

Schedule Data Profile Guidance Series

Extended SDP Guidance: Quick Start Version

SDP Version 2.0 Description – Quick Start Version

March 21, 2011

The Schedule Data Profile (SDP) enables transit agencies to exchange schedule and related data for use by many different downstream systems. The SDP is a specification that describes the meanings and syntax related to schedule and transit asset data that meet the requirements of the downstream systems. The SDP is defined as a robust data model and implemented in an XML Schema. This document is one of several Guidance documents that describe the data concepts, requirements, and the XML Schema implementation for the schedule data model. In particular, this document explains the mandatory elements and nested elements in the SDP XML Schema and describes the organization of the schema document.

An exchange format enables data derived from different formats and representing similar concepts to be described in a normative way, adhering to the same meanings and following specified business rules. The SDP XML Schema describes a normative format for documenting and checking data to ensure that downstream systems may *automatically* use the data.

The SDP XML Schema is currently being used to document and submit data to the regional Transit Trip Planner in 511NY. Full documentation for the SDP requirements, schema and other formats may be found in <http://ngtsip.pbworks.com>.

Table of Contents

TERMS AND ACRONYMS

PURPOSE OF THIS DOCUMENT

SDP XML SCHEMA AND BRANCH LEVEL ELEMENTS

- AGENCY REGISTRATION
- SERVICE
- TRANSIT NETWORK
- TRANSIT GAZETTEER
- TRANSIT FACILITIES
- MANDATORY SDP XML SCHEMA ELEMENTS

HIGH-LEVEL ELEMENT DESCRIPTIONS

- SDP DOCUMENT AND SCHEDULE VERSION
- AGENCY
- SCHEDULE REVISION
- ROUTE
- ROUTE DIRECTION
- DAY TYPE
- TRIP
- TRIP TIMES

PATTERN
TRANSITPOINTEVENT
LOCATION

NESTED ELEMENT DESCRIPTIONS

RELATIVE LOCATION
ADDRESSSTRUCTURE
UNIQUE IDENTIFIERS
DATA TYPE DESCRIPTIONS

SCHEDULE CALENDAR XML SCHEMA

DAY TYPE CODES

SAMPLE SDP DOCUMENT

Terms and Acronyms

This section describes the terms and acronyms that are used in this document.

Term	Definition
Branch	A complex type element that classifies a set of nested elements in the XML Schema.
Element	A data concept that is included in an XML Schema. An element description may be defined as a <i>simple type</i> element which describes a field or record in the XML schema, or a <i>complex type</i> which includes multiple simple or complex type elements.
Nested Element	A complex type element that is embedded in another complex type element.
SDP	Schedule Data Profile
Valid Submission	Checking of a registered document to ensure that conforms to the requirements of the SDP XML Schema and other business rules outlined in the SDP Functional Requirements document. XML validation checks for correct data type conformance, data organization and mandatory elements as defined by the SDP XML Schema. The SDP business rules checks for identifier uniqueness, matching identifier references, dates within range, and other key rules.

Additional terms and acronyms may be found in the *SDP Guidance Document Glossary and Acronyms*. [<http://ngtsip.pbworks.com/w/page/12503389/Library>]

Purpose of this Document

This Quick Start Guidance document provides guidance on implementing the *mandatory* elements of the SDP. For more detailed guidance that covers all the elements and nested elements of the SDP, please refer to complete Guidance document.

[<http://ngtsip.pbworks.com/w/page/12503389/Library>]

SDP XML Schema and Branch Level Elements

This section describes the SDP XML Schema. Both Mandatory and optional high level elements are described in this section. Figure 1: SDP Schema Organization: High Level Element Hierarchy shows the SDP Schema organization of all the elements included in each branch level. [Note: Figure 2: SDP Schema of Mandatory Elements Only shows the high level elements that are mandatory. This Quick Start Version only includes descriptions and guidance related to mandatory elements depicted in Figure 2.]

Agency Registration

The Agency Registration branch describes general information related to an agency that registers the actual data that is stored in the SDP document. Included in this branch is information on the agency, schedule version and revisions, and the organization that submitted the data or is referred to by the data set content. In addition, the Agency Registration branch contains the routes, special route groupings and special service day type descriptions that are used by other elements in the Service, Network or Gazetteer branches of the SDP document.

Service

The Service branch describes information on the service provided by the agency or organization unit that generated the SDP document. The scheduling elements include information on trips and trip times, scheduling notes, bus assignment schedules, and coordinated trips (EventConnection).

Transit Network

The Transit Network branch describes the route path traversed by transit service. The network is composed of transit paths called patterns. Events occur on each pattern, and patterns may be composed of segments, often called route segments or timepoint intervals (TPI).

Transit Gazetteer

The Transit Gazetteer defines places and their location. A valuable addition is the incorporation of a "Location Table" which aggregates spatial references. The element simplifies data maintenance and enables the linking of places to equivalent locations (which may be described using a different location referencing system). For example, Penn Station Long Island Rail Road (LIRR) is the equivalent location as Penn Station Amtrak, New Jersey Transit, New York City Transit Lines A, C, E. Other transit features such as timepoints and transfer locations are also included in this category.

Transit Facilities

The Transit Facilities branch, although part of a typically Gazetteer, includes elements that provide a robust model of the complex transit facilities in the New York region. The branch contains high level elements such as Transit Facility and Parking Facility together with its Plant Components. For example, parking lot, boarding area (platform), track, entrance, and elevator are plant components of a train station. Transit Facilities includes the asset inventories related to transit facilities including transit stops, amenities, portals, passenger access elements and tracks (associated with a transit facility). Due to the complex nature of key New York City transit facilities like Port Authority Bus Terminal, Jamaica Station, Grand Central Terminal, and Pennsylvania Station in New York, the model enables a facility to be part of another facility.

