

团体标准

T/ITS 0293.3-XXXX

自主式交通系统 交通语义表示语言 第3部分：交互框架

Autonomous transportation system

Traffic semantic representation language part 3: Interaction framework

(征求意见稿)

(本稿完成时间：2025年9月)

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

20XX-XX-XX 发布

20XX-XX-XX 实施

中国智能交通产业联盟 发布

中国智能交通产业联盟

目 次

前 言 1

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 缩略语 1

5 语义信息交互框架 2

6 交通语义交互行为 6

附 录 A （资料性附录） 道路交通场景语义信息交互示例 14

附 录 B （资料性附录） 轨道交通场景语义信息交互示例 18

附 录 C （资料性附录） 水运交通场景语义信息交互示例 21

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中国智能交通产业联盟（C-ITS）提出并归口。

本文件主要起草单位：待定。

本文件主要起草人：待定。

自主式交通系统 交通语义表示语言 第3部分：交互框架

1 范围

本文件规定了交通语义表示语言的交互框架，包括交互流程、交互消息的层次化结构以及交通语义交互行为。

本文件适用于采用交通语义表示语言的主体在交互过程中实现语义通信，包括但不限于人、载运装备、基础设施和交通管控中心等交通主体间的语义交互场景。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

T/ITS 0098-2017 合作式智能运输系统 通信架构

T/CSAE 53—2020 合作式智能运输系统 车用通信系统应用层及应用数据交互标准（第一阶段）

3 术语和定义

下列术语和定义适用于本文件。

3.1

自主式交通系统 autonomous transportation system

自主式交通系统是以自感知、自适应、自学习、自组织为特征的高度自治的交通系统。

3.2

交通语义表示语言 traffic semantic representation language

交通语义表示语言是用于描述和定义自主式交通系统中交通行为、规则、状态及交互的规范化语言

3.3

交通语义信息交互 traffic semantic information interaction

人、载运装备、基础设施、交通管控中心这四类交通主体通过交互框架传递语义信息，实现可理解的信息交互。

4 缩略语

- FV: 前方车辆 (Front Vehicle)
- HV: 主车 (Host Vehicle)
- ID: 标识 (Identification)
- MF: 消息帧 (Message Frame)
- OBU: 车载单元 (Onboard Unit)
- RSU: 路侧单元 (Road Side Unit)
- RV: 后方车辆 (Rear Vehicle)
- TCP: 传输控制协议 (Transmission Control Protocol)
- UDP: 用户数据报协议 (User Datagram Protocol)

5 交通语义信息交互框架

5.1 交互框架基本介绍和要求

本文件关注的是语义信息交互框架，包括交互流程和交互消息，对应于消息层进行语义信息交互的基本流程和消息的结构形式，可承载符合第1、2部分规范的语义表示内容。本文件通过对交通主体不同的交互动作行为分析，定义在语义信息交互过程中，不同交通方式下人-载运装备-基础设施-交通管控中心这些自主式交通主体通用的交互流程与方式，使得所涉及的交通主体都能在准确的时机进行语义信息交互（包括请求、响应、拒绝等），从而使得整个交互过程能够准确、及时完成，以支持应用层的各项应用。本文件不指定底层的通信方式，可以用于各种不同的网络层、接入层技术，帮助异构交通主体在语义层面实现统一的信息交互。

