateTime              ,
        updateTime                          DsDateTime              ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

EventTargetObstaclesInfo ::= SEQUENCE{
        informationCollectInfo              DsOrganization          OPTIONAL,
        informerInfo                        DsOrganization          OPTIONAL,
        relationEventIdentifier             INTEGER                 OPTIONAL,
        eventStatusCode                     ENUMERATED {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventTargetObstacles                ENUMERATED {
                detailsUnknown(0),cargo(1),corrugatedCardboardBox(2),woodenBox(3)
                ,lumber(4),oil(5),sheeting(6),tire(7),autoparts(8),stoneSandAndGravel(9),
                animal(10),child(11),adult(12),magazinesAndTheLike(13),fluid(14),
```

```
                plywood(15),vinyl(16),corpseOfAnimal(17),tireFragment(18),fallingObject(81),
                clutter(82),oilLeak(83),others(97),unknown(98),invalidData(99)},
        eventObstaclesForm              ENUMERATED {
                detailsUnknown(0),dropped(1),scattered(2),drifted(3),windBlown(4),
                strayedIn(5),unknown(98),invalidData(99)}               ,
        eventObstaclesStatus            ENUMERATED {
                detailsUnknown(0),dealingWithTheSituation(1),
                removalWorksBeingConducted(2),checking(3),rescuing(4),working(5),
                wreckerInOperation(6),removing(7),removingScatteredArticles(8),
                disposingOfSpilt(9),inspecting(10),restrictionsToBeLiftedSoon(11),
                noInformation(96),others(97),unknown(98),invalidData(99)}  ,
        eventDateTime                   DsDateTime              ,
        updateTime                      DsDateTime              ,
        locationInfo                    DsLocationInfo          ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED {
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

EventTrafficRestrictionInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventRestrictionContent         ENUMERATED{
                noDetails(0),closedToTraffic(1),turningRestriction(2),speedLimit(3),
                trafficLaneRestriction(4),oneSidedControl(5),chainRestriction(6),
                onRampControl(7),closedToLargeSizedSpecialVehicles(8),
                intransitRestriction(9),offRampControl(10),roadNarrowAheads(12),
                trafficCaution(13),others(97),unknown(98),invalidData(99)},
        eventRestrictionContentDetails  ENUMERATED{
                noDetails(0),noEntry(101),roadClosedInWinter(102),
                roadShoulderRoadClosed(103),nightRoadClosed(104),timeClosed(105),
                bywalkClosed(106),rampClosed(107),noRightTurn(201),noLeftTurn(202),
```

```
        noStraightOn(203),noRightOrLeftTurn(204),speedLimit10(301),
        speedLimit20(302),speedLimit30h(303),speedLimit40(304),speedLimit50(305),
        speedLimit60(306),speedLimit70kmPerh(307),speedLimit80kmPerh(308),
        speedLimit90kmPerh(309),speedLimit100kmPerh(310),
        speedLimit110kmPerh(311),speedLimit120(312),speedLimit130(313),
        drivingSlowly(314),speedLimit(315),oneLaneRestricted(401),
        twoLanesRestricted(402),threeLanesRestricted(403),fourLanesRestricted(404),
        fiveLanesRestricted(405),sixLanesRestricted(406),sevenLanesRestricted(407),
        eightLanesRestricted(408),overtakingLaneRestricted(409),
        allLanesRestricted(410),crawlerLaneRestricted(411),
        roadShoulderRestricted(412),running1(413),running2(414),
        running1Running2(415),running2Overtaking(416),crawlerLaneRunning(417),
        intransitRestriction(418),bywalkRestricted(419),oneSidedAlternatingTraffic(501),
        oneSidedControl(502),twoWayTraffic(503),carryingChain(601),chainNeeded(602),
        chainEquipment(603),slipStopperCarry(604),slipStopperDemand(605),
        slipStopperLoad(606),entranceClosed(700),entranceLimited(701),
        closedToLargeSizedVehicles(801),closedToLargeSizedSpecialVehicles(802),
        closedToLargeSizedTrucks(803),bermRestricted(901),centerRestricted(902),
        offRampControl(1001),rampTrafficCaution(1301),bywalkTrafficCaution(1302),
        unknown(9800),invalidData(9900)}                    OPTIONAL,
eventRestrictionCause               ENUMERATED {
        noCauseOfEvent(0),accident(1),fire(2),brokenDownVehicle(3),roadObstacle(4),
        construction(5),operations(6),entertainmentEtc(7),weatherConditions(8),
        disaster(9),seismicAlert(10),others(14),unknown(15),rain(16),snow(17),mist(18),
        fog(19),freezing(20),sideWind(21),strongWind(22),storm(23),earthquake(24),
        fallenObject(25),trafficJam(26),constructionPlan(27),thunderstorm(28),
        windAndRain(29),strongWindAndRain(30),fallenSnow(31),snowstorm(32),
        snowstormBlownUpFromTheGroundByTheWind(33),highWaves(34),
        floodTide(35),bridgeCollapse(36),roadShoulderCollapse(37),surfaceCollapse(38),
        roadCollapse(39),seaWallCollapse(40),faceOfSlopeCollapse(41),
        snowRemovalOperation(42),tsunami(43),alert(44),warning(45),
        beforeRegulationOfTraffic(46),aLongConstruction(47),detonation(48),
        invalidData(99)}                                    ,
eventCauseDetails                   ENUMERATED{
        naturalCongestion(1),bottleneckCongestion(2),lookingSideways(3),
        accidentInvolvingVehicle(101),accidentInvolvingPeople(102),
        accidentInvolvingObject(103),accidentInvolvingOverturning(104),
```

accidentInvolvingHeadOnCollision(105),accidentByRearEndCollision(106),
accidentInvolvingCollision(107),accidentInvolvingOverRiding(108),
accidentInvolvingBreak(109),accidentInvolvingRollingOver(110),
accidentInvolvingFalling(111),accidentInvolvingCollisionWithFacilities(112),
vehicleFire(201),roadFire(202),medianStripFire(203),tunnelFire(204),
roadShoulderFire(205),roadsideFire(206),slopeFire(207),slopeSurfaceFire(208),
roadFacilitiesFire(209),vehicleWithFlatTyre(301),fallenObject(401),
droppedCargo(402),scatteredCargo(403),spiltCargo(404),oilLeak(405),
bstacle(406),person(407),animal(408),electricalWork(501),gasFitting(502),
waterWork(503),pavingWork(504),pavementWork(505),
telephoneConstruction(506),undergroundConstruction(507),sewerWork(508),
faceOfSlopeWork(509),bridgeConstruction(510),roadWork(511),
gardeningWork(512),roadSignWork(513),crashBarrierConstruction(514),
trafficSafetyFacilityConstruction(515),trafficControlFacilityConstruction(516),
soundproofWallConstruction(517),accidentRestorationWork(518),
disasterRestorationWork(519),installationWorkInsideTheTunnel(520),
constructionInTheCave(521),lightingInstallationWork(522),
laneMarkingWork(523),roadFacilitiesConstruction(524),intensiveWork(525),
constructionPlan(526),longTermConstruction(527),roadImprovementWork(528),
intersectionWork(529),bridgeWork(530),maintenanceWork(531),
rehabilitationWork(532),guardfenceWork(533),footbridgeWork(534),
commonDuctWork(535),communicationCableBoxWork(536),
informationBoxWork(537),undergroundCrossingWork(538),
undergroundWork(539),disasterMitigationWork(540),snowshedWork(541),
informationPanelWork(542),paintingWork(543),communicationWork(544),
railwayWork(545),expresswayWork(546),plantingConstruction(547),
disasterRecovery(548),tunnelWork(549),snowRemovalWork(550),
roadFacilitiesCleaning(601),treeCutting(602),plantingWork(603),
weedingWork(604),snowRemovalOperation(605),antifreezeSprayingWork(606),
faceOfSlopeOperation(607),drainingWork(608),bridgeRepairwork(609),
cleaningAndInspectionInsideTheTunnel(610),
lightingFacilityCleaningAndInspection(611),laneMarkingOperating(612),
slowCarOperation(613),inspectionWork(614),cleaning(615),check(616),
vipGuard(701),event(702),parade(703),festival(704),demo(705),
vehicleFreePromenade(706),marathon(707),exhibition(708),guard(709),
guardForStateGuest(710),snow(801),windAndSnow(802),heavySnow(803),
snowstorm(804),avalanche(805),snowfall(806),freezing(807),storm(808),