Transit Vehicles

The Transit Vehicles branch, included in the extended SDP version 2.0, includes descriptions for transit revenue vehicle bus and rail vehicles. A transit operator defines a Fleet Type (that is often assigned to a block) as a category of conveyance assigned to public transportation revenue service that carries public transportation patrons; the type may be associated with any fleet type that is used for passenger service including bus, train, light rail, heavy rail, or ferry. The PTV (public transit vehicle), a single “piece” transit vehicle, includes a conveyance’s characteristics that is assigned to specific work. The Rail Service Unit (RSU) is a train (including locomotive and rail cars) that is assigned to specific work. A consist are the rail cars that make up a train, and finally, the rail car describes the characteristics contained in a specific car including car capacity.

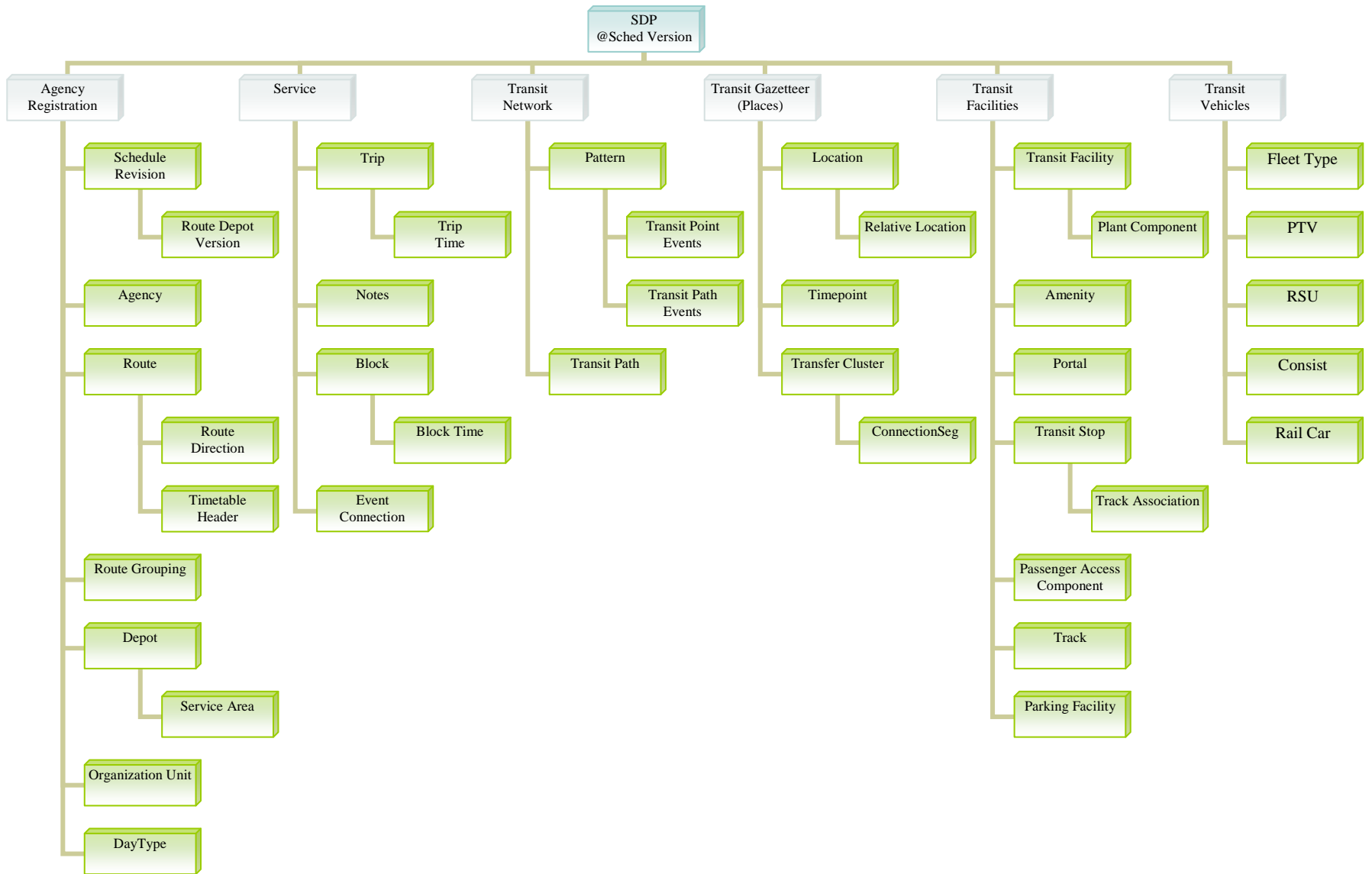


Figure 1: SDP Schema Organization: High Level Element Hierarchy

Mandatory SDP XML Schema Elements

The SDP XML Schema is designed to meet many existing and future needs. As a result, the Schema contains both mandatory and optional SDP XML schema elements. The SDP must meet the schedule information requirements of several downstream applications such as Trip Planning, Timetable, and Ad Hoc Schedule Planning¹, as well as support the existing upstream requirements of data providers. The Extended SDP 2.0 must also support Fare Calculator and Regional Planning functions that are defined in the SDP extension requirements documents². To support the diverse requirements of the region, there are many optional elements in the SDP Schema.

The high level mandatory elements that are required by every validated SDP document are included in Figure 2.