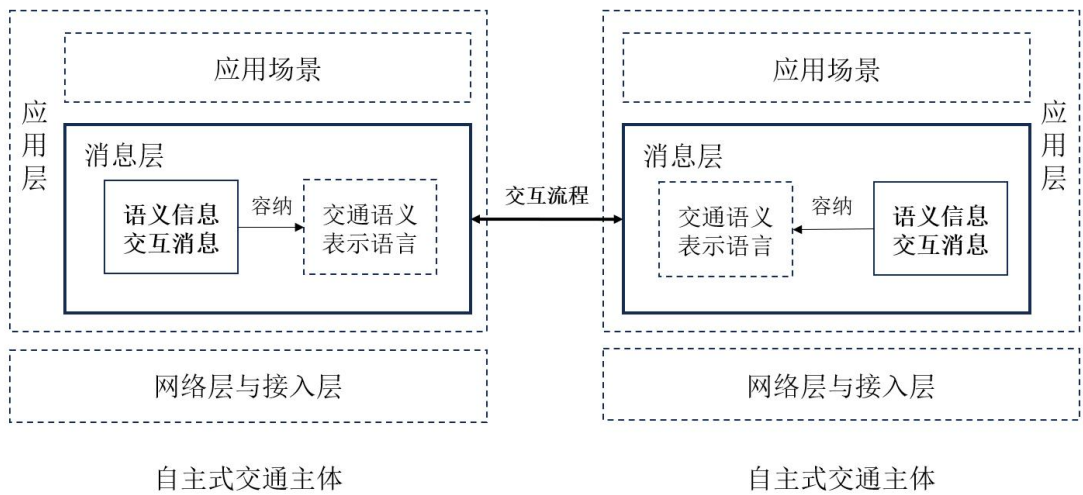


图1 交通语义信息交互框架

交互消息遵循“消息帧-消息体-消息参数”层层嵌套的逻辑进行制定。

5.2 语义信息交互基本流程

图2说明了多交通主体进行语义信息交互应满足的基本流程。图2包含两个交通主体：交通主体A和交通主体B，其中A是会话的发起者，它最先发送消息。图2显示交通主体语义交互的总体过程：交通主体A期望达成目标G1，因此形成特定意图I1，决定采取语义交互行为Act1，将该行为形成消息M，并对消息M编码，使其满足底层通讯协议与服务的要求，传输给交通主体B；交通主体B接收消息后，解码出消息M，并根据事先定义的规范以及自身目标G2与意图I2，采取语义交互行为Act2并选择是否回复消息N，，同时也可能执行某些操作O，此后重复这一流程。

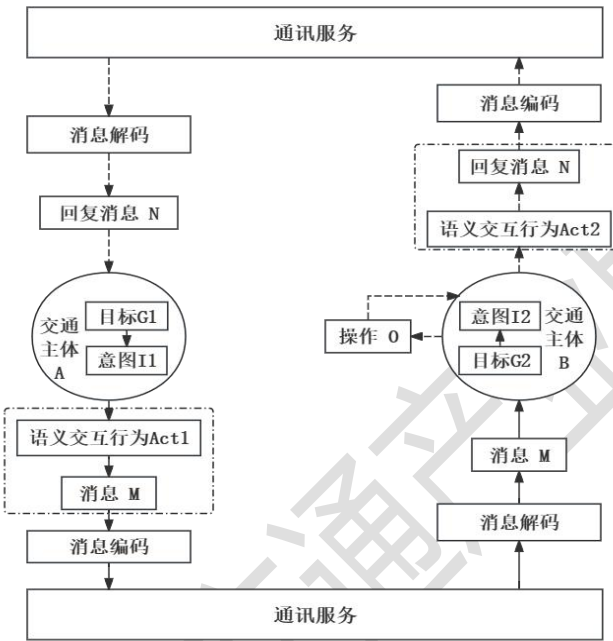


图2 多交通主体语义信息交互基本流程

5.3 语义信息交互消息

本文件定义的语义信息交互消息帧、消息体以及消息参数的层次结构，如图3所示。



图3 语义信息交互消息层次结构示意图

5.3.1 消息帧

一个消息帧可以容纳一个或多个消息体。消息帧结构形式如表1所示。消息体包含一系列参数，如表2。其中，消息体与消息内容弱耦合，消息体只控制包含的消息内容的描述语言与编码方式。消息内容的具体实现不在本文件的范围之内。

表1 消息帧结构形式

类型	表示形式	说明
消息帧	<pre> MessageFrame ::= SEQUENCE { Message1 Message2 Message3 ... } </pre>	MessageFrame 为消息帧名；SEQUENCE 表示后续内容依次序，“{}”中为包含的消息体；Message1 等为消息体名；消息体的内容在此处省略。