```
                sideWind(809),fog(810),thunder(811),slush(812),compressedSnow(813),
                thunderstorm(814),rain(815),windAndRain(816),heavyRain(817),typhoon(818),
                flood(819),riverFlooding(820),wave(821),floodTide(822),highWaves(823),
                highFlowWave(824),tsunami(825),eruption(826),mudSlide(827),collapse(828),
                earthquake(829),blizzard(830),denseFog(831),strongWind(832),
                strongWindAndRain(833),alert(834),warning(835),shiftingSand(836),
                landslide(901),rockfall(902),floodedRoad(903),overheadFlood(904),
                roadCollapse(905),roadDamage(906),fallenTree(907),gasExplosion(908),
                gasLeak(909),powerCut(910),lightning(911),bridgeCollapse(912),
                surfaceCollapse(913),roadShoulderCollapse(914),seaWallCollapse(915),
                faceOfSlopeCollapse(916),explosion(917),advanceTheRegulationOfPassage(918),
                unknown(1000),invalidData(9999)}                    OPTIONAL,
        roadLaneType                ENUMERATED{
                berm(0),upRunningLane(1),oneRunningLane(2),twoRunningLane(3),
                threeRunningLane(4),fourRunningLane(5),fiveRunningLane(6),
                sixRunningLane(7),overtakingLane(8),allLanes(9),others(10),spare(11),
                sevenRunningLane(12),eightRunningLane(13),invalidData(99)},
        eventTrafficRestrictionVehicle  ENUMERATED {
                closedToLargeSizedVehicles(0),closedToLargeSizedSpecialVehicles(1),
                closedToLargeSizedTrucks(2),closedToTraffic(3),invalidData(9)},
        eventDateTime               DsDateTime          ,
        updateTime                  DsDateTime          ,
        locationInfo                DsLocationInfo      ,
        eventDetourInfo             DsRouteLocation     OPTIONAL,
        eventRelationInfo           DsEventInfo         OPTIONAL,
        relationRelationType        ENUMERATED {
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                        OPTIONAL,
        eventCauseInfo              DsEventInfo         OPTIONAL,
        relationLinkageIdentifier   INTEGER             OPTIONAL,
        eventTrackingNumber         OCTET STRING        OPTIONAL,
        eventTrafficRestrictionTotalLanes   INTEGER     OPTIONAL,
        routeCourseDistance         INTEGER             OPTIONAL
        }

EventConstructionInfo ::= SEQUENCE{
        informationCollectInfo      DsOrganization      OPTIONAL,
```

```
informerInfo                    DsOrganization          OPTIONAL,
relationEventIdentifier         INTEGER                 OPTIONAL,
eventStatusCode                 ENUMERATED {
        detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
eventConstructionOperationConditions  ENUMERATED {
        continuous(1),intermittent(2),others(99),invalidData(98)}   ,
eventWeatherConditions          ENUMERATED {
        rainOrShine(1),rainedOff(2),postponeDueToRain(3),others(99),invalidData(98)},
roadLaneType                    ENUMERATED{
        berm(0),upRunningLane(1),oneRunningLane(2),twoRunningLane(3),
        threeRunningLane(4),fourRunningLane(5),fiveRunningLane(6),
        sixRunningLane(7),overtakingLane(8),allLanes(9),others(10),spare(11),
        sevenRunningLane(12),eightRunningLane(13),invalidData(99)},
eventDateTime                   DsDateTime              ,
updateTime                      DsDateTime              ,
restrictionDateTime             DsDateTime              ,
restrictionUpdateTime           DsDateTime              ,
locationInfo                    DsLocationInfo          ,
eventConstructionType           ENUMERATED {
        noDetails(0),repairWork(1),constructionStoppingAllTraffic(2),
        urgentConstruction(3),improvementConstruction(4),intensiveConstruction(5),
        snowWork(6),mowingWork(7),others(97),unknown(98),invalidData(99)},
eventOperationContent           ENUMERATED{
        noDetails(0),roadFacilitiesCleaning(1),plantingWork(2),weedingWork(3),
        snowRemovableWorks(4),antifreezeSprayingWork(5),faceOfSlopeWork(6),
        drainingWork(7),bridgeRepairWork(8),pavementConstruction(9),
        gardeningWork(10),crashBarrierConstruction(11),
        trafficSafetyFacilityConstruction(12),trafficControlFacilityConstruction(13),
        roadSignWork(14),soundproofWallConstruction(15),accidentRestorationWork(16),
        disasterRestorationWork(17),cleaningAndInspectionInsideTheTunnel(18),
        lightingFacilityCleaningAndInspection(19),
        installationWorkInsideTheTunnel(20),lightingInstallationWork(21),
        roadFacilitiesImprovementConstruction(22),laneMarkingWork(23),
        intensiveConstruction(24),urgentConstruction(25),electricalWork(26),
        waterWork(27),pavementWork(28),undergroundConstruction(29),
        bridgeConstruction(30),constructionInTheCave(31),gasFitting(32),
        telephoneConstruction(33),sewerWork(34),intersectionWork(35),
```

```
                footbridgeWork(36),informationPanelWork(37),disasterMitigationWork(38),
                guardfenceWork(39),snowshedWork(40),tunnelWork(41),undergroundWork(42),
                commonDuctWork(43),communicationCableBoxWork(44),
                informationBoxWork(45),roadImprovementWork(46),bridgeWork(47),
                expresswayWork(48),railwayWork(49),paintingWork(50),
                undergroundCrossingWork(51),communicationWork(52),
                plantingConstruction(53),maintenanceWork(54),rehabilitationWork(55),
                check(56),cleaning(57),others(98),invalidData(99)}          ,
        eventDetourInfo                 DsRouteLocation             OPTIONAL,
        eventRelationInfo               DsEventInfo                 OPTIONAL,
        relationRelationType            ENUMERATED {
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL,
        eventTrackingNumber             OCTET STRING                OPTIONAL,
        eventConstructionName           UTF8String                  OPTIONAL,
        eventConstructionPurpose        UTF8String                  OPTIONAL,
        eventConstructionRelationInfo   EventConstructionRelationInfo
                                                                    OPTIONAL,
        eventConstructionOwnerOrganization  DsOrganization          OPTIONAL
        }

EventConstructionRelationInfo::=SEQUENCE{
        eventConstructionOwnerURL       OCTET STRING                OPTIONAL,
        eventProjectEvaluationURL       OCTET STRING                OPTIONAL
        }

WeatherInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization              OPTIONAL,
        informerInfo                    DsOrganization              OPTIONAL,
        relationEventIdentifier         INTEGER                     OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),
                invalidData(9)}                                     ,
        eventWeatherPattern             ENUMERATED {
                notSpecified(0),fine(1),precipitation(2),snow(3),fog(4),thunder(5),
                windAndRain(6),thunderAndRain(7),heavyRain(8),windAndSnow(9),
```

```
        heavySnow(10),snowstorm(11),blowingSnow(12),freezing(13),fearOfFreezing(14),
        deepSnows(15),sherbet(16),snowCompaction(17),snowslide(18),sideWind(19),
        typhoon(20),earthquake(21),flooding(22),riverOverflow(23),leveeCollapse(24),
        waves(25),floodTide(26),highWaves(27),toppingWaves(28),tsunami(29),
        eruption(30),debrisFlow(31),thinCloud(32),overcast(33),dustStorm(34),
        sleet(35),graupel(36),hail(37),fineAndCloudy(38),others(98),invalidData(99)},
eventWeatherConditionForecast       ENUMERATED {
        invalidData(0),strong(1),weak(2),gettingStronger(3),gettingWeaker(4),
        subsiding(5),approaching(6),warningAnnounced(7),
        cautionAnnounced(8),cancelled(9)}                       ,
eventDateTime                       DsDateTime                  ,
updateTime                          DsDateTime                  ,
locationInfo                        DsLocationInfo              ,
eventRainfallStatus                 ENUMERATED {
        noRainfall(0),someRainfallIs(1),cautionMoreThan15mmPerhRainfall(2),
        cautionaryWarningOver30mmPerhRainfall(3),invalidData(9)}
                                                                OPTIONAL,
eventSnowfallStatus                 ENUMERATED  {
        noSnowfall(0),weakSnow(1),snow(2),strongSnow(3),drivenSnow(4),
        invalidData(9)}                                         OPTIONAL,
eventWindSpeedConditions            ENUMERATED {
        normal(1),caution(2),warning(3),invalidData(9)}         OPTIONAL,
eventWaveCondition                  ENUMERATED {
        normal(1),caution(2),warning(3),drop(4),invalidData(9)} OPTIONAL,
eventTransmittanceDecrease          ENUMERATED {
        declineInVisibilityIs(0),noDecline(1),invalidData(9)}   OPTIONAL,
eventVisibility                     ENUMERATED {
        thirtyMeter(1),fiftyMeter(2),seventyMeter(3),oneHundredMeter(4),
        oneHundredTwentyMeter(5),oneHundredFiftyMeter(6),
        oneHundredSeventyMeter(7),twoHundredMeter(8),invalidData(9)}
                                                                OPTIONAL,
eventRelationInfo                   DsEventInfo                 OPTIONAL,
relationRelationType                ENUMERATED{
        detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
eventPrecipitationType              ENUMERATED {
        rain(1),rainOrSnow(2),snowOrRain(3),snow(4),invalidData(9)}
```