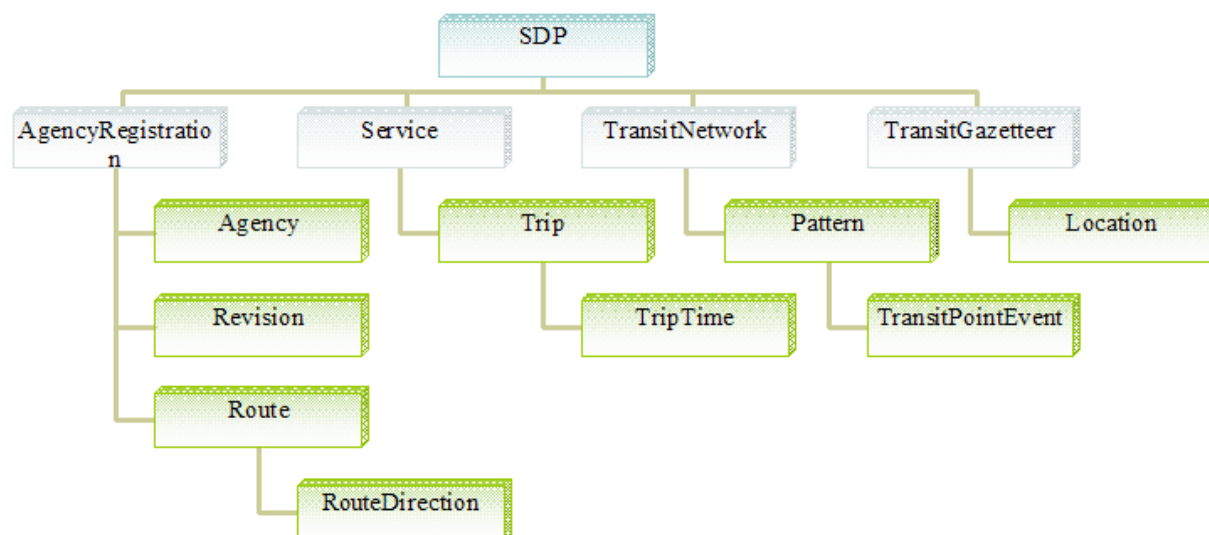


Figure 2: SDP Schema of Mandatory Elements Only

Only the *mandatory* high level elements (and their nested elements) are discussed in the section on *High-Level and Nested Element Descriptions*.

High-Level Element Descriptions

High level elements are those that are contained in Figure 2 SDP Schema of Mandatory Elements.

¹The high level requirements for these processes may be found in the Appendices of the TSDEA Concept of Operations document.

² Extension needs are described in Concept of Operations and Requirements document found at <http://ngtsip.pbworks.com/w/page/34418345/Final-TSIP-Publications>

SDP Document and Schedule Version

Definition: The time period, described by the start date and time and optionally end date and time, when an agency service provision is valid. Due to the nature of transit schedules, different routes, depots and organizational units may implement various versions that operate during the same time period.

Example: Long Island Bus typically issues a new Schedule Version in March, June, September and January.

The Schedule Version is included as SDP Document header attribute group. Even if values are not available, all elements of the attribute group are mandatory.

Table 1: Schedule Version Attribute Group Elements

Attribute	Description
scheduleVersionID	<p>Mandatory (embedded element scheduleVersionID_id) UNIQUE.</p> <p>Use the General Order (often used by a rail operator), identifier used in the pick, or designation for the schedule.</p> <p>An example of a schedule version is <code>scheduleVersionID="211"</code>.</p> <p>This scheduleVersionID implies 2nd Quarter in the year 2011.</p>
scheduleVersionDescription	<p>Mandatory (string).</p> <p>A string that is used to describe the schedule version.</p> <p>In the example 211, the scheduleVersionDescription might read: <code>scheduleVersionDescription="2nd Quarter 2011"</code> or <code>scheduleVersionDescription="3-20-2011 to 6-30-2011"</code></p>
pickNo	<p>Mandatory (string).</p> <p>This field is sometimes the same as the scheduleVersionID. Although the attribute tag must be inserted into the SDP document, you may leave this element blank if no pickNo is needed or if it repeats the scheduleVersionID.</p> <p>Example: <code>pickNo=""</code> (for a null value)</p>
activationDate	<p>Mandatory (date). Format [yyyy-mm-dd]</p> <p>The date the schedule version begins operating.</p> <p>Example: <code>activationDate="2011-03-20"</code></p>
deactivationDate	<p>Mandatory (date). Format [yyyy-mm-dd]</p> <p>The date the schedule version ends operating. Use a default value of 9999-12-31 if the end date is unknown. This rule applies to all fields</p>

Attribute	Description
	that include a deactivationDate. Example: deactivationDate= “2011-06-30”
placementDate	Mandatory (date). Format [yyyy-mm-dd] Date on which schedule was generated or developed (use date compiled into SDP format if no date is available). Example: placementDate= “2011-03-14”

Agency

Definition: A transit agency is an organization that provides transportation services by bus, rail, or other conveyance to the general public or special services on a regular, continuous basis.

Table 2: Agency Element Descriptions

Element	Description
agencyID	Mandatory (agencyID_id) UNIQUE. This designator is assigned by SDP/TSDEA registration. (use agencyAcronym if not available.)
agencyAcronym	Mandatory (string). Agency acronym that is known by the public, for example MNR or MTA MNR for the Metro-North Railroad.
agencyName	Optional (string). Full name of agency, for example “Metropolitan Transportation Authority New York City Transit” or “MTA New York City Transit”.
webAddress	Optional (string). The universal resource link or home page web address of the agency.
headquarterAddress	Optional. (nested element AddressStructure). See address structure guidance in AddressStructure
headquarterTelephone	Optional (TELEPHONE type) Format: this field has a maximum length of 10 numbers [aaa-ttt-tttt] (where aaa refers to the area code followed by seven digits). The general information number or a general customer information telephone number.
serviceAreaDescription	Optional (string). A description of service area. The description is used to present to the public. As such the description should avoid abbreviations and excessive capitalization. Example XML Fragment: <serviceAreaDescription>

Element	Description
	<p>“The LIRR system is comprised of over 700 miles of track on 11 different branches, stretching from Montauk -- on the eastern tip of Long Island -- to the refurbished Penn Station in the heart of Manhattan, approximately 120 miles away. Along the way, the LIRR serves 124 stations in Nassau, Suffolk, Queens, Brooklyn and Manhattan.”</p> <p></serviceAreaDescription></p>
hrsOfOperations	<p>Optional (string).</p> <p>A basic description of the hours of operation, for example: “5:30 am to 10 pm” or “24 hours”.</p> <p>Example XML Fragment: <hrsOfOperations>5:30 am to 10 pm</hrsOfOperations></p>

An example XML fragment of Agency may appear as follows:

```

<Agency>
  <agencyID>100</agencyID>
  <agencyAcronym>NYCT</agencyAcronym>
  <agencyName>New York City Transit Bus</agencyName>
  <webAddress>http://www.mta.nyc.ny.us/nyct/bus/index.html</webAddress>
  <headquarterAddress>
    <addressNumber>2</addressNumber>
    <streetName>Broadway</streetName>
    <completeName>2 Broadway</completeName>
    <postalCommunity>New York</postalCommunity>
    <postalState>NY</postalState>
    <postalCode>10004</postalCode>
  </headquarterAddress>
</Agency>

```

Schedule Revision

Definition: Records the state and manages the changes to the designated schedule version. A schedule revision is a submission of one or more parts of a schedule version.