5.3.2 消息体

消息体用于传输语义消息，其结构规范如表2所示：

表2 消息体结构规范表

类型	表示规范	说明
消息体	<pre> MessageName ::= SEQUENCE{ Message- Identifier (<number>) Performative (<performative>) : Sender (<subject>) : Receiver (<subject>) : Reply-To (<subject>) : Content “<expression>” : Language (<language>) : Ontology (<ontology>) : Protocol (<protocol>) : Conversation- Identifier (<number>) : Reply-With (<expression>) : In-Reply-To (<expression>) : Reply-By (<time>) } </pre>	MessageName 为消息体名，“{}”中为包含的消息参数，消息参数的内容由< >中的对象表示。

5.3.3 消息体参数

消息体应包含一个或多个消息参数，如表2。消息体的类别与所需参数的种类及个数由参数“Performative”决定，该参数实质上代表了语义交互行为类型。

消息体参数支持扩展。除了可以使用表3中指定的消息参数之外，在面向具体实现时，用户可以自由地定义消息体参数。这些非标准附加参数的名称必须使用前置字符串“X-”。

表3 消息体参数表

参数类别	参数	说明
消息体的名称	Message-Identifier	表示消息体标识符。
交互的行为类型	Performative	表示消息的类型，对应于语义消息交互的行为类型，。
交互的参与者	Sender	表示消息的发送者。
	Receiver	表示消息的预期接收者的身份。
	Reply-To	表示后续消息的回复对象。
消息内容	Content	表示消息的内容，为交互语句。
内容描述	Language	表示消息语言，表达参数“Content”消息内容所使用的语言。
	Ontology	表示用于赋予参数“Content”消息内容中的符号含义的本体。
交互控制	Protocol	表示通信协议
	Conversation-Identifier	表示会话标识符。
	Reply-With	表示响应标识符。
	In-Reply-To	表示回复标识符。
	Reply-By	答复最晚时间。

各个参数的具体说明如下：

- a) Message-Identifier。消息体标识符。用于唯一地表示消息帧中的消息体。可以是数字或某种编码。“Message-Identifier”是所有交互消息的必需参数；
- b) Performative。述行词。表示消息交互的行为类型，同时也体现出消息的类型。“Performative”是所有交互消息的必需参数。此外，不同的述行词决定了消息体的参数数量与类型；
- c) Sender。发送者。表示消息发送者的身份。“sender”参数是大多数消息的参数。但是，如果发送消息的主体希望保持匿名，则可以省略发送者参数；
- d) Receiver。接收者。表示消息的预期接收者的身份。“Receiver”参数可以是单个交通主体或多个主体的非空集合。后者对应于消息多播的情况；
- e) Reply-To。消息回复对象。该参数指示此会话中的后续消息将定向到“Reply-To”参数中指定的主体，而不是发送到“Sender”参数中指定的主体；
- f) Content。消息内容。表示消息的内容，为交通语义表示语言的语句。任何消息的内容含义均由消息的接收方进行解释；
- g) Language。消息语言。表示表达参数“Content”消息内容所使用的语言。如果已知接收消息的主体知道表达消息内容所使用的语言，则可以省略该字段。
- h) Ontology。本体。表示用于赋予参数“Content”消息内容中的符号含义的本体。当发送者与

接收者都已知该参数值时，可省略。在交通领域，“Ontology”可指不同交通方式下各类符号及其含义的本体。例如，道路交通本体“RoadTraffic”，轨道交通主体“RailTransit”，水运交通本体“WaterTransportation”，它们应定义各自交通方式中各类符号及其含义；

- i) Protocol。通信协议。表示发送代理在消息中采用的底层通信协议，如 TCP、UDP 等；
- j) Conversation-Identifier。会话标识符。表示正在进行的消息序列所属哪个会话。交通主体可以使用会话标识符来标记消息，以管理其通信策略和活动。允许交通主体与多个主体进行单独会话。需要对 Conversation-id 参数使用全局唯一值，以便参与者能够区分多个并发会话；
- k) Reply-With。响应标识符。表示响应主体将使用该表达式来识别此消息。“Reply-With”参数设计用于在多个对话同时发生的情况下跟踪对话线程。例如，如果主体 i 向主体 j 发送一条消息，其中包含：

Reply-With <expression>

主体 j 将回复一条消息，其中应包含：

In-Reply-To <expression> ；

- l) In-Reply-To。回复标识符。表示该消息为此前较早消息的回复消息。用法如上所示；
- m) Reply-By。最晚答复时间。表示发送者希望接收者答复的最晚时间。

