Hogia
Public Transport Systems

POSition Real-time Output Interface 1.0
Interface Specification

Document identity: IS-PT/I/POSROI/1
Revision: PA5
Date: 2012-04-27

POSition Real-time Output Interface 1.0 Interface Specification

Copyright

Revision History

Revision	Date	Update	Updated by
PA1	2012-04-25		Ulf Bjersing
PA2	2012-04-26	Corrections	Ulf Bjersing
PA3	2012-04-27	Adjustments	Ulf Bjersing
PA4	2012-04-27	Corrections	Ulf Bjersing
PA5	2012-04-27	Clarifications	Ulf Bjersing

Interface Version History

A leading P in the document revision indicates a version under preparation. A – (minus) indicates that the document was not part of the version of the interface. A / (slash) indicates that the document is no longer used in the version of the interface.

Document/product identity	**3.0**
IS-PT/I/POSROI/1	A

Table of content

1 Introduction

The purpose of this document is to describe PubTrans POSition Real-time Output Interface (POSROI) and how clients can utilise the content and functionality of POSROI.

1.1 What is POSROI?

POSROI is one of the PubTrans public interfaces. With POSROI it is possible for client applications to receive current vehicle positions combined with real time data from the public transport operation. The data is provided in a uniform format, regardless of which source system it came from.

POSROI can be used for many types of purposes where there is use for exposing current vehicle positions for vehicles working in public transport.

POSROI provides information of current position, destination, line, journey and previous and next stop for vehicles working in public transport. The information is based on position and vehicle progress reporting in real-time from source systems to PubTrans.

POSROI does not provide all real time data neither does it provide the long term planned operation. That data is available through other PubTrans interfaces, primarily ROI and DOI.

POSROI works according to the request-response pattern of interaction which means that PubTrans provides current information to the client systems when requested. The client system receives the real time data according to a specific response encoding. In this document we have defined responses for JSON-encoded data, but other encodings such as XML are also possible.

POSROI is designed to allow caching, which makes it a good candidate for large scale public use.

1.2 Related Standards

POSROI is modelled based on TRANSMODEL, the European reference data model for public transportation. The design of POSROI has to some extent also been inspired by the CEN standardisation in progress, mainly the work done in SIRI.

2 Process Overview

2.1 Process Phases

In short, to get current position information from PubTrans the following steps are required:

1. The client uses the HTTP GET method to make a request to the POSROI service.
2. The POSROI service responds with data containing information for the requested selection according to a certain data encoding.
3. The client extracts the data that is relevant for its purposes from the response.
4. The client waits a period of time before making a new request.
5. The above steps are repeated as long as the client needs information.

2.2 Process in Client

2.2.1 Acquiring Background Information

If the client system does not already have information about names and location of the Stop Areas this can be requested from the POSROI service. It is assumed that the client will cache this information for a period of time before asking again as this information is relatively static.

The client sends a HTTP GET request for *StopAreas.* In response data describing the Stop Areas is returned.

2.2.2 Acquiring Positions for Journeys

The client sends a HTTP GET request for *Journeys* matching a certain *selection.* In response data describing the positions for current Journeys covered by the selection is returned. A *selection* could for instance represent a set of public transport lines.

2.2.3 Acquiring Positions and Additional Information for Journeys

The client sends a HTTP GET request for *Extended Journeys* matching a certain *selection.* In response data describing the positions and some other information for current Journeys covered by the selection is returned.

2.2.4 Optimization

If the client is interested in only a few vehicles, but wants frequent position updates for those vehicles it is possible to use a special optimization technique mixing requests for the more compact *Journeys* with requests for *Extended Journeys.*

Journeys and *Extended Journeys* share some attributes. Items in *Extended Journeys* have all the attributes of items in *Journeys* but also some additional attributes which are not included in *Journeys.*

A special checksum constructed from the attributes that are only included in *Extended Journeys* is included among both *Journeys* and *Extended Journeys* items.