The schedule version is the umbrella that groups all the service and changes to the service to a schedule period or operator selection period. The revision element provides a means of tracking updates of all or parts of a schedule on a permanent or temporary basis.

Table 3: Schedule Revision Element Descriptions

Element	Description
revisionNumber	<p>Mandatory (revisionNumber_id) UNIQUE.</p> <p>The revisionNumber is a unique, potentially sequential, number within a scheduleVersionID. The original schedule may be set to "0" and incremented whenever a change is made.</p>

activationDate	<p>Mandatory (date). Format [yyyy-mm-dd]</p> <p>The date of service activation. An original schedule revision will be assigned the same date as the Schedule Version activationDate. When the revision is either permanent or temporary (for example a long term detour routing), then start date of revised schedule is used.</p> <p>Example: for a permanent or temporary revision, the activation date may be 2011-03-20.</p>
deactivationDate	<p>Mandatory (date). Format [yyyy-mm-dd]</p> <p>The date of service deactivation. An original schedule will be assigned the same date as the Schedule Version deactivationDate. When the revision is permanent the deactivationDate will be assigned the same as the Schedule Version deactivationDate. When the revision is temporary, then the deactivationDate is assigned the date the service changes ends.</p> <p>Example: for a temporary revision, such as a detour, the service deactivationDate may be 2011-06-30.</p>
placementDate	<p>Mandatory (date). Format [yyyy-mm-dd]</p> <p>Date schedule revision was generated or placed. When the actual date is not known then use the date for which the SDP document was generated.</p>
scheduleVersionType	<p>Mandatory (scheduleVersionType_cd).</p> <p>A classification for the type of schedule that is included in this SDP document. Codes include [original, rev-permanent, rev-temporary, suspended]. The type value should be assigned as follows:</p> <ul style="list-style-type: none"> • “original” is assigned to an original schedule. • “rev-permanent” is used when the revision replaces the last revision until the end of the schedule version operating period. • “rev-temporary” is a temporary replacement to the last revision. When the deactivation date is reached, the schedule version reverts to the last permanent revision. • “suspended” implies that there is no replacement for the schedule. It is no longer in operation (until a new schedule is registered). <p>See Complete Guidance Document, Chapter 4 for discussion on schedule revision precedence when using these values (see http://ngtsip.pbworks.com/library)</p>
history	<p>Mandatory (string).</p> <p>History is a list of the changes to the schedule since the last revision. Use a standard convention for generating the history, for example “1- original; 2- updated routes 2, 10, 12;” etc.</p>

An example XML fragment of ScheduleRevision may appear as follows:

<ScheduleRevision>

```

<revisionNumber>0</revisionNumber>
<activationDate>2011-03-20</activationDate>
<deactivationDate>2011-06-30</deactivationDate>
<placementDate>2011-03-14</placementDate>
<scheduleVersionType>original</scheduleVersionType>
<history>1-original</history>
</ScheduleRevision>

```

Route

Definition: A collection of patterns and associated trips in revenue service with a common identifier or name. Railroad operators may use the term line or branch by which to group service.

Table 4: Route Element Descriptions

Element	Description
routeID	<p>Mandatory (routeID_id) UNIQUE.</p> <p>A unique designation for route. This designator should be used for routeID throughout the rest of the document. It will be used to aggregate the service related components (i.e., trips and physical paths/patterns) with the route.</p> <p>Example: M01</p>
routeName	<p>Optional (string).</p> <p>A route may refer to an internal name or designator by one or more stakeholders or applications within an organization. The value of this field refers to an alternate name used to reference a route.</p> <p>Example: "M 0001 FIFTH-MADISON-PARK AV S"</p>
mode	<p>Mandatory (mode_cd).</p> <p>A transit service classification characterized by specific right of way, technological and operational features.</p> <p>Mode codes are listed in mode_cd; e.g., MB for bus, CR for commuter rail, HR for subway (heavy rail), FR for ferry, etc.</p>
publicRouteName	<p>Mandatory (string).</p> <p>A name recognized for the route by the public for the route.</p> <p>Example: "M1 Fifth-Madison Park Av S"</p>
publicRouteNumber	<p>Mandatory (string).</p> <p>A route number recognized for the route by the public.</p> <p>Example: M1</p>
routeDescription	<p>Optional (string).</p> <p>A description for the route. This field is not needed if it is redundant with publicRouteName. The field may be used to describe the physical path of the route, or if it contains abbreviations for other internal or machine-to-machine data transfer.</p>

adaCompliance	<p>Optional (adaCompliance_cd).</p> <p>The adaCompliance describes the nature of the vehicle types and stop facilities that serve this route. The code values include:</p> <ul style="list-style-type: none"> notCompliant fullyCompliant mobilityChallengedAccess visuallyImpairedAccess hearingImpairedAccess mobility-VisuallyImpairedAccess visually-HearingImpairedAccess mobility-HearingImpairedAccess <p>The designation implies that the route inclusive of vehicles and stops provide services that accommodate persons with specific impairments.</p>
routeBeginDate	<p>Optional (date). Format [yyyy-mm-dd]</p> <p>The date on which the route is placed into service with respect to the schedule version or revision.</p>
routeEndDate	<p>Optional (date). Format [yyyy-mm-dd]</p> <p>The last day of service on this route with respect to the schedule version or revision. If no end date is present in a native route record, then use a default value such as the schedule deactivationDate or the default end date 9999-12-31.</p>
routeDirectionList	<p>Mandatory (RouteDirection).</p> <p>The routeDirectionList is a nested element. In most cases there are two nested elements, first and second directions. The nested elements are used to customize public route direction names for each route. Complete Route Direction record for the number of route directions that are listed by the patterns in this route. (See next section for more on Route Direction).</p>