6 交通语义交互行为

6.1 交互行为类型

在不同的交通方式下，交通主体参与的交通场景有很大的差异，但这些场景大多需要人、载运装备、基础设施与交通管控中心进行协同配合才能实现。从人-载运装备-基础设施-交通管控中心之间进行信息交互的场景中，抽象出交通主体进行信息交互的动作行为，分为信息交互、协作控制、错误处理三类交互类型，包括告知、查询、请求、重复执行、接受、拒绝、失败等行为，具体对应于告知消息、查询消息、请求消息、重复执行消息、接受消息、拒绝消息、失败消息、困惑消息等语义消息，如表 4 所示。

表4 语义消息分类表

语义消息类型	行为类型	交互类型
告知消息	告知	信息交互
查询消息	查询	
请求消息	请求	协作控制
重复执行消息	重复执行	
接受消息	接受	
拒绝消息	拒绝	
失败消息	失败	错误处理
困惑消息	困惑	

6.2 对应的语义消息结构

按照ASN.1标准，对信息交互、协作控制以及错误处理这三类语义信息交互类型中包含的交互行为进行结构语法定义。这些交互行为对应于消息参数“Performative”的值，并决定了消息体的结构，即其需要的参数种类及个数。为表示简洁，假设交互主体双方均已知消息语言、表示符号含义的本体以及使用的通信协议，每类消息体均省略参数“Language”，“Ontology”，“Protocol”。在附录A，B，C中，对道路、轨道、水运三种交通方式下应用该方法进行交通主体语义信息交互做了详细说明。

6.2.1 信息交互

a) Inform。告知行为。该行为表示发送者告知接收者某些信息。

【语法规范】

```
MessageName {
    Message-Identifier (<number>)
    Performative (Inform)
    : Sender (<subject>)
    : Receiver (<subject>)
    : Content
    “<expression>”
    : Conversation-Identifier (<number>)
    : In-Reply-To (<expression>)
}
```

【示例】路侧单元 RSU 向周围车辆 Vehicle1, Vehicle2, Vehicle3 等传达地图数据

```
InformMessage {
    Message-Identifier (1)
    Performative (Inform)
    : Sender (RSU)
    : Receiver (Vehicle1, Vehicle2, Vehicle3, ...)
    : Content
    “MapData”
    : Conversation-Identifier (1)
    : In-Reply-To (Map_Data01)
}
```

b) Query。询问行为。该行为表示发送者希望接收者答复某些信息。

【语法规范】

```
MessageName {
    Message-Identifier (<number>)
```

```
Performative (Query)
: Sender (<subject>)
: Receiver (<subject>)
: Reply-To (<subject>)
: Content
  "<expression>"
: Conversation-Identifier (<number >)
: Reply-With (<expression>)
: Reply-By (<time>)
}
```

【示例】 车辆询问信号灯当前颜色

```
QueryMessage {
  Message-Identifier (1)
  Performative (Query)
  : Sender (Vehicle)
  : Receiver (TrafficLight)
  : Content
    "TrafficLight"
  : Conversation-Identifier (1)
  : Reply-With (traffic_light01)
  : Reply-By (1s)
}
```

6.2.2 协作控制

a) Request。请求行为。表示发送者请求接收者执行某些行为。

【语法规范】

```
MessageName {
  Message-Identifier (<number>)
  Performative (Request)
  : Sender (<subject>)
  : Receiver (<subject>)
  : Content
    "<expression>"
  : Conversation-Identifier (<number >)
```

```
    : Reply-With (<expression>)
    : Reply-By (<time>)
}
```

【示例】 自车 HV 请求后车 RV 减速

```
RequestMessage {
    Message-Identifier (1)
    Performative (Request)
    : Sender (HV)
    : Receiver (RV)
    : Content
      "SlowSpeed(RV)"
    : Conversation-Identifier (1)
    : Reply-With (speed01)
    : Reply-By (1s)
}
```