This means that the following optimized process is possible:

1) The client request the *Extended Journeys*, and caches the checksum value for each item that is of interest.
2) The client request the *Journeys*, and checks the returned checksum value against the cached checksum value for each item that is of interest.
3) If the checksum differs for any of the items of interest in the answer, then the client requests the *Extended Journeys* again. If the checksum matches, it can be assumed that the extra attributes in *Extended Journey* remains the same.
4) After waiting for a short time the client again requests *Journeys,* compare relevant checksum values and so on.

3 Requesting Data

Requests are made using HTTP GET. Below are examples describing the general structure of the URI to use in different situations.

3.1 Stop Areas

It is possible to request Stop Areas relevant for a certain subset of the public transport by providing a selection parameter in the URI.

The URI should have the following structure: *http://<domain:port>/POSROI/StopAreas/<selection-criteria>*

The following is an example request for Stop Areas relevant for lines included in the "ST"-selection:

http://123.123.123.123:8090/POSROI/StopAreas/ST

3.2 Journeys

It is possible to request journey information for different subsets of the public transport by providing different selection parameters in the URI.

The URI should have the following structure: *http://<domain:port>/POSROI/Journeys/<selection-criteria>*

The following is an example request for current position information for journeys on lines included in the "ST"-selection:

http://123.123.123.123:8090/POSROI/Journeys/ST

3.3 Extended Journeys

It is possible to request extended journey information for different subsets of the public transport by providing different selection parameters in the URI.

The URI should have the following structure:

http://<domain:port>/POSROI/ExtendedJourneys/<selection-criteria>

The following is an example request for current position and additional information for journeys on lines included in the "ST"-selection:

http://123.123.123.123:8090/POSROI/ExtendedJourneys/ST

4 Response Data

The response data is returned using a certain data encoding. See Appendices for details of how the response data is managed for specific data encodings.

Response data comes in the form of sets of entities. Below follows a description of the different sets of entities possible to request with POSROI.

4.1 Stop Areas

Stop Areas contain a set of Stop Area entities.

4.1.1 Stop Area Attributes

Name	Optional?	Description
StopID	No	A unique identity for this StopArea constructed by selecting the significant part of the StopAreaGid. This means that the value equals the StopArea.Number + TransportAuthority.Number * 1000000. Numeric. Max 9 characters.
StopAreaNumber	No	Stop Area Number. Numeric. Max 6 characters.
StopAreaName	No	Name of Stop Area. Max 50 characters.
StopAreaShortName	Yes	Short version of Stop Area Name. Max 16 characters.
StopAreaLatitude	Yes	Latitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.
StopAreaLongitude	Yes	Longitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.

4.2 Journeys

Journeys contain a set of journey entities.

4.2.1 Journey Attributes

Name	Optional?	Description
LineID	No	A unique identity for this Line constructed by selecting the significant part of the LineGid. This means that the value equals the Line.Number + TransportAuthority.Number * 10000. Numeric. Max 7 characters.
JourneyNumber	No	Journey Number. Numeric. Max 5 characters.
Checksum	No	Checksum of values for attributes not exposed in this item but exposed in Extended Journeys. If the checksum value has changed för this journey since last request, then one or more attribute values exclusively available in Extended Journey have changed. Numeric. Max 4 characters.

Name	Optional?	Description
PositionLatitude	No	Latitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.
PositionLongitude	No	Longitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.
PositionTime	No	The time when the position was recorded on the format "HH:MM:SS". Max 8 characters.
SpeedKmPerHour	Yes	Reported speed in km/h. Numeric. Max 3 characters.
Heading360Degrees	Yes	Reported heading. Positive value 0-359. 0 means moving northward. 90 means moving eastward. Numeric. Max 3 characters.
PositionQuality	No	Value describing the quality of the provided data. The value is constructed by combining a value describing the source of the physical position with a value describing if the progress reporting is believed to be reliable or not. The following values are possible for the first part: • GPS = Current GPS position • XPS = No current GPS position, but vehicle at stop according to real time progress reporting • XP1 = Extrapolated position type 1 • XP2 = Extrapolated position type 2 The following values are possible for the second part: • R = Real time progress reporting • S = Simulated progress reporting Max 6 characters.

4.3 Extended Journeys

Extended Journeys contain a set of extended-journey entities.