An example XML fragment of Route may appear as follows:

```

<Route>
  <routeID>M1</routeID>
  <mode>MB</mode>
  <publicRouteName>M1 FIFTH-MADISON-PARK AV S</publicRouteName>
  <publicRouteNumber>M1</publicRouteNumber>
  <routeDescription>FIFTH-MADISON-PARK AV S</routeDescription>
  <routeDirectionList>
    <routeDirection>first</routeDirection>
    <routeDirectionDescription>N</routeDirectionDescription>
    <publicRouteDirection>North</publicRouteDirection>
  </routeDirectionList>
  <routeDirectionList>
    <routeDirection>second</routeDirection>
    <routeDirectionDescription>S</routeDirectionDescription>
    <publicRouteDirection>South</publicRouteDirection>
  </routeDirectionList>

```

</Route>

Route Direction

Definition: The direction of travel of a transit conveyance along the physical path. The element allows for public facing information.

Note: The RouteDirection element is nested under Route, as such it inherits the routeID from the Route element.

Table 5: RouteDirection Element Descriptions

Element	Description
routeDirection	Mandatory (routeDirection_cd). Enumerated index used to match routeDirection in Pattern element. [first, second]
routeDirectionDescription	Optional (string). The name or code used in the native data. Content is determined by the user, and may be used to support downstream applications.
publicRouteDirection	Mandatory (string). The route direction name known and viewed by the public Note: The public route direction name may be the destination or origin of the route, for example, “New York City” or “Port Washington” in the case of Long Island Rail Road.

An example XML fragment of RouteDirection may appear as follows:

```
<routeDirectionList>  
  <routeDirection>second</routeDirection>  
  <routeDirectionDescription>S</routeDirectionDescription>  
  <publicRouteDirection>South</publicRouteDirection>  
</routeDirectionList>
```

Another example XML fragment of RouteDirection that uses the destination location as the direction is as follows:

```
<routeDirectionList>  
  <routeDirection>first</routeDirection>  
  <routeDirectionDescription>New York City (Penn Station) </routeDirectionDescription>  
  <publicRouteDirection>Penn Station</publicRouteDirection>  
</routeDirectionList>  
<routeDirectionList>  
  <routeDirection>second</routeDirection>  
  <routeDirectionDescription>Port Washinton via Shea Stadium</routeDirectionDescription>  
  <publicRouteDirection>Port Wash</publicRouteDirection>  
</routeDirectionList>
```

Day Type

Definition: A classification for the service provided on a type of day or day of the week, or a “designation for a type of day characterized by one or more properties that affect public transport operation.” [NTCIP 1404: 2000]

The DayType element is a lookup table for locally defined day type definitions. Enumerated values defined by the dayType code include: Monday, weekday, Saturday, holiday, Thanksgiving, Wed before Thanksgiving, New Years Eve, or New Years Day.

Table 6: DayType Element Description

Element	Description
dayType	<p>Mandatory (dayType_cd).</p> <p>An index used to identify the service or day type (dayType) element in a Trip element or Schedule Calendar document.</p> <p>The data type must be a numeric value if not contained in the list of enumerated types (see Day Type Codes below). For example, special service may be designated by the date of occurrence: <code><dayType>20111002</dayType></code></p>
dayTypeDescription	<p>Mandatory (Description).</p> <p>A name that may be provided to the public to identify the special day (e.g., ticker tape parade). <code><dayTypeDescription>”Ticker Tape Parade”</dayTypeDescription></code></p>
dayTypeValue	<p>Optional (string).</p> <p>A code used by the organization or another specification / standard. For example, “parade”. <code><dayTypeValue>parade</dayTypeValue></code></p>
dateTimeBegin	<p>Optional (datetime)</p> <p>The time and date a day type begins. This element should be provided if the service is scheduled only on a single day or range of dates. <code><dateTimeBegin>2011-10-02</dateTimeBegin></code></p>
dateTimeEnd	<p>Optional (datetime)</p> <p>The time and date a day type is completed. This element should be provided if the service is scheduled for a range of dates. <code><dateTimeEnd>2011-10-02</dateTimeEnd></code></p>
agencyID	<p>Optional (agencyID_id)</p> <p>The agency that defines the service day. This element is included when more than one agency is included in the SDP document and there are several locally defined day types that are included in the file.</p>

An example XML fragment of DayType may appear as follows:

```

<DayType>
  <dayType>mem-day</dayType>
  <dayTypeDescription>Memorial Day</dayTypeDescription>
  <dayTypeValue>memday</dayTypeValue>
  <dateTimeBegin>2011-05-30</dateTimeBegin>
</DayType>

```

Trip

Definition: A one way scheduled movement of a transit vehicle between starting and ending locations. Each trip is an instance of a pattern where service is provided for a route in a given direction. The railroad alias for this term is “train”.