- b) Request-whenEVER. 重复执行。表示发送者希望接收者在某个命题变为真时立即执行某些操作，此后每次该命题再次变为真时，再次执行同样操作。

【语法规范】

```
MessageName {
    Message-Identifier (<number>)
    Performative (Request-whenEVER)
    : Sender (<subject>)
    : Receiver (<subject>)
    : Content
      "<expression>"
    : Conversation-Identifier (<number >)
    : Reply-With (<expression>)
    : Reply-By (<time>)
}
```

【示例】 路侧单元 RSU 请求车辆 Vehicle 在超过道路 Road 限速时减速

```
RequestWheneverMessage {
    Message-Identifier (1)
    Performative (Request-WhenEVER)
    : Sender (RSU)
```

: Receiver (Vehicle)
: Content
“LimitSpeed(Road, Data)
Decelerate(Vehicle)”
: Conversation-Identifier (1)
: Reply-With (LimitSpeed01)
}

c) Accept。接受行为。表示发送者接受先前收到的要求并以成功执行相关操作。

【语法规范】

MessageName {
 Message-Identifier (<number>)
 Performative (Accept)
 : Sender (<subject>)
 : Receiver (<subject>)
 : Content
 “<expression>”
 : Conversation-Identifier (<number >)
 : In-Reply-To (<expression>)
}

【示例】主车 HV 接受前车 FV 的减速请求，并返回最终速度

AcceptMessage {
 Message-Identifier (1)
 Performative (Accept)
 : Sender (HV)
 : Receiver (FV)
 : Content
 “Decelerate(HV)
 Speed(HV, Data)”
 : Conversation-Identifier (1)
 : In-Reply-To (SlowSpeed01)
}

d) Refuse。拒绝行为。表示发送者拒绝执行某些行为并给出理由。

【语法规范】

MessageName {

```

Message-Identifier (<number>)
Performative (Refuse)
: Sender (<subject>)
: Receiver (<subject>)
: Content
  “<expression>”
: Conversation-Identifier (<number >)
: In-Reply-To (<expression>)

```

```

}

```

【示例】 自车 HV 拒绝前车 FV 发来的减速请求，并给出理由

```

RefuseMessage {
  Message-Identifier (1)
  Performative (Refuse)
  : Sender (HV)
  : Receiver (FV)
  : Content
    “Decelerate(FV)
    Reason(HV, Data)”
  : Conversation-Identifier (1)
  : In-Reply-To (SlowSpeed01)
}

```

6.2.3 异常处理

a) Failure。失败行为。表示发送者告知接收者其已尝试执行某项操作但尝试失败。

【语法规范】

```

MessageName {
  Message-Identifier (<number>)
  Performative (Failure)
  : Sender (<subject>)
  : Receiver (<subject>)
  : Content
    “<expression>”
  : Conversation-Identifier (<number >)
  : In-Reply-To (<expression>)
}

```

}

【示例】主车 HV 告知前车 FV 自身并不可以减速，减速失败

FailureMessage {
 Message- Identifier (1)
 Performative (Failure)
 : Sender (HV)
 : Receiver (FV)
 : Content
 “Decelerate(FV)”
 : Conversation-Identifier (1)
 : In-Reply-To (SlowSpeed01)
}

b) Confuse。困惑行为。表示发送主体无法理解接收信息的内容。

【语法规范】

MessageName {
 Message- Identifier (<number>)
 Performative (Confuse)
 : Sender (<subject>)
 : Receiver (<subject>)
 : Content
 “<expression>”
 : Conversation-Identifier (<number >)
 : In-Reply-To (<expression>)
}

【示例】车辆无法理解从路侧单元 RSU 接收到的信息，对路侧单元反馈

ConfuseMessage {
 Message-Identifier (1)
 Performative (Confuse)
 : Sender (Vehicle)
 : Receiver (RSU)
 : Content
 “Original Message”
 : Conversation-Identifier (1)
 : In-Reply-To (confuse01)

中国智能交通产业联盟

附录 A
(资料性附录)
道路交通场景语义信息交互示例

A.1 道路交通场景描述

在高速公路上行驶的车辆，可能需要进行车道变换以适应不同的行驶要求，比如超车、驶出高速公路或避让慢速行驶的车辆等。车辆变道时需要考虑周围车辆的位置和速度，以确保变道过程中的安全通行。此外，车辆变道可能受到最低限速、最大限速和道路出口位置等因素的影响，因此需要根据具体情况采取适当的行驶策略。这些信息可通过车-车以及车-路侧设施之间的语义信息交互获取。