4.3.1 Extended Journey Attributes

Name	Optional?	Description
LineID	No	A unique identity for this Line constructed by selecting the significant part of the LineGid. This means that the value equals the Line.Number + TransportAuthority.Number * 10000. Numeric. Max 7 characters.
JourneyNumber	No	Journey Number. Numeric. Max 5 characters.

Name	Optional?	Description
JourneyState	No	Journey state value (2 = Expected, 3 = Assigned, 4 = Cancelled, 5 = Signed On, 6 = At origin, 8 = Normal progress, 12 = Aborted). Numeric. Max 2 characters.
LineDesignatation	No	Line designation. Usually the line number, but could be alphanumeric. Max 8 characters.
PrimaryDestinationName	No	Main destination. Max 50 characters.
SecondaryDestinationType	Yes	Code for type of secondary destination if there is a secondary destination. The code could be V for Via. Max 1 character.
SecondaryDestinationName	Yes	Typically used for via destination. Max 50 characters.
StartStopID	No	A reference to the stop where this journey starts. Numeric. Max 9 characters.
StartStopTimetabled-EarliestDepartureTime	No	The planned time on the format “HH:MM” for departing from the first stop on this journey. Max 5 characters.
PreviousStopID	No	Normally a reference to the stop this journey last departed from. If the journey has not yet started it will be the first stop of the journey. Numeric. Max 9 characters.
PreviousStopQuay	Yes	Stop Point Designation showing the stop position within the Stop Area for the previous stop. Could be a track number. Max 4 characters.
PreviousStopTimetabled-EarliestDepartureTime	No	The planned time on the format “HH:MM” for departing from the previous stop on this journey. Max 5 characters.
DelaySeconds	Yes	Delay in seconds. Normally deducted from the difference between actual and planned departure time from previous stop. Negative value indicates departure before planned time. Numeric. Max 6 characters.
NextStopID	No	A reference to the next stop after the previous stop. This is normally either the stop the vehicle is approaching or the stop it is standing at. If the journey has not yet started it will be the second stop of the journey. Numeric. Max 9 characters.
NextStopQuay	Yes	Stop Point Designation showing the stop position within the Stop Area for the next stop. Could be a track number. Max 4 characters.
NextStopTimetabled-LatestArrivalTime	Yes	The planned latest arrival time on the format “HH:MM” for the next stop on this journey. Max 5 characters.
NextStopTimetabled-EarliestDepartureTime	Yes	The planned time on the format “HH:MM” for departing from the next stop on this journey. Max 5 characters.

Name	Optional?	Description
NextStopDepartureState	No	Departure state value (2 = Expected, 6 = At stop, 9 = Departed, 3 = Cancelled, 10 = Passed, 11 = Missed, 12 = Replaced). Observe that if the journey is cancelled according to the journey state then the all departures on the journey are cancelled regardless of the value of this state. Numeric. Max 2 characters.
Checksum	No	Checksum of attribute values listed above and DeviationMessage below. Numeric. Max 4 characters.
PositionLatitude	No	Latitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.
PositionLongitude	No	Longitude according to WGS84. Decimal degrees with five decimals. Numeric. Max 8 characters.
PositionTime	No	The time when the position was recorded on the format "HH:MM:SS". Max 8 characters.
SpeedKmPerHour	Yes	Reported speed in km/h. Numeric. Max 3 characters.
Heading360Degrees	Yes	Reported heading. Positive value 0-359. 0 means moving northward. 90 means moving eastward. Numeric. Max 3 characters.
PositionQuality	No	Value describing the quality of the provided data. The value is constructed by combining a value describing the source of the physical position with a value describing if the progress reporting is believed to be reliable or not. The following values are possible for the first part: • GPS = Current GPS position • XPS = No current GPS position, but vehicle at stop according to real time progress reporting • XP1 = Extrapolated position type 1 • XP2 = Extrapolated position type 2 The following values are possible for the second part: • R = Real time progress reporting • S = Simulated progress reporting Max 6 characters.
DeviationMessage	Yes	Intended for deviation message that applies for all of the journey or specifically at next stop and is suited for this context. May or may not be provided. Always NULL if not provided.