Table 7: Trip Element Description

Element	Description
tripID	<p>Mandatory (tripID_id) UNIQUE.</p> <p>A unique designation for a trip. The value for this element need only be unique within this SDP document. It will be used to relate to elements from other elements.</p> <p>Example: 1024</p>
routeID	<p>Mandatory (routeID_id).</p> <p>The routeID should match the convention of the associated routeID in the Route element.</p>
patternID	<p>Mandatory (patternID_id).</p> <p>The patternID should match the convention of the associated patternID in the Pattern element.</p>
dayType	<p>Mandatory (list of dayType_cd).</p> <p>A dayType is the type of service generally assigned to the type of day. For example, weekday, Saturday, Sunday services. Multiple elements may be included for dayType. For example, if the trip is used for multiple special days: 3rd and 4th Sunday before Christmas, and Veteran's Day, then three dayTypes may be included in the Trip structure. The dayType must cover all the nested elements (noteList and tripTimeList) in the Trip structure.</p> <p>The code includes over 50 general and special day types (including holidays such as Thanksgiving, New Year's Eve Day, New Year's Day, and more). The Guidance document includes special rules for assigning dayTypes in order to mitigate overlap and confusion when generating timetables. These may be found at Day Type Codes.</p>
tripName	<p>Optional (string).</p> <p>A name or number for trip or train that is recognized by the public.</p> <p>For example, The "Overnighter".</p>

tripType	<p>Optional (tripType_cd).</p> <p>A classification for a type of trip. Enumerated values include [revenue, pullIn, pullOut, extra, deadhead]. Only revenue trips are needed for regional applications.</p>
serviceType	<p>Optional (serviceType_cd)</p> <p>A classification for a type of service provided by trip.</p>
timeBegin	<p>Mandatory (signed integer).</p> <p>The field timeBegin is the time when a trip begins. (timeBegin should match the first nested tripTimeList (Trip Time structure) element tripTime value.)</p> <p>Units are in seconds past midnight. Negative values imply time before midnight, and positive values may exceed 24 hours.</p> <p>Example: 23340 [6:29:00 AM]; -60 [day before at 11:59:00 PM]</p>
timeEnd	<p>Mandatory (signed integer)</p> <p>The field timeEnd is the time when the trip ends. (timeEnd should match the last nested tripTimeList (TripTime structure) element tripTime value.)</p> <p>See timeBegin for example and unit guidance.</p>
locationBegin	<p>Mandatory (locationID_id).</p> <p>The field locationBegin is the place where a trip begins. (locationBegin should match the first nested tripTimeList (Trip Time structure) element locationID value.) locationID should match a Location element with the same locationID.</p> <p>Example: 100546 is an example of a locationID.</p>
locationEnd	<p>Mandatory (locationID_id).</p> <p>The field locationEnd is the place where a trip ends. (locationEnd should match the last nested tripTimeList (Trip Time structure) element locationID value.) locationID should match a Location element with the same locationID.</p>
noteList	<p>Optional (list of noteID_id).</p> <p>The noteList is a set of nested elements with a single element noteID. The noteID should match a Note structure element with the same identifier.</p>
tripTimeList	<p>Mandatory (ordered list of TripTime).</p> <p>The tripTimeList is a set of nested elements of a sequence of times where the transit vehicle passes (arrives/departs) during its traversal of the trip. Typically the elements of the list are stored in sequential order, first time to last time of the trip.</p> <p>See Trip Times section for details of the tripTimeList structure.</p>

An example XML fragment of Trip may appear as follows:

```

<Trip>
  <tripID>1</tripID>
  <routeID>M1</routeID>
  <patternID>M01_0040_2</patternID>
  <dayType>sat</dayType>

```



```

<dayType>sun</dayType>
<tripType>revenue</tripType>
<timeBegin>5400</timeBegin>
<timeEnd>8340</timeEnd>
<locationBegin>26fd</locationBegin>
<locationEnd>847d</locationEnd>
<tripTimeList>
  <tripTime>5400</tripTime>
  <tripEventType>AlightBoard</tripEventType>
  <timeType>departure</timeType>
  <locationID>26fd</locationID>
</tripTimeList>
  --- More tripTimeList elements here --
<tripTimeList>
  <tripTime>8340</tripTime>
  <tripEventType>AlightBoard</tripEventType>
  <timeType>arrival</timeType>
  <locationID>847d</locationID>
</tripTimeList>
</Trip>

```

Note: this example includes two day types. Although multiple instances of the trip are typically documented, some agencies may decide to include multiple day types in a trip in order to avoid complex day type definitions when trips operate on non-typical schedules (e.g., tues through thurs).

Trip Times

Definition: The time along a trip when a vehicle is scheduled to pass (arrives at/departs from). The trip time may be published or unpublished, coordinated or uncoordinated with trip times of other trips.

Note: The TripTime element is nested under the Trip element. As such it inherits the tripID and routeID from Trip.

Table 8: TripTime Element Descriptions

Element	Description
tripTime	<p>Mandatory (integer).</p> <p>Similar to timeBegin and timeEnd in the Trip element, tripTime is a time when a transit vehicle delivers service to a specific location along a trip.</p> <p>Units are in seconds past midnight. Negative values imply time before midnight, and positive values may exceed 24 hours.</p> <p>Example: 23340 [6:29:00 AM]; -60 [day before at 11:59:00 PM]</p>
timeEventType	<p>Optional (timeEventType).</p> <p>A classification for a trip event to which this element is associated. For example, the event may include a fare set change, headsign change, route change, and other events that may not be covered by a trip on a single route or associated with a route/pattern. Business rules define the coding standards for the Time Event Type. See Guidance Document for additional information on enumeration values.</p>

timeType	<p>Mandatory (timeType_cd).</p> <p>A classification of the trip time as a first or last timing point, arrival or departure timing point.</p> <p>timeType enumeration values include:</p> <ul style="list-style-type: none"> • arrival --Time the transit vehicle arrives at a place • departure -- Time the transit vehicle departs a place • firsttime -- The first timing point along the trip. This point is always a departure point. • lasttime -- The last timing point along the trip. This point is always an arrival point. <p>This field must be included in a list of points associated with a trip.</p> <p>Typically, if the tripTime is the first, then the default field is departing. If the field is the last, then the default is arrival. The default for all other occurrences depends on scheduling policy.</p>
locationID	<p>Mandatory (locationID_id).</p> <p>The locationID identifies the place where the trip time occurs. This field must match a locationID in a Location element.</p>
platformNo	<p>Optional (string).</p> <p>The place where boarding or alighting occurs. A platform is also known as a bus stop, berth or platform. The number may be associated with a stopID from a TransitStop element.</p>
seqNo	<p>Mandatory (unsigned integer -- unsignedInt).</p> <p>The order of this tripTime in the trip sequence. This field is Mandatory since sometimes the same tripTime is used for multiple tripTime elements.</p>
notes	<p>Optional (list of noteID_id).</p> <p>A note identifier (noteID) that corresponds to a note element in the Note library. A note might indicate that the event occurs during certain conditions, for example, school is in session.</p>