由于本文件并不指定具体的底层通信协议，此处省略参数“Protocol”。

对于道路场景的信息定义以及场景知识表达参照标准《自主式交通系统 交通语义表示语言 第1部分：通用术语》与《自主式交通系统 交通语义表示语言 第2部分：语法规则》。参数“Language”的值即为交通语义表示语言，记为TSRL，即Traffic Semantic Representation Language，参数“Ontology”为系列标准第1部分定义的道路交通本体库，记为RoadTraffic。

A.2 道路交通主体语义信息交互具体实现

(1) 车车交互。

车辆Vehicle1向车辆Vehicle2询问Vehicle2的位置、速度、距离高速公路出口的距离等信息。

```
QueryMessageV2V {  
    Message- Identifier (1)  
    Performative (Query)  
    : Sender (Vehicle1)  
    : Receiver (Vehicle2)  
    : Content  
    "InLane(Vehicle2, Data),  
    Speed(Vehicle2, Data),  
    DistanceToExit(Vehicle2, Data),  
    ..."  
    : Language (TSRL)  
    : Ontology (RoadTraffic)  
    : Conversation-Identifier (1)  
    : Reply-With (VehicleInfo01)  
    : Reply-By (1s)}
```

车辆 Vehicle2 向车辆 Vehicle1 答复 Vehicle2 的位置、速度、距离高速公路出口的距离等信息。

```
InformMessageV2V {
    Message- Identifier (2)
    Performative (Inform)
    : Sender (Vehicle2)
    : Receiver (Vehicle1)
    : Content
    "Lane(Vehicle2, Data),
    Speed(Vehicle2, Data),
    DistanceToExit(Vehicle2, Data),
    ..."
    : Language (TSRL)
    : Ontology (RoadTraffic)
    : Conversation-Identifier (1)
    : In-Reply-To (VehicleInfo01)
    : Reply-By (1s)
}
```

车辆 Vehicle1 请求车辆 Vehicle2 减速

```
RequestMessage{
    Message-id (3)
    Performative (Request)
    : Sender (Vehicle1)
    : Receiver (Vehicle2)
    : Content
    "Let(Vehicle2, SpeedDown)"
    : Conversation-id (1)
    : Reply-With (VehicleQuest01)
    : Reply-By (1s)
}
```

车辆 Vehicle2 接受车辆 Vehicle1 的减速请求

```
AcceptMessage {
    Message-id (4)
```

```

Performative (Accept)
: Sender (Vehicle2)
: Receiver (Vehicle1)
: Content
  "Let(Vehicle2, SpeedDown)"
: Conversation-id(1)
: In-Reply-To (VehicleQuest01)
  }

```

(2) 车路交互

路侧单元 RSU 向周围车辆 Vehicle1, Vehicle2, Vehicle3 等交通环境数据, 包括道路条件、天气状况、能见度等。

```

InformMessageV2R {
  Message-Identifier (3)
  Performative (Inform)
  : Sender (RSU)
  : Receiver (Vehicle1, Vehicle2, Vehicle3, ...)
  : Content
    "RoadCondition(Road, Condition),
    WeatherCondition(Weather),
    Visibility(Visibility),
    ..."
  : Language (TSRL)
  : Ontology (RoadTraffic)
  : Conversation-Identifier (2)
  : Reply-With (RoadCondition01)
  : Reply-By (1s)
}

```

车辆 Vehicle 向路侧单元 RSU 查询当前道路 Road 的道路数据, 包括道路限速等信息。

```

QueryMessageV2R {
  Message-Identifier (4)
  Performative (Query)
  : Sender (Vehicle)

```

```

: Receiver (RSU)
: Content
  “MaxSpeed (Road, Data),
    MinSpeed (Road, Data),
    ...”
: Language (TSRL)
: Ontology (RoadTraffic)
: Conversation-Identifier (2)
: Reply-With (RoadData01)
: Reply-By (1s)
  }