```
                                                                    OPTIONAL,
    eventWeather                    ENUMERATED {
            clear(1),fine(2),thinCloud(3),overcast(4),haze(5),dustStorm(6),blizzard(7),fog(8),
            drizzle(9),rain(10),sleet(11),snow(12),graupel(13),hail(14),thunder(15),
            fineAndCloudy(16),heavyRain(17),invalidData(99)}        ,
    eventAttentionType              ENUMERATED{
            heavyRain(1),heavySnow(2),snowStorm(3),thunder(4),strongWind(5),
            waves(6),meltingSnow(7),flood(8),highWaves(9),densefog(10),dry(11),
            avalanche(12),lowTemperature(13),frost(14),iceAccretion(15),
            snowAccretion(16),other(17),invalidData(99)}            ,
    eventAttentionAndWarningContents UTF8String                     ,
    relationLinkageIdentifier       INTEGER                         OPTIONAL,
    eventMeasuredScaleOfAnEarthquake INTEGER                        OPTIONAL,
    calculationEarthquakeScale      INTEGER                         OPTIONAL
    }

FireInfo ::= SEQUENCE{
    informationCollectInfo          DsOrganization                  OPTIONAL,
    informerInfo                    DsOrganization                  OPTIONAL,
    relationEventIdentifier         INTEGER                         OPTIONAL,
    eventStatusCode                 ENUMERATED  {
            detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
    eventFireType                   DsTypeTrafficVolume             ,
    eventConditionOfFire            ENUMERATED{
            fireBeingExtinguished(1),spreading(2),invalidData(9)}   ,
    eventHandlingConditions         ENUMERATED{
            detailsUnknown(0),fireBeingExtinguished(1),fireSpreading(2),putOut(20),
            others(98),invalidData(99)}                             ,
    eventDateTime                   DsDateTime                      ,
    updateTime                      DsDateTime                      ,
    locationInfo                    DsLocationInfo                  ,
    eventRelationInfo               DsEventInfo                     OPTIONAL,
    relationRelationType            ENUMERATED {
            detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
    eventTunnelFireDetection        ENUMERATED {
            notDetected(0),detected(1),invalidData(9)}              ,
```

```
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DisasterInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED  {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDisasterType               ENUMERATED {
                noDetails(0),rockfall(1),landslide(2),fallenTree(3),roadDamage(4),
                submergedWater(5),roadCave(6),lackOfRoadShoulder(7),roadBroken(8),
                powerCut(9),lightning(10),gasExplosion(11),gasLeak(12),roadsideFire(13),
                roadShoulderFire(14),flood(15),mudSlide(16),pyrogenousOutflow(17),
                eruption(18),tsunami(19),roadCollapse(20),bridgeCollapse(21),
                seaWallCollapse(22),faceOfSlopeCollapse(23),disaster(24),others(25),
                unknown(26),invalidData(99)}                    ,
        eventDateTime                   DsDateTime              ,
        updateTime                      DsDateTime              ,
        locationInfo                    DsLocationInfo          ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
        eventProjectSectionNameOfDisasterOccurrence
                                        UTF8String              ,
        eventDisasterOccurrenceFacilityType     ENUMERATED  {
                bridge(0),crossRoadBridge(1),sharedGutter(2),tunnel(3),forseRoad(4),
                banking(5),retainingWall(6),rockShedSnowShed(7),invalidData(99)},
        eventDisasterOccurrenceFacilityQuantity
                                        INTEGER                 ,
        eventCauseDetails               ENUMERATED{
                naturalCongestion(1),bottleneckCongestion(2),
                lookingSideways(3),accidentInvolvingVehicle(101),accidentInvolvingPeople(102),
                accidentInvolvingObject(103),accidentInvolvingOverturning(104),
                accidentInvolvingHeadOnCollision(105),accidentByRearEndCollision(106),
                accidentInvolvingCollision(107),accidentInvolvingOverRiding(108),
```

accidentInvolvingBreak(109),accidentInvolvingRollingOver(110),
accidentInvolvingFalling(111),accidentInvolvingCollisionWithFacilities(112),
vehicleFire(201),roadFire(202),medianStripFire(203),tunnelFire(204),
roadShoulderFire(205),roadsideFire(206),slopeFire(207),slopeSurfaceFire(208),
roadFacilitiesFire(209),vehicleWithFlatTyre(301),fallenObject(401),
droppedCargo(402),scatteredCargo(403),spiltCargo(404),oilLeak(405),
obstacle(406),person(407),animal(408),electricalWork(501),gasFitting(502),
waterWork(503),pavingWork(504),pavementWork(505),
telephoneConstruction(506),undergroundConstruction(507),sewerWork(508),
faceOfSlopeWork(509),bridgeConstruction(510),roadWork(511),
gardeningWork(512),roadSignWork(513),crashBarrierConstruction(514),
trafficSafetyFacilityConstruction(515),trafficControlFacilityConstruction(516),
soundproofWallConstruction(517),accidentRestorationWork(518),
disasterRestorationWork(519),installationWorkInsideTheTunnel(520),
constructionInTheCave(521),lightingInstallationWork(522),
laneMarkingWork(523),roadFacilitiesConstruction(524),intensiveWork(525),
constructionPlan(526),longTermConstruction(527),roadImprovementWork(528),
intersectionWork(529),bridgeWork(530),maintenanceWork(531),
rehabilitationWork(532),guardfenceWork(533),footbridgeWork(534),
commonDuctWork(535),communicationCableBoxWork(536),
informationBoxWork(537),undergroundCrossingWork(538),
undergroundWork(539),disasterMitigationWork(540),snowshedWork(541),
informationPanelWork(542),paintingWork(543),communicationWork(544),
railwayWork(545),expresswayWork(546),plantingConstruction(547),
disasterRecovery(548),tunnelWork(549),snowRemovalWork(550),
roadFacilitiesCleaning(601),treeCutting(602),plantingWork(603),
weedingWork(604),snowRemovalOperation(605),antifreezeSprayingWork(606),
faceOfSlopeOperation(607),drainingWork(608),bridgeRepairwork(609),
cleaningAndInspectionInsideTheTunnel(610),
lightingFacilityCleaningAndInspection(611),laneMarkingOperating(612),
slowCarOperation(613),inspectionWork(614),cleaning(615),check(616),
vipGuard(701),event(702),parade(703),festival(704),demo(705),
vehicleFreePromenade(706),marathon(707),exhibition(708),guard(709),
guardForStateGuest(710),snow(801),windAndSnow(802),heavySnow(803),
snowstorm(804),avalanche(805),snowfall(806),freezing(807),storm(808),
sideWind(809),fog(810),thunder(811),slush(812),compressedSnow(813),
thunderstorm(814),rain(815),windAndRain(816),heavyRain(817),typhoon(818),

```
                flood(819),riverFlooding(820),wave(821),floodTide(822),highWaves(823),
                highFlowWave(824),tsunami(825),eruption(826),mudSlide(827),collapse(828),
                earthquake(829),blizzard(830),denseFog(831),strongWind(832),
                strongWindAndRain(833),alert(834),warning(835),shiftingSand(836),
                landslide(901),rockfall(902),floodedRoad(903),overheadFlood(904),
                roadCollapse(905),roadDamage(906),fallenTree(907),gasExplosion(908),
                gasLeak(909),powerCut(910),lightning(911),bridgeCollapse(912),
                surfaceCollapse(913),roadShoulderCollapse(914),seaWallCollapse(915),
                faceOfSlopeCollapse(916),explosion(917),advanceTheRegulationOfPassage(918),
                unknown(1000),invalidData(9999)}                    OPTIONAL,
        eventDisasterDetection          ENUMERATED{
                roadCollapse(1),bridgeDamage(2),highWaves(3),faceOfSlopeDamage(4),
                rockFall(5),landslide(6),snowslide(7),mudSlide(8),highSeas(9),eruption(10),
                roadsideFire(11),tunnelFire(12),invalidData(99)}    ,
        eventDisasterOutline            UTF8String                  ,
        eventDisasterDetail             UTF8String                  OPTIONAL,
        eventRestorationOutline         UTF8String                  ,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

EventEarthquakeWarningInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization              OPTIONAL,
        informerInfo                    DsOrganization              OPTIONAL,
        relationEventIdentifier         INTEGER                     OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventEarthquakeWarningAnnouncementPlace
                                        ENUMERATED{
                detailsUnknown(0),offshoreTokai(1),southernKanto(2),others(98),
                invalidData(99)}                                    ,
        dateTime                        DsDateTime                  ,
        eventRelationInfo               DsEventInfo                 OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }
```