An example XML fragment of TripTime may appear as follows:

```

<tripTimeList>
  <tripTime>4900</tripTime>
  <tripEventType>Board</tripEventType>
  <timeType>departure</timeType>
  <locationID>26ab</locationID>
  <seqNo>1</ seqNo >
</tripTimeList>
<tripTimeList>
  <tripTime>5400</tripTime>
  <tripEventType>AlightBoard</tripEventType>
  <timeType>departure</timeType>
  <locationID>26bc</locationID>
  <seqNo>2</ seqNo >

```

```

</tripTimeList>
<tripTimeList>
  <tripTime>5400</tripTime>
  <tripEventType>AlightBoard</tripEventType>
  <timeType>departure</timeType>
  <locationID>26fd</locationID>
  <seqNo>3</ seqNo >
</tripTimeList>

```

(rest of list to follow)

Pattern

Definition: A unique, non-branching, ordered sequence of transit paths, time points, or transit stops to be followed by a transit vehicle in scheduled service for a route in a given direction.

Table 9: Pattern Element Descriptions

Element	Description
patternID	<p>Mandatory (patternID_id) UNIQUE.</p> <p>A pattern identifier (patternID) that designates a unique path traversed by a transit vehicle in revenue service.</p> <p>For example: M01_0010 is an example of a patternID.</p>
patternName	<p>Optional (string).</p> <p>A name given to a Pattern and used by either a downstream application or known to the public. For example, "Route 1 East Express".</p>
description	<p>Optional (string).</p> <p>A description of the pattern. The field may not be needed for most downstream applications.</p>
routeID	<p>Mandatory (routeID_id).</p> <p>The route identifier associated with the pattern. This field must match a routeID in one of the Route elements.</p>
routeDirection	<p>Mandatory (routeDirection_cd).</p> <p>One of the routeDirection elements contained in the Route element. The standard enumerations use 'first' or 'second'. The direction name may be looked up in the nested routeDirection element in the Route element under the publicRouteDirection field.</p>
origin	<p>Mandatory (locationID_id).</p> <p>The originating location of the pattern. This field should match the first entry in the TransitPointEvent.location element. The locationID should exist in a Location element.</p>
destination	<p>Mandatory (locationID_id).</p> <p>The destination location of the pattern. This field should match the last entry in the TransitPointEvent.location element. The locationID should exist in a Location element.</p>

patternType	Optional (patternType_cd). A classification for a pattern. Acceptable values include [revenue, non-revenue]. Current downstream applications do not need non-revenue patterns or trips.
eventList	Mandatory (Choice of ordered list of TransitPointEvent or TransitPathEvent). The elements in this entry are the events that occur along the pattern. A commuter rail system needs only one pattern for each origin-destination pair.

An example XML fragment of Pattern may appear as follows:

```

<Pattern>
  <patternID>M01_0040_1</patternID>
  <routeID>M1</routeID>
  <routeDirection>first</routeDirection>
  <origin>26fd</origin>
  <destination>847d</destination>
  <eventList>
    <transitPointEvent>
      <locationID>26fd</locationID>
      <seqNo>1</seqNo>
      <headsignDesc>HARLEM 147 ST via MADISON</headsignDesc>
    </transitPointEvent>
    <transitPointEvent>
      --- More transitPointEvent elements here --
    </transitPointEvent>
    <transitPointEvent>
      <locationID>847d</locationID>
      <seqNo>64</seqNo>
      <headsignDesc>HARLEM 147 ST via MADISON</headsignDesc>
    </transitPointEvent>
  </eventList>
</Pattern>

```

TransitPointEvent

Definition: A place where transit service is delivered along the transit network.

Note: The TransitPointEvent element is nested under the Pattern element. As such it inherits routeID and patternID from Pattern.

Table 10: TransitPointEvent Element Descriptions

Element	Description
locationID	Mandatory (locationID_id). A reference to the location where the event occurs. The point may be any type of transit feature, physical point, timepoint, transit stop or facility, or other event (e.g., change headsign, trigger location). The locationID should exist in the same SDP document, under the Location element.

seqNo	<p>Mandatory (seqNo_id).</p> <p>The sequence number of the event in the pattern eventList. This number may be derived if the distanceFromOrigin is defined.</p> <p>Note: the data type is assigned an IDENTIFIER type so that it may be validated as unique within each route pattern.</p>
trackNo	<p>Optional (trackNo_id).</p> <p>For trains, the trackNo on which the event occurs.</p>
stopID	<p>Optional (stopID_id).</p> <p>If appropriate, the stop identifier or platform number associated with the event.</p>
distanceFromOrigin	<p>Optional (distanceType).</p> <p>The distance from the Pattern origin. Although this information is not universally collected, it is needed to reduce ambiguity by many downstream applications, and thus is a key field for the SDP.</p> <p>Also includes optional attribute unit which accepts the enumerated values [feet or meters]. Default when no attribute is present is feet.</p> <p>Example XML format:</p> <pre style="text-align: center; color: red;"><distanceFromOrigin units="feet">3.14159</distanceFromOrigin></pre>
ptEventType	<p>Optional (eventType_cd).</p> <p>A classification for an event. More than one event may occur at a single point, for example, exiting a bus stop zone and “checking out” of a Transit Signal Priority zone may be co-located.</p> <p>Enumeration values include:[timing, other]</p> <p>Additional values may be added when needed by downstream applications.</p>
headsignDesc	<p>Optional (string).</p> <p>Although this field may be included multiple times, the field may also contain additional fields to be used throughout the pattern, for example, headsign information such as "via Brooklyn/Flatbush" or "Go Mets".</p>

An example XML fragment of transitPointEvent may appear as follows:

```
<transitPointEvent>
  <locationID>847d</locationID>
  <seqNo>64</seqNo>
  <headsignDesc>HARLEM 147 ST via MADISON</headsignDesc>
</transitPointEvent>
```

Location

Definition: This entity represents a place in the Transit Gazetteer. It contains the location description for a point used to describe or associated with a transit network over which transit service is provided. (Note: the featureType element enables the user to specialize the location

so that different features may be aggregated along a single polyline feature. This also alleviates the need to describe each facility or stop when only a location is required.)