Appendices

1 JSON Encoded Response Data

When using POSROI with JSON encoded response data some meta data items are added in the response.

The first meta data item describes the extent of the response by providing the selection criteria applied to select the subset of data for this response.

The second meta data item describes the time when the data for the response was extracted and assembled.

Thirdly there is a meta data item in the form of an array containing the names of the data attributes.

The actual data is provided as a multidimensional string array for reasons of making the data more compact.

1.1.1 Example JSON Response

```
{
 "selection":"ST",
 "timeStamp":"2012-04-19 08:46:36",
 "stopAreas":{
  "keys":["StopID", "StopAreaNumber", "StopAreaName", "StopAreaShortName", "StopAreaLatitude", "StopAreaLongitude"],
  "data":[
            ["12007022", "7022", "Älmhult station", "Älmhult station", "56.54521", "14.13548"],
            ["12007051", "7051", "Alvesta", "Alvesta", "56.90153", "14.56197"],
            ["12095006", "95006", "Göteborg C", "Göteborg C", "57.70914", "11.97368"]
  ]
 }
}
```

1.2 Stop Areas

Actual data is provided as a multidimensional string array where the outer array is an array of Stop Areas and the inner array is an array of string values containing the attributes for a Stop Area.

1.2.1 JSON Schema

```
 {
   "type":"object",
   "properties":{
     "selection":{
       "description":"Extent of the response.",
       "type":"string",
       "required":true
     },
     "timeStamp":{
       "description":"Time of response in the format YYYY-MM-DD HH:MM:SS.",
       "required":true,
       "type":"string",
       "required":true
     },
     "stopAreas":{
       "type":"object",
       "properties":{
```

```
          "keys":{
            "description":"Array of Stop Area attribute names.",
            "type":"array",
            "items":{
              "type":"string"
            }
          },
          "data":{
            "description":"Array of Stop Areas.",
            "type":"array",
            "items":{
              "description":"Array of Stop Area attribute values.",
              "type":"array",
              "items":{
                "type":"string"
              }
            }
          }
        }
      }
    }
```

1.3 Journeys

Actual data is provided as a multidimensional string array where the outer array is an array of journeys and the inner array is an array of string values containing the attributes for a journey.

1.3.1 JSON Schema

```
{
  "type":"object",
  "properties":{
    "selection":{
      "description":" Code representing a predefined set of public transport lines covered by the response.",
      "type":"string",
      "required":true
    },
    "timeStamp":{
      "description":"Time of response in the format YYYY-MM-DD HH:MM:SS.",
      "required":true,
      "type":"string",
      "required":true
    },
    "journeys":{
      "type":"object",
      "properties":{
        "keys":{
          "description":"Array of journey attribute names.",
          "type":"array",
          "items":{
            "type":"string"
          }
```

```
        },
        "data":{
          "description":"Array of journeys.",
          "type":"array",
          "items":{
            "description":"Array of journey attribute values.",
            "type":"array",
            "items":{
              "type":"string"
            }
          }
        }
      }
    }
  }
```

1.4 Extended Journeys

Actual data is provided as a multidimensional string array where the outer array is an array of extended journeys and the inner array is an array of string values containing the attributes for an extended journey.

1.4.1 JSON Schema

```
{
  "type":"object",
  "properties":{
    "selection":{
      "description":" Code representing a predefined set of public transport lines covered by the response.",
      "type":"string",
      "required":true
    },
    "timeStamp":{
      "description":"Time of response in the format YYYY-MM-DD HH:MM:SS.",
      "required":true,
      "type":"string",
      "required":true
    },
    "extendedJourneys":{
      "type":"object",
      "properties":{
        "keys":{
          "description":"Array of extended journey attribute names.",
          "type":"array",
          "items":{
            "type":"string"
          }
        },
        "data":{
          "description":"Array of extended journeys.",
          "type":"array",
          "items":{
            "description":"Array of extended journey attribute values.",
```

```
            "type":"array",
            "items":{
              "type":"string"
            }
          }
        }
      }
    }
  }
}
```