```
DurationInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        startLocationInfo               DsLocationInfo          ,
        endLocationInfo                 DsLocationInfo          ,
        routeSectionUnitDuration        INTEGER                 ,
        routeSectionAverageSpeed        INTEGER                 ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
        routeSectionStaticCourseDuration INTEGER                ,
        routeStaticTravelSpeed          INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

RouteInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED  {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        routeAttribute                  ENUMERATED  {
                timeBeelineRoute(1),goalRoute(2),noTrespassingRoute(3),
                particularizeCarServicePassRoute(4),busServiceRoute(5),
                roadManagementCarServiceRoute(6),refugeRoute(7),emergencyCarRoute(8),
                pedestrianRoute(9),emergencyTransportationRoute(10),invalidData(99)},
        startLocationInfo               DsLocationInfo          ,
        byLocationInfo                  DsLocationInfo          ,
        endLocationInfo                 DsLocationInfo          ,
        routeCourseDistance             INTEGER                 ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
```

```
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

StViolationInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization              OPTIONAL,
        informerInfo                    DsOrganization              OPTIONAL,
        relationEventIdentifier         INTEGER                     OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDataViolationRegulation    ENUMERATED{
                vehicleWidthOffence(1),vehicleLengthOffence(2),vehicleHeightOffence(3),
                vehicleWeightOffence(4),vehicleMaximumAxleWeightOffence(5),
                vehicleAdjacentAxleWeightOffence(6),vehicleCargoWidthOffence(7),
                vehicleCargoHeightOffence(8),vehicleCargoLengthOffence(9),
                vehicleCargoWeightOffence(10),trafficConditionOffence(11),
                others(99),invalidData(98)}                         ,
        dateTime                        DsDateTime                  ,
        locationInfo                    DsLocationInfo              ,
        eventRelationInfo               DsEventInfo                 OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier       INTEGER                     OPTIONAL
        }

OverloadingViolationInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization              OPTIONAL,
        informerInfo                    DsOrganization              OPTIONAL,
        relationEventIdentifier         INTEGER                     OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDataViolationRegulation    ENUMERATED  {
                vehicleWidthOffence(1),vehicleLengthOffence(2),vehicleHeightOffence(3),
                vehicleWeightOffence(4),vehicleMaximumAxleWeightOffence(5),
                vehicleAdjacentAxleWeightOffence(6),vehicleCargoWidthOffence(7),
```

```
                vehicleCargoHeightOffence(8),vehicleCargoLengthOffence(9),
                vehicleCargoWeightOffence(10),trafficConditionOffence(11),
                others(99),invalidData(98)}                               ,
        dateTime                        DsDateTime                        ,
        locationInfo                    DsPointLocation                   ,
        vehicleNumber                   DsVehicleNumber                   ,
        calculationBodyVehicleGrossWeightMeasurementResult
                                        INTEGER                           ,
        calculationMaximumAxialLoadMeasurementResult
                                        INTEGER                           ,
        calculationAdjointingAxialLoadMeasurementResult
                                        INTEGER                           ,
        eventRelationInfo               DsEventInfo                       OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                          OPTIONAL,
        relationLinkageIdentifier       INTEGER                           OPTIONAL
        }

RoadSurfaceInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization                    OPTIONAL,
        informerInfo                    DsOrganization                    OPTIONAL,
        relationEventIdentifier         INTEGER                           OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        dateTime                        DsDateTime                        ,
        pointLocation                   DsLocationInfo                    ,
        eventRoadSurfaceConditions      ENUMERATED{
                frozen(1),accumulationOfSnow(2),damp(3),filmOfWater(4),dry(5),
                invalidData(9)}                                           ,
        eventAhsRoadSurfaceConditions13Detail
                                        ENUMERATED{
                verySlipperyThickSnow(1),verySlipperyIceSheet(2),verySlipperyIceFilm(3),
                iceSheet(4),iceFilm(5),iceSheetUnderPowderSnow(6),
                iceSheetUnderGranularSnow(7),thickSnow(8),powderSnow(9),granularSnow(10),
                sherbet(11),wet(12),dry(13),invalidData(99)}              ,
        eventAhsRoadSurfaceConditions7  ENUMERATED{
```

```
                verySlipperyRoad(1),iceRink(2),thickSnow(3),powderAndGranularSnow(4),
                sherbet(5),wet(6),dry(7),invalidData(9)}                    ,
        eventAhsRoadSurfaceConditions5      ENUMERATED{
                frozen(1),snow(2),damp(3),waterFilm(4),dry(5),invalidData(9)},
        eventRelationInfo                   DsEventInfo             OPTIONAL,
        relationRelationType                ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

SuddenIncidentInfo ::= SEQUENCE{
        informationCollectInfo              DsOrganization          OPTIONAL,
        informerInfo                        DsOrganization          OPTIONAL,
        relationEventIdentifier             INTEGER                 OPTIONAL,
        eventStatusCode                     ENUMERATED  {
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        dateTime                            DsDateTime              ,
        pointLocation                       DsLocationInfo          ,
        eventSuddenIncidentDetect           ENUMERATED{
                accident(0),fire(1),roadObstacle(2),disaster(3),seism(4),highSeas(5),mudSlide(6),
                submergedWater(7),landslide(8),rockFall(9),roadCollapse(10),roadBroken(11),
                bridgeCollapse(12),surfaceCollapse(13),roadShoulderCollapse(14),
                seaWallCollapse(15),faceOfSlopeCollapse(16),others(17),invalidData(99)},
        eventRelationInfo                   DsEventInfo             OPTIONAL,
        relationRelationType                ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                    OPTIONAL,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

DriverDisorderInfo::= SEQUENCE{
        informationCollectInfo              DsOrganization          OPTIONAL,
        informerInfo                        DsOrganization          OPTIONAL,
        relationEventIdentifier             INTEGER                 OPTIONAL,
        eventStatusCode                     ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
```

```
        eventAhsConfirmationUnusualBreakOutToDriver
                                        ENUMERATED {
                driverAbnormal(1),driverNotAbnormal(2),invalidData(9)}    ,
        driverDisorderDateTime          DsDateTime                        ,
        driverDisorderLocation          DsLocationInfo                    ,
        relationLinkageIdentifier       INTEGER                           OPTIONAL
        }

RoadAdministratorsFreeformatInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization                    OPTIONAL,
        informerInfo                    DsOrganization                    OPTIONAL,
        relationEventIdentifier         INTEGER                           OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventOtherInformation           UTF8String                        ,
        relationLinkageIdentifier       INTEGER                           OPTIONAL
        }

OtherOperatorsFreeformatInfo ::= SEQUENCE{
        informationCollectInfo          DsOrganization                    OPTIONAL,
        informerInfo                    DsOrganization                    OPTIONAL,
        relationEventIdentifier         INTEGER                           OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventAnotherSubjectInfo         UTF8String                        ,
        relationLinkageIdentifier       INTEGER                           OPTIONAL
        }

EventConstructionManagementInfo ::=SEQUENCE{
        informationCollectInfo          DsOrganization                    OPTIONAL,
        informerInfo                    DsOrganization                    OPTIONAL,
        relationEventIdentifier         INTEGER                           OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDateTime                   DsDateTime                        ,
        updateTime                      DsDateTime                        ,
        locationInfo                    DsLocationInfo                    ,
```