Table 11: Location Element Descriptions

Element	Description
locationID	Mandatory (locationID_id) UNIQUE. An identifier for each feature or physical point needed to associate a schedule component to the transportation network (map database) or location in space.
featureType	Mandatory (featureType_cd). A classification for a place characterized by its location reference. Enumerated values include: [transitFacility, timepoint, transitStop, transferCluster, portal, amenity, passengerAccessComponent, physicalPoint, event, poi, other]
locationDesc	Mandatory (Description – string type). A description of the location. This field may be redundant with intersection or publicLocationDescription.
intersection	Optional (string). The full name of an intersection. For easier parsing, the accepted convention is "on street" @ "at street". Avoid using "&" symbol and other html characters.
geometry	Optional (kml:PolygonType). The geometry or “closed region of space”. The PolygonType is a box with at least five points. The polygon, by definition may contain as many points as needed to define the region. The major requirement is that the points are ordered and the first and last points are the same. Use coordinates of the WGS '84 datum. The specification for the PolygonType comes from KML (see http://code.google.com/apis/kml/schema/kml22beta.xsd for information on the schema). The SDP Guidance document (TBD) includes guidance on its use.
streetDirection	Optional (streetDirection_cd). The general orientation of the street. Enumerated values include [northSouth, eastWest, north, south, east, west]. Note: northSouth and eastWest are typically used to describe the direction of rail alignments where trains may travel in both directions.
relativeLocation	Optional (RelativeLocationStructure). RelativeLocation is a nested element (see detailed description in the section below). The nested element is used when linear referencing methods are used to describe location (e.g., nearside, off street, distance from intersection).

longitude	<p>Mandatory (Longitude).</p> <p>Location of a place on the earth East of the Greenwich meridian. The datum for all latitude and longitude measurements shall be NAD 83. Latitude and longitude will be represented in geographic coordinates, using decimal degrees.</p> <p>Note longitude error varies by latitude, however, a minimum of five decimal places (-73.58391) should be included. The preferred number of decimal places is six (see example below).</p> <p>Example: <code><longitude>-73.583913</longitude></code></p>
latitude	<p>Mandatory (Latitude).</p> <p>Location of a place on the Earth North of the equator. The datum for all latitude and longitude measurements shall be NAD 83. Latitude and longitude will be represented in geographic coordinates, using decimal degrees.</p> <p>Note a minimum of five decimal places (44.12345) should be used in order to provide an accuracy of 1.11 meters or 43 inches. Six decimal places are recommended as documented in the example below.</p> <p>Example: <code><latitude>40.680051</latitude></code></p>
x_coord	<p>Optional (double).</p> <p>State Plane coordinates such as Easting/Northing. For the New York/New Jersey area, the UTM zone is 18.</p>
y_coord	<p>Optional (double).</p> <p>State Plane coordinates such as Easting/Northing. For the New York/New Jersey area, the UTM zone is 18.</p>
county	<p>Optional (string).</p> <p>A standard name for county</p>
city	<p>Optional (string).</p> <p>A standard name for a city.</p>
state	<p>Optional (postalState_cd).</p> <p>The state field should be populated from ANSI/FIPS codes [NY, NJ, CT]</p>
zip	<p>Optional (string).</p> <p>A postal code or zip +4 USPS zip code.</p>
communityName	<p>Optional (string).</p> <p>A local community name in which the location is known.</p>
serviceArea	<p>Optional (string).</p> <p>A description of the service area. This description may define a fare zone, for example.</p>

publicLocationDescription	<p>Mandatory (Description – string type).</p> <p>This is the field that is used to describe the location to the public. The field should not contain abbreviations or codes, and it should use proper punctuation and capitalization.</p>
isGeneralized	<p>Mandatory (boolean).</p> <p>If this location is contained in, is part of, or has a "many" relationship to another location, then isGeneralized is 'false'. For example, a timepoint location may be associated with one or more bus stop locations. In the timepoint/bus stop case, the timepoint is true and the bus stops are false.</p>
generalizeLocation	<p>Optional (locationID_id).</p> <p>When isGeneralized is false, and the feature points to another location, then the specific record to which it points is included in this field. For example, WWM01 and WWM02 point to WWM locationID. The WWM location may be considered the timepoint location while the WWM01 and WWM02 locations may refer to bus stop locations.</p>

An example XML fragment of Location may appear as follows:

```

<Location>
  <locationID>100</locationID>
  <featureType>timepoint</featureType>
  <locationDesc>PARK AVE nearest bus stop RVPA and Phys Pt RVPA0</locationDesc>
  <intersection>PARK AVE @ NEWTON PL</intersection>
  <relativeLocation>
    <onStreet>PARK AVE</onStreet>
    <atStreet>NEWTON PL</atStreet>
  </relativeLocation>
  <longitude>-73.583913</longitude>
  <latitude>40.680051</latitude>
  <publicLocationDescription>PARK AVE at NEWTON PL</publicLocationDescription>
  <isGeneralized>>false</isGeneralized>
</Location>

```

Nested Element Descriptions

Nested elements are data concepts that are included as elements within other elements. Although they are not depicted in Figure 1 or Figure 2, they are complex types that are incorporated in the SDP mandatory elements.

Relative Location

Definition: A linear reference or attribute related to a Location. Fields include position