```

路侧单元 RSU 向车辆 Vehicle 答复当前道路 Road 的道路数据，包括道路限速等信息。

```

InformMessageR2V {
  Message-Identifier (5)
  Performative (Inform)
  : Sender (RSU)
  : Receiver (Vehicle)
  : Content
    “MaxSpeed (Road, Data),
      MinSpeed (Road, Data),
      ...”
  : Language (TSRL)
  : Ontology (RoadTraffic)
  : Conversation-Identifier (2)
  : In-Reply-To (RoadData01)
  : Reply-By (1s)
}

```

附录 B
(资料性附录)
轨道交通场景语义信息交互示例

B.1 轨道交通场景描述

选取轨道交通中的越行场景。列车越行指的是在双线或多线铁路区段上，同方向行驶的两列速度不同的列车，速度快的列车（快车）超过速度慢的列车（慢车）的过程。

在这个场景中，需要考虑列车的安全通行以及站点线路、道岔的交通管理。首先，调度集中系统根据运行计划，向越行站的计算机联锁系统下达越行指令；随后，联锁系统自动操控道岔转换至正确位置，为慢车接入侧线做好准备，并通过轨道电路实时监测线路占用情况；当信号机向慢车显示停车信号引导其驶入侧线停稳后，联锁系统立即为正线锁闭通往安全的通过进路，并向快车显示绿色通行信号；待快车依据信号提示高速通过车站后，轨道电路检测到其完全驶离，调度中心便再次授权联锁系统转换道岔，引导慢车驶回正线继续旅程。在这个场景中，主要涉及联锁系统与轨道电路、信号机与车辆、调度中心与车辆的语义信息交互。

由于本文件并不指定具体的底层通信协议，此处省略参数“Protocol”。

对于轨道场景的信息定义以及场景知识表达参照标准《自主式交通系统 交通语义表示语言 第1部分：通用术语》与《自主式交通系统 交通语义表示语言 第2部分：语法规则》。参数“Language”的值即为该标准定义的语言，记为TSRL，即Traffic Semantic Representation Language。参数“Ontology”为该标准定义的轨道交通本体库，记为RoadTraffic。

B.2 轨道交通主体语义信息交互具体实现

(1) 联锁系统与轨道电路交互。

联锁系统要求轨道电路变更道岔位置。

```
QuestMessageIS2TC {  
    Message-Identifier (1)  
    Performative (Quest)  
    : Sender (InterlockingSystem)  
    : Receiver (TrackCircuit)  
    : Content  
    "Let(Turnout, State1),  
    ..."  
    : Language (TSRL)  
    : Ontology (RailTransit)  
    : Conversation-Identifier (1)  
    : Reply-With (TrainQuest01)  
    : Reply-By (1s)  
}
```

轨道电路向联锁系统告知快车与慢车的位置、速度、加速度等信息。

```

InformMessageTC2IS {
    Message-Identifier (2)
    Performative (Inform)
    : Sender (TrackCircuit)
    : Receiver (InterlockingSystem)
    : Content
    “Position(PriorityTrain, Data),
    Speed(PriorityTrain, Data),
    Acceleration(PriorityTrain, Data),
    ...”
    : Language (TSRL)
    : Ontology (RailTransit)
    : Conversation-Identifier (1)
    : In-Reply-To (TrainQuest01)
    : Reply-By (1s)
}

```

(2) 列车与信号设备交互。

信号设备向慢车广播停车信号以及道岔的开合状态。

```

InformMessageSE2ST {
    Message-Identifier (3)
    Performative (Inform)
    : Sender (SignalEquipment)
    : Receiver (SubordinateTrain)
    : Content
    “SignalState(SignalEquipment, Data),
    TurnoutState(Turnout, Data)
    ...”
    : Language (TSRL)
    : Ontology (RailTransit)
    : Conversation-Identifier (2)
    : In-Reply-To (SignalInfo01)
}