```
eventConstructionContent        ENUMERATED {
        noDetails(0),roadFacilitiesCleaning(1),plantingWork(2),weedingWork(3),
        snowRemovableWorks(4),antifreezeSprayingWork(5),faceOfSlopeWork(6),
        drainingWork(7),bridgeRepairWork(8),pavementConstruction(9),
        gardeningWork(10),crashBarrierConstruction(11),
        trafficSafetyFacilityConstruction(12),trafficControlFacilityConstruction(13),
        roadSignWork(14),soundproofWallConstruction(15),
        accidentRestorationWork(16),disasterRestorationWork(17),
        cleaningAndInspectionInsideTheTunnel(18),
        lightingFacilityCleaningAndInspection(19),
        installationWorkInsideTheTunnel(20),lightingInstallationWork(21),
        roadFacilitiesImprovementConstruction(22),laneMarkingWork(23),
        intensiveConstruction(24),urgentConstruction(25),electricalWork(26),
        waterWork(27),pavementWork(28),undergroundConstruction(29),
        bridgeConstruction(30),constructionInTheCave(31),gasFitting(32),
        telephoneConstruction(33),sewerWork(34),others(98),invalidData(99)},
eventRelationInfo               DsEventInfo             OPTIONAL,
relationRelationType            ENUMERATED{
        detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                        OPTIONAL,
eventProgressLengthOfConstructionLink
                                INTEGER                 ,
eventCurrentRateOfConstruction  INTEGER                 ,
eventNamesOfOperationalConstructionEquipmentForRestoration
                                ENUMERATED{
        policeVehicle(1),crane(2),roadSignCleaningVehicle(3),debrisCarryingShip(4),
        seaSurfaceCleaningShip(5),dustCarryingSystem(6),weedingVehiclelarge(7),
        weedingMachineSmall(8),waterSprinklingVehicle(9),
        roadCleaningVehicle(10),crashBarrierCleaningVehicle(11),
        drainCleaningVehicle(12),sideDrainCleaningVehicle(13),
        tunnelCleaningVehicle(14),liftingVehicle(15),bridgeMaintenanceVehicle(16),
        skidResistanceTestVehicle(17),roadMaintenanceVehicleFlexMeasurement(18),
        examinationVehicleAmphibian(19),motorizedGrader(20),forkLiftTruck(21),
        lightingVehicle(22),sandBagMakingVehicle(23),
        damageRecoveryVehicleMultiPurposeType(24),
        damageRecoveryVehicleHeadquarter(25),satelliteCommunicationVehicle(26),
        pumpVehicle(27),portableBridge(28),areaInChargeCheckingMachine(29),
```

```
        runningVehicleWeightMeter(30),invalidData(99)},
    eventNumbersOfOperationalConstructionEquipmentForRestoration
                                    INTEGER             ,
    eventRoadStructureToBeConstructed   ENUMERATED{
        roadSurface(1),roadShoulder(2),faceOfSlope(3),bridge(4),
        retainingWallSeaWall(5),crossingFacilities(6),tunnel(7),roadFacilities(8),
        commonDuct(9),occupiedFacilities(10),others(11),invalidData(99)},
    relationLinkageIdentifier       INTEGER             OPTIONAL
    }

DisasterDamageInfo ::=SEQUENCE{
    informationCollectInfo          DsOrganization      OPTIONAL,
    informerInfo                    DsOrganization      OPTIONAL,
    relationEventIdentifier         INTEGER             OPTIONAL,
    eventStatusCode                 ENUMERATED{
        detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
    eventDateTime                   DsDateTime          ,
    updateTime                      DsDateTime          ,
    locationInfo                    DsLocationInfo      ,
    eventRelationInfo               DsEventInfo         OPTIONAL,
    relationRelationType            ENUMERATED{
        detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                        OPTIONAL
    dsRoadSurfaceDamage             DsRoadSurfaceDamage ,
    dsShoulderDamage                DsShoulderDamage    ,
    dsSlopeFaceDamage               DsSlopeFaceDamage   ,
    dsBridgeDamage                  DsBridgeDamage      ,
    dsRetainingWallRevetmentDamage  DsRetainingWallRevetmentDamage,
    dsCrossingFacilityDamage        DsCrossingFacilityDamage ,
    dsTunnelDamage                  DsTunnelDamage      ,
    dsRoadAccessoriesDamage         DsRoadAccessoriesDamage ,
    dsCommonDuctDamage              DsCommonDuctDamage  ,
    dsRoadBoardDamageConditions     DsRoadBoardDamageConditions,
    dsBankDamage                    DsBankDamage        ,
    eventRoadDamageLength           INTEGER             ,
    eventRoadDamageArea             INTEGER             ,
    eventRoadDamageSoilVolume       INTEGER             ,
```

```
        eventRestorationStatus              ENUMERATED{
                nomeasures(0),closedToVehicleTraffic(1),trafficRegulation(2),invalidData(9)},
        eventEmergencyRestorationConditions ENUMERATED{
                notInitiated(0),underConstruction(1),constructionCompleted(2),invalidData(9)},
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

DsRoadSurfaceDamage ::= SET OF EventRoadSurfaceDamage
EventRoadSurfaceDamage ::= ENUMERATED{
        deformationOfCrack(1),sinking(2),localSwelling(3),breakOfJoint(4),bump(5),
        soil(6),rockFall(7),inundation(8),invalidData(9)}

DsShoulderDamage ::= SET OF EventShoulderDamage
EventShoulderDamage ::= ENUMERATED{
        deformationOfCrack(1),sinking(2),bump(3),invalidData(9)}

DsSlopeFaceDamage ::= SET OF EventSlopeFaceDamage
EventSlopeFaceDamage ::= ENUMERATED{
        crack(1),collapse(2),rockfall(3),crumblingRock(4),protectionCollapse(5),
        drainCollapse(6),collapseOfAntiRockfallSystem(7),
        collapseOfOtherFaceOfSlopeFacilities(8),runoff(9),invalidData(99)}
EventBridgeDamage ::= ENUMERATED{
        crackInFloorConcrete(1),swellingOfPavementOfBridge(2),
        crackInPavementOfSurface(3),breakageOfHandrail(4),
        breakageOfWeldingPartOfExpansionJoint(5),deformationOfExpansionJoint(6),
        deformationOrTwistingOfMainStructure(7),crackInMainStructure(8),
        crackInConcreteSpar(9),collapseOfSubstructure(10),
        submersionOrSlantOfSubstructure(11),otherBreakageOfFacilities(12),
        fallABridge(13),openOfBridge(14),bumpOfBridge(15),invalidData(99)}

DsRetainingWallRevetmentDamage ::= SET OF EventRetainingWallRevetmentDamage
EventRetainingWallRevetmentDamage ::= ENUMERATED{
        crackInRetainingWall(1),crackInJointOfRetainingWall(2),
        localSwellingOfRetainingWall(3),deformationOfRetainingWall(4),
        crackOfBackFillOfSeaWall(5),submersion(6),invalidData(9)}

DsCrossingFacilityDamage ::= SET OF EventCrossingFacilityDamage
```

```
EventCrossingFacilityDamage ::= ENUMERATED{
        troubleWithPoleOrSparOfPedestrianBridge(1),
        breakageOfJointOfPedestrianBridge(2),
        submersionOfFoundationOfPedestrianBridge(3),
        troubleWithPedestrianTunnel(4),
        troubleWithDrainPumpOfPedestrianTunnel(5),invalidData(9)}

DsTunnelDamage ::= SET OF EventTunnelDamage
EventTunnelDamage ::= ENUMERATED{
        breakageOfLiningLighting(1),crackInLining(2),leakageOfWater(3),
        collapseOfGateOrGateSlope(4),breakageOfVentilation(5),
        breakageOfLightning(6),breakageOfEmergencyFacilities(7),collapseOfAMask(8),
        invalidData(9)}

DsRoadAccessoriesDamage ::= SET OF EventRoadAccessoriesDamage
EventRoadAccessoriesDamage ::= ENUMERATED{
        bendingOfLightingPoleOrRoadSigns(1),breakageOfLightingConnection(2),
        troubleWithBicycleParking(3),invalidData(9)}

DsCommonDuctDamage ::= SET OF EventCommonDuctDamage
EventCommonDuctDamage ::= ENUMERATED{
        crackInBody(1),leakageOfWaterFromJoint(2),troubleWithManhole(3),
        otherBreakageOfDrainageVentilationLightingElectricalSupply(4),invalidData(9)}

DsRoadBoardDamageConditions ::= SET OF EventRoadBoardDamageConditions
EventRoadBoardDamageConditions ::= ENUMERATED{
        noEntry(0),collapse(1),washout(2),upheaval(3),invalidData(9)}

DsBankDamage ::= SET OF EventEmbankmentDamageConditions
EventEmbankmentDamageConditions ::= ENUMERATED{
        noEntry(0),collapse(1),washout(2),invalidData(9)}

EarthquakeSourceInfo ::=SEQUENCE{
        informationCollectInfo      DsOrganization      OPTIONAL,
        informerInfo                DsOrganization      OPTIONAL,
        relationEventIdentifier     INTEGER             OPTIONAL,
        eventStatusCode             ENUMERATED{
```

```
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDateTime                   DsDateTime              ,
        updateTime                      DsDateTime              ,
        locationInfo                    DsLocationInfo          ,
        calculationCentrumDepth         INTEGER                 ,
        calculationEarthquakeScale      INTEGER                 ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

EarthquakeDamageInfo ::=SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDateTime                   DsDateTime              ,
        updateTime                      DsDateTime              ,
        locationInfo                    DsLocationInfo          ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                                                OPTIONAL,
        dsOrganization                  DsOrganization          ,
        dsRoadObstruction               DsRoadObstruction       ,
        dsRoadSideFire                  DsRoadSideFire          ,
        dsRoadSurfaceDamage             DsRoadSurfaceDamage     ,
        dsShoulderDamage                DsShoulderDamage        ,
        dsSlopeFaceDamage               DsSlopeFaceDamage       ,
        dsBridgeDamage                  DsBridgeDamage          ,
        dsRetainingWallRevetmentDamage  DsRetainingWallRevetmentDamage,
        dsCrossingFacilityDamage        DsCrossingFacilityDamage,
        dsTunnelDamage                  DsTunnelDamage          ,
        dsRoadAccessoriesDamage         DsRoadAccessoriesDamage ,
```

```
        dsCommonDuctDamage              DsCommonDuctDamage      ,
        dsRoadBoardDamageConditions     DsRoadBoardDamageConditions,
        dsBankDamage                    DsBankDamage            ,
        eventRoadDamageLength           INTEGER                 ,
        eventRoadDamageArea             INTEGER                 ,
        eventRoadDamageSoilVolume       INTEGER                 ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }

DsRoadObstruction ::= SET OF EventOutlineInformationRoadPassingObstacle
EventOutlineInformationRoadPassingObstacle ::= ENUMERATED{
        notAvailable(0),available(1),invalidData(9)}

DsRoadSideFire ::= SET OF EventOutlineInformationRoadsideFire
EventOutlineInformationRoadsideFire ::= ENUMERATED{
        notAvailable(0),available(1),invalidData(9)}

EarthquakeDisasterInfo ::=SEQUENCE{
        informationCollectInfo          DsOrganization          OPTIONAL,
        informerInfo                    DsOrganization          OPTIONAL,
        relationEventIdentifier         INTEGER                 OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailsUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        eventDateTime                   DsDateTime              ,
        updateTime                      DsDateTime              ,
        locationInfo                    DsLocationInfo          ,
        eventRelationInfo               DsEventInfo             OPTIONAL,
        relationRelationType            ENUMERATED{
                detailUnknown(0),cause(1),effect(2),concurrent(3),invalidData(9)}
                                        OPTIONAL                ,
        eventRestorationStatus          ENUMERATED{
                nomeasures(0),closedToVehicleTraffic(1),trafficRegulation(2),invalidData(9)},
        eventEmergencyRestorationConditions  ENUMERATED  {
                notInitiated(0),underConstruction(1),constructionCompleted(2),invalidData(9)},
        eventRestorationOutline         UTF8String              ,
        relationLinkageIdentifier       INTEGER                 OPTIONAL
        }
```

```
DsFacilitiesInfo ::= CHOICE{
        autoPark                        AutoPark        ,
        saPa                            SaPa            ,
        rs                              Rs              ,
        publicFacility                  PublicFacility  ,
        otherFacility                   OtherFacility
        }

AutoPark ::= SEQUENCE{
        informationCollectInfo          DsOrganization  OPTIONAL,
        informerInfo                    DsOrganization  OPTIONAL,
        dsDateTime                      DsDateTime      ,
        facilityAutoParkName            UTF8String      ,
        facilityAutoParkTelephoneNum    UTF8String      ,
        facilityFacilityCode            OCTET STRING    ,
        entranceLocationInfo            DsLocationInfo  ,
        locationInfo                    DsLocationInfo  ,
        facilityAutoParkAddress         UTF8String      ,
        facilityAutoParkType            UTF8String      ,
        facilityAutoParkStructure       UTF8String      ,
        facilityAmountVehiclesToBeAccommodated
                                        ENUMERATED {
                lessThan20(0),under50(1),under100(2),under200(3),under500(4),under1000(5),
                over1000(6),unknown(7),invalidData(9)}  ,
        facilityAutoParkCapacity        INTEGER         ,
        facilityHeightRestriction       ENUMERATED {
                nil(0),restrictions(1),unused(2),unknown(3),invalidData(9)}  ,
        facilityVehicleTypeRestriction  ENUMERATED {
                nil(0),largeSizedVehicle(1),passengerCarWith3Number(2),others(3),
                unknown(4),invalidData(9)}              ,
        facilityAutoParkLengthRestriction UTF8String    ,
        eventDateTime                   DsDateTime      ,
        updateTime                      DsDateTime      ,
        businessDay                     DsDateTime      ,
        facilityParkingFee              INTEGER         ,
        facilityChargeUnit              ENUMERATED{
```

```
                thirtyMinutes(0),oneHour(1),twoHours(2),threeHours(3),halfDay(4),oneDay(5),
                oneTime(6),unknown(7),invalidData(9)}                    ,
        facilityDiscountedFee           ENUMERATED{
                nil(0),discountAvailable(1),unused(2),noInformation(3),invalidData(9)},
        facilityAutoParkFee             INTEGER                          ,
        facilityCongestionStatus        ENUMERATED {
                empty(0),fullCapacity(1),crowded(2),invalidData(9)}      ,
        facilityParkingTurnoverRate     INTEGER                          ,
        facilityWaitingTime             INTEGER                          ,
        facilityNumberOfParkingVehicles INTEGER                          ,
        facilityCongestionDegreeForecast ENUMERATED{
                thisConditionWillContinue(1),willBecomeCrowded(2),willBecomeEmpty(3),
                uncertain(4),invalidData(98),other(99)}                  ,
        relationLinkageIdentifier       INTEGER                  OPTIONAL
        }

SaPa ::= SEQUENCE{
        informationCollectInfo          DsOrganization           OPTIONAL,
        informerInfo                    DsOrganization           OPTIONAL,
        dsDataTime                      DsDateTime               ,
        facilitySaPaName                UTF8String               ,
        facilityFacilityCode            OCTET STRING             ,
        locationInfo                    DsLocationInfo           ,
        facilityFacilityContent         UTF8String               ,
        startDateTime                   DsDateTime               ,
        endDateTime                     DsDateTime               ,
        relationLinkageIdentifier       INTEGER                  OPTIONAL
        }

Rs ::= SEQUENCE{
        informationCollectInfo          DsOrganization           OPTIONAL,
        informerInfo                    DsOrganization           OPTIONAL,
        dsDataTime                      DsDateTime               ,
        facilityRsName                  UTF8String               ,
        facilityFacilityCode            OCTET STRING             ,
        locationInfo                    DsLocationInfo           ,
        facilityFacilityContent         UTF8String               ,
```

```
        startDateTime                   DsDateTime          ,
        endDateTime                     DsDateTime          ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

PublicFacility::= SEQUENCE{
        informationCollectInfo          DsOrganization      OPTIONAL,
        informerInfo                    DsOrganization      OPTIONAL,
        dsDateTime                      DsDateTime          ,
        facilityNameOfFacility          UTF8String          ,
        facilityTypeOfFacility          UTF8String          ,
        facilityAddressOfFacility       UTF8String          ,
        facilityTelephoneNumOfFacility  UTF8String          ,
        facilityOpenDay                 UTF8String          ,
        openHoursOpen                   DsDateTime          ,
        openHoursClose                  DsDateTime          ,
        facilityChargeOfFacility        UTF8String          ,
        facilityInformationOfEvent      UTF8String          OPTIONAL,
        facilityCapacityOfFacility      INTEGER             ,
        facilityNumberOfBooking         INTEGER             ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

OtherFacility ::= SEQUENCE{
        informationCollectInfo          DsOrganization      OPTIONAL,
        informerInfo                    DsOrganization      OPTIONAL,
        dsDateTime                      DsDateTime          ,
        facilityOtherFacilityName       UTF8String          ,
        locationInfo                    DsLocationInfo      ,
        facilityFacilityCode            OCTET STRING        ,
        facilityFacilityContent         UTF8String          ,
        startDateHour                   DsDateTime          ,
        endDateHour                     DsDateTime          ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsFreightOperationSituationsInfo::=CHOICE{
```

```
        freightVehicleInfo                  FreightVehicleInfo          ,
        freightOperationSituationInfo       FreightOperationSituationInfo
        }