```

```

: Reply-By (1s)
}

```

(3) 列车与调度中心交互。

调度中心请求慢车减速并停在站内的侧线上。

```

QuestMessageDC2ST {
    Message-Identifier (4)
    Performative (Quest)
    : Sender (DispatchCenter)
    : Receiver (SubordinateTrain)
    : Content
    "Let(SubordinateTrain, Stop),
    ..."
    : Language (TSRL)
    : Ontology (RailTransit)
    : Conversation-Identifier (3)
    : Reply-With (DispatchCenterQuest01)
    : Reply-By (1s)
}

```

慢车接受调度中心 DispatchCenter 的请求

```

AcceptMessageST2DC {
    Message-id (5)
    Performative (Accept)
    : Sender (TrainA)
    : Receiver (DispatchCenter)
    : Content
    "Let(SubordinateTrain, Stop),
    ..."
    : Conversation-id(3)
    : In-Reply-To (DispatchCenterQuest01)
}

```

附录 C
(资料性附录)
水运交通场景语义信息交互示例

C.1 水运交通场景描述

选取水运交通场景，考虑两艘船在航道上相遇的情况。航道可以是海洋、河流或其他水域中的指定航行通道。船舶A和船舶B在航道上沿着各自的航线航行，这导致它们在某一时刻会在航道上相遇或交叉。由于航道空间有限，这种相遇可能会带来潜在的碰撞风险。因此，在船舶相遇时，必须采取适当的航行规则和安全措施，以确保船舶之间的安全通行。在这一过程中，船舶的速度、航向和位置都是动态变化的，船舶之间的距离也会随着时间变化而变化。船舶相遇时的航行决策需要考虑诸多因素，包括航行规则、船舶动态特性以及海上交通情况，以最大程度地降低碰撞风险，并保障水运交通的安全和效率。

由于本文件并不指定具体的底层通信协议，此处省略参数“Protocol”。

对于水运场景的信息定义以及场景知识表达参照标准《自主式交通系统 交通语义表示语言 第1部分：通用术语》与《自主式交通系统 交通语义表示语言 第2部分：语法规则》。参数“Language”的值即为该标准定义的语言，记为TSRL，即Traffic Semantic Representation Language。参数“Ontology”为该标准定义的水运交通本体库，记为WaterTransportation。

C.2 水运交通主体语义信息交互具体实现

(1) 船船交互。

船只 Ship1 向船只 Ship2 询问 Ship2 的位置、速度等信息

QueryMessageS2S {
Message-Identifier (1)
Performative (Query)
: Sender (Ship1)
: Receiver (Ship2)
: Content
“Position (Ship2, Data),
Speed(Ship2, Data),
...”
: Language (TSRL)
: Ontology (WaterTransportation)
: Conversation-Identifier (1)
: Reply-With (ShipQuery01)

```
: Reply-By (1s)
}
```

船只 Ship2 向船只 Ship1 答复 Ship2 的位置、速度、等信息

```
InformMessageS2S {
    Message-Identifier (2)
    Performative (Inform)
    : Sender (Ship2)
    : Receiver (Ship1)
    : Content
    "Position(Ship2, Data),
    Speed(Ship2, Data),
    ..."
    : Language (TSRL)
    : Ontology (WaterTransportation)
    : Conversation-Identifier (1)
    : In-Reply-To (ShipQuery01)
    : Reply-By (1s)
}
```

(2) 船只与航道中的传感设备交互。

航道中的传感设备 SensingDevice 向船只 Ship1、Ship2 等广播航道长度、宽度、天气状况等信息。

```
InformMessageSD2S {
    Message-Identifier (3)
    Performative (Inform)
    : Sender (SensingDevice)
    : Receiver (Ship1, Ship2, ...)
    : Content
    "ChannelLength(Channel, Data),
    ChannelWidth(Channel, Data),
    WeatherCondition(Data),
    ..."
    : Language (TSRL)
    : Ontology (WaterTransportation)
}
```

: Conversation-Identifier (2)
: In-Reply-To(ChannelInfo01)
: Reply-By (1s)
}

中国智能交通产业联盟

中国智能交通产业联盟

标准

自主式交通系统 交通语义表示语言 第3部分：交互框架

T/ITS 0293.3-XXXX

北京市海淀区西土城路8号（100088）

中国智能交通产业联盟印刷

网址：<http://www.c-its.org.cn>

2025年X月第一版 2025年X月第一次印刷