FreightVehicleInfo ::= SEQUENCE{
        informationCollectInfo              DsOrganization              OPTIONAL,
        informerInfo                        DsOrganization              OPTIONAL,
        movableOnboardDeviceVehicleCode     ENUMERATED{
                miniTwoWheeledVehicle(1),smallSizeTwoWheeledVehicle(2),
                miniThreeWheeledVehicle(3),miniFourWheeledPassengerCar(4),
                miniFourWheeledTruck(5),smallSizedPassengerCar(6),smallSizedTruck(7),
                standardSizedPassengerCar(8),trailerNormal(9),standardSizedTruckNormal(10),
                microBus(11),trailerMediumSized(12),standardSizedTruckLargeSize(13),bus(14),
                trailerLargeSized(15),standardSizedTruckExtraLarge(16),trailerExtraLarge(17),
                extraLargeSizedSpecialPurposeVehicle(18),busExtraLarge(19),invalidData(99)},
        movableVehicleLicencePlateNumber    OCTET STRING                ,
        movableVehicleLengthOverall         INTEGER                     ,
        movableVehicleHeightOverall         INTEGER                     ,
        movableVehicleWidth                 INTEGER                     ,
        movableBodyCapacityCargoLoad        INTEGER                     ,
        movableVehicleWeight                INTEGER                     ,
        movableBodyVehicleGrossWeight       INTEGER                     ,
        calculationVehicleSpecificCharacteristics
                                            UTF8String                  ,
        statisticsBodyCapacityPassengers    INTEGER                     ,
        movableTotalPistonDisplacement      INTEGER                     ,
        movableBodyFuelType                 ENUMERATED  {
                gasoline(1),lightOil(2),lpg(3),electric(4),others(5),unknown(8),invalidData(9)},
        movableAxleLoadBeforehand           INTEGER                     ,
        movableLongitudinalAxisMultiple     INTEGER                     ,
        movablePostfrontAxleMultiple        INTEGER                     ,
        movableAxleLoadFuture               INTEGER                     ,
        movableVehicleIdentificationNumber  INTEGER                     ,
        calculationCarTypeMeasurementResult ENUMERATED{
                minicar(0),standardSizedCar(1),bus(2),lightLoads(3),smallTruck(4),
                goodsAndPassengersVehicle(5),ordinaryTruck(6),specialUsesVehicle(7),
                undeterminable(8),invalidData(99)}                      ,
```

```
        calculationAxisNumberMeasurementResult
                                        INTEGER             ,
        calculationLicencePlateNumberMeasurementResult
                                        OCTET STRING        ,
        calculationAxleLoadMeasurementResult
                                        INTEGER             ,
        calculationVehicleHeightMeasurementResult
                                        INTEGER             ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

FreightOperationSituationInfo ::=SEQUENCE{
        movableFreightOperationSituationInfo  UTF8String    ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsRoadAdministratorInfo::=CHOICE{
        dsAlertSystem                   DsAlertSystem       ,
        dsRecoveryVehicleInfo           DsRecoveryVehicleInfo ,
        dsDisasterRecovery              DsDisasterRecovery  ,
        dsInformationBoardProvidingInfo DsInformationBoardProvidingInfo
        }

DsAlertSystem::= SEQUENCE{
        informationCollectInfo          DsOrganization      OPTIONAL,
        informerInfo                    DsOrganization      OPTIONAL,
        eventStatusCode                 ENUMERATED{
                detailUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        startDateTime                   DsDateTime          ,
        inputDateTime                   DsDateTime          ,
        eventOfficeAlertSystemCode      ENUMERATED  {
                preliminaryAlert(1),alert(2),highAlert(3),emergency(4),invalidData(9)},
        eventOfficeSystemName           UTF8String          ,
        regionalConstructionBureauDateTime  DsDateTime      ,
        regionalConstructionBureauInputTime DsDateTime      ,
        eventRegionalConstructionBureauSystemCode
                                        ENUMERATED {
```

```
            preliminaryAlert(1),alert(2),highAlert(3),emergency(4),invalidData(9)},
        eventRegionalConstructionBureauSystemName
                                        UTF8String          ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsRecoveryVehicleInfo::= SEQUENCE{
        informationCollectInfo          DsOrganization      OPTIONAL,
        informerInfo                    DsOrganization      OPTIONAL,
        eventStatusCode                 ENUMERATED{
            detailUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
        movableVehicleCode              INTEGER             ,
        eventConstructionEquipmentNameOwnedByConstructionOffice
                                        ENUMERATED {
            policeVehicle(1),crane(2),roadSignCleaningVehicle(3),debrisCarryingShip(4),
            seaSurfaceCleaningShip(5),dustCarryingSystem(6),weedingVehicleLarge(7),
            weedingMachineSmall(8),waterSprinklingVehicle(9),roadCleaningVehicle(10),
            crashBarrierCleaningVehicle(11),drainCleaningVehicle(12),
            sideDrainCleaningVehicle(13),tunnelCleaningVehicle(14),liftingVehicle(15),
            bridgeMaintenanceVehicle(16),skidResistanceTestVehicle(17),
            roadMaintenanceVehicleFlexMeasurement(18),
            examinationVehicleAmphibian(19),motorizedGrader(20),forkLiftTruck(21),
            lightingVehicle(22),sandBagMakingVehicle(23),
            damageRecoveryVehicleMultiPurposeType(24),
            damageRecoveryVehicleHeadquarter(25),satelliteCommunicationVehicle(26),
            pumpVehicle(27),portableBridge(28),areaInChargeCheckingMachine(29),
            runningVehicleWeightMeter(30),invalidData(99)}  ,
        movableOnboardDeviceID          INTEGER             ,
        organizationOrganizationCode    DsOrganization      ,
        relationLinkageIdentifier       INTEGER             OPTIONAL
        }

DsDisasterRecovery::= SEQUENCE{
        informationCollectInfo          DsOrganization      OPTIONAL,
        informerInfo                    DsOrganization      OPTIONAL,
        eventStatusCode                 ENUMERATED{
            detailUnknown(0),plan(1),expected(2),occurrence(3),finished(4),invalidData(9)},
```

```
pointLocation                   DsLocationInfo          ,
collectionDateTime              DsDateTime              ,
eventMaterialTypeLoadedIntoRestorationVehicle
                                ENUMERATED{
        typeOfMaterialCarriedOnRestorationVehicle}      ,
eventMaterialAmountLoadedIntoRestorationVehicle
                                INTEGER                 ,
eventNameOfDisasterMeasureConstructionEquipmentAccordingToDeploymentPlan
                                ENUMERATED {
        damageRecoveryVehicleMultiPurposeType(1),
        damageRecoveryVehicleHeadquarter(2),pumpVehicle(3),
        sandBagMakingMachine(4),sandBagMakingVehicle(5),lightingVehicle15KVA(6),
        lightingVehicle25KVA(7),portableBridge(8),operationsVehicle(9),
        bridgeMaintenanceVehicle(10),invalidData(99)}   ,
eventNumberOfDisasterMeasureConstructionEquipmentAccordingToDeploymentPlan
                                INTEGER                 ,
eventConstructionEquipmentNameOwnedByConstructionOffice
                                ENUMERATED {
        policeVehicle(1),crane(2),roadSignCleaningVehicle(3),debrisCarryingShip(4),
        seaSurfaceCleaningShip(5),dustCarryingSystem(6),weedingVehicleLarge(7),
        weedingMachineSmall(8),waterSprinklingVehicle(9),roadCleaningVehicle(10),
        crashBarrierCleaningVehicle(11),drainCleaningVehicle(12),
        sideDrainCleaningVehicle(13),tunnelCleaningVehicle(14),liftingVehicle(15),
        bridgeMaintenanceVehicle(16),skidResistanceTestVehicle(17),
        roadMaintenanceVehicleFlexMeasurement(18),
        examinationVehicleAmphibian(19),motorizedGrader(20),forkLiftTruck(21),
        lightingVehicle(22),sandBagMakingVehicle(23),
        damageRecoveryVehicleMultiPurposeType(24),
        damageRecoveryVehicleHeadquarter(25),satelliteCommunicationVehicle(26),
        pumpVehicle(27),portableBridge(28),areaInChargeCheckingMachine(29),
        runningVehicleWeightMeter(30),invalidData(99)}  OPTIONAL,
eventNumberOfConstructionEquipmentsOwnedByConstructionOffice
                                INTEGER                 OPTIONAL,
relationLinkageIdentifier       INTEGER                 OPTIONAL
}

DsInformationBoardProvidingInfo::= SEQUENCE{
```

```
        informationCollectInfo          DsOrganization                  OPTIONAL,
        informerInfo                    DsOrganization                  OPTIONAL,
        provisionStartTime              DsDateTime                      OPTIONAL,
        provisionUpdateEndTime          DsDateTime                      OPTIONAL,
        provisionDeviceMngInfo          MDeviceMngInfo                  ,
        provisionDeviceLocationInfo     DsLocationInfo                  ,
        provisionDeviceStatus           DeviceControlAnswerConfirmInfo,
        provisionContentsInfo           SEQUENCE OF ProvisionContents,
        relationLinkageIdentifier       INTEGER                         OPTIONAL
        }

ProvisionContents::= SEQUENCE{
        watchInfoProvisionInfoDef       INTEGER                         OPTIONAL,
        watchInfoProvisionInfoDefSub    INTEGER                         OPTIONAL,
        watchInfoDisplayCharacterContent UTF8String                     OPTIONAL,
        imageContents                   ImageInfo                       OPTIONAL
        }

ImageInfo::= SEQUENCE{
        imageDataFormat                 ENUMERATED {
                unknown(0),gif(1),png(2),jpeg(3),jpeg2000(4),tiff(5),bmp(6),pict(7),pcx(8),
                invalidData(9999)}                                      ,
        imageDataBody                   OCTET STRING
        }

DsCommercialVehicleOperatorsData ::= CHOICE{
        specialTruckInfo                SpecialTruckInfo
        --serviceStatusInfo
        }

SpecialTruckInfo ::= CHOICE{
        applicationStInfo               ApplicationStInfo               ,
        permissionStInfo                PermissionStInfo
        }

ApplicationStInfo ::= SEQUENCE{
        specialtruckApplicantPersonName OCTET STRING                    ,
```

```
dateDates                          DsDateTime          ,
frontVehicleNumber                 DsVehicleNumber     ,
rearVehicleNumber                  DsVehicleNumber     ,
specialtruckVehicleTypeApplication ENUMERATED{
        truck(1),constructionMachine(2),semiTrailerHeavy(3),
        semiTrailerShippingContainerClassified(4),
        semiTrailerShippingContainer(5),semiTrailerVanType(6),
        semiTrailerTankType(7),semiTrailerTopType(8),
        semiTrailerContainer(9),semiTrailerCarCarrier(10),
        semiTrailerOtherType(11),poleTrailer(12),fullTrailer(13),
        doublesTrailer(14),newSpecificationVehicle(15),invalidData(99)},
specialtruckVehicleWidthApplication  INTEGER     ,
specialtruckVehicleHeightApplication INTEGER     ,
specialtruckVehicleLengthApplication INTEGER     ,
specialtruckGrossWeightApplication   INTEGER     ,
specialtruckMaximumAxialWeightApplication
                                     INTEGER     ,
specialtruckAdjointingAxialWeightApplication
                                     INTEGER     ,
specialtruckCargoNameApplication   ENUMERATED  {
        truckCrane(101),constructionMachineExceptTruckCrane(102),
        bus(103),dumpTruckForOffRoad(104),powerSourceVehicle(105),
        vehicleBody(106),vehicleOthers(107),constructionMachine(201),
        carForPersonalUse(202),powerSourceVehicleBody(203),
        vehicleOnTruckTrailerOthers(204),shippingContainerBox(301),
        shippingContainerTank(302),container(303),jRContainer(304),
        steelBridgeGirder(401),steelPipe(402),steelPlate(403),rail(404),
        shapedSteelHShapedAluminiumSteel(405),
        heavyPlateCopperAluminium(406),coilsSteelAluminium(407),
        copperProductOthersCopperContainerCastIronProduct(408),
        concreteBridgeGirder(501),concreteStake(502),
        prefabricatedHouseParts(503),utilityPole(504),boxCulvert(505),
        fumePipe(506),concreteProductOthers(507),
        industrialMachinePlantMachineMachineToolMetalWorkingMachineMachineFra
        me(601),maintenanceMachine(602),rotationFurnace(603),
        rotaryFurnactOthersTankWeldingMachine(604),
        volatileOilPetrolLightOilParaffin(701),
```

```
                liquefiedProductLPGasHydrogenOxygen(702),
                petrochemicalProductOthersPhenolPolyestersPetrochemicalProducts(703),
                generator(801),transformer(802),pump(803),airBlower(804),
                wireCableDrum(805),householdElectricalProducts(806),
                electricProductsOthers(807),timber(901),woodenProducts(902),
                trees(903),woodOthers(904),farmProducts(1001),
                marineProducts(1002),feed(1003),foodOthers(1004),
                generalGoods(1101),cement(1102),rollPaper(1103),othersOthers(1104),
                invalidData(9999)}                                  ,
        specialtruckCargoWidthApplication   INTEGER                 ,
        specialtruckCargoHeightApplication  INTEGER                 ,
        specialtruckCargoLengthApplication  INTEGER                 ,
        movableTruckCargoLoad               INTEGER                 ,
        roadRoute                           DsRouteLocation         ,
        specialtruckApplicationContent      UTF8String              ,
        relationLinkageIdentifier           INTEGER                 OPTIONAL
        }

PermissionStInfo ::= SEQUENCE{
        specialtruckPermissionPerson        OCTET STRING            ,
        dateDatesTimeStart                  DsDateTime              ,
        dateDatesTimeEnd                    DsDateTime              ,
        specialtruckPermissionCondition     UTF8String              ,
        specialtruckPermissionNumber        INTEGER                 ,
        frontVehicleNumber                  DsVehicleNumber         ,
        rearVehicleNumber                   DsVehicleNumber         ,
        specialtruckVehicleTypePermission   ENUMERATED{
                truck(1),constructionMachine(2),semiTrailerHeavy(3),
                semiTrailerShippingContainerClassified(4),
                semiTrailerShippingContainer(5),semiTrailerVanType(6),
                semiTrailerTankType(7),semiTrailerTopType(8),
                semiTrailerContainer(9),semiTrailerCarCarrier(10),
                semiTrailerOtherType(11),poleTrailer(12),fullTrailer(13),
                doublesTrailer(14),newSpecificationVehicle(15),invalidData(99)},
        specialtruckVehicleWidthPermission  INTEGER                 ,
        specialtruckVehicleHeightPermission INTEGER                 ,
        specialtruckVehicleLengthPermission INTEGER                 ,
```

```
specialtruckGrossWeightPermission      INTEGER                    ,
specialtruckMaximumAxialWeightPermission
                                       INTEGER                    ,
specialtruckAdjointingAxialWeightPermission
                                       INTEGER                    ,
specialtruckCargoNamePermission        ENUMERATED{
        truckCrane(101),constructionMachineExceptTruckCrane(102),
        bus(103),dumpTruckForOffRoad(104),powerSourceVehicle(105),
        vehicleBody(106),vehicleOthers(107),constructionMachine(201),
        carForPersonalUse(202),powerSourceVehicleBody(203),
        vehicleOnTruckTrailerOthers(204),shippingContainerBox(301),
        shippingContainerTank(302),container(303),jRContainer(304),
        steelBridgeGirder(401),steelPipe(402),steelPlate(403),rail(404),
        shapedSteelHShapedAluminiumSteel(405),
        heavyPlateCopperAluminium(406),coilsSteelAluminium(407),
        copperProductOthersCopperContainerCastIronProduct(408),
        concreteBridgeGirder(501),concreteStake(502),
        prefabricatedHouseParts(503),utilityPole(504),boxCulvert(505),
        fumePipe(506),concreteProductOthers(507),
        industrialMachinePlantMachineMachineToolMetalWorkingMachineMachineFra
        me(601),maintenanceMachine(602),rotationFurnace(603),
        rotaryFurnactOthersTankWeldingMachine(604),
        volatileOilPetrolLightOilParaffin(701),
        liquefiedProductLPGasHydrogenOxygen(702),
        petrochemicalProductOthersPhenolPolyestersPetrochemicalProducts(703),
        generator(801),transformer(802),pump(803),airBlower(804),
        wireCableDrum(805),householdElectricalProducts(806),
        electricProductsOthers(807),timber(901),woodenProducts(902),
        trees(903),woodOthers(904),farmProducts(1001),marineProducts(1002),
        feed(1003),foodOthers(1004),generalGoods(1101),cement(1102),rollPaper(1103),
        othersOthers(1104), invalidData(9999)}     ,
specialtruckCargoWidthPermission       INTEGER                    ,
specialtruckCargoHeightPermission      INTEGER                    ,
specialtruckCargoLengthPermission      INTEGER                    ,
movableTruckCargoLoad                  INTEGER                    ,
roadRoute                              DsRouteLocation            ,
specialtruckApplicationContent         UTF8String                 ,
```

```
        relationLinkageIdentifier                INTEGER          OPTIONAL
        }
END

RCS-data-dictionary
DEFINITIONS  AUTOMATIC TAGS  EXTENSIBILITY IMPLIED ::=
BEGIN
EXPORTS
DatesDateOfYear,DatesUnitOfTime,LocationXYNormalCoordinates,LocationLatitudeLongitudeDegre
e;

DatesDateOfYear ::= SEQUENCE{
        datesYear INTEGER(1900..2155),
        datesMonth INTEGER(1..12),
        datesDate INTEGER(1..31)
        }

DatesUnitOfTime ::= SEQUENCE{
        datesHour  INTEGER(0..23),
        datesMinute  INTEGER(0..59),
        datesSecond  INTEGER(0..59),
        datesMilliSecond  INTEGER(0..999) OPTIONAL
        }

LocationXYNormalCoordinates ::= SEQUENCE{
        locationSecondaryCoordinatesCode  INTEGER(0..999999),
        locationXNormalCoordinates  INTEGER(0..10000),
        locationYNormalCoordinates  INTEGER(0..10000)
        }

LocationLatitudeLongitudeDegree ::= SEQUENCE{
        locationLatitudeDegree  INTEGER(-90000000..90000000),
        locationLongitudeDegree  INTEGER(-180000000..180000000)
        }

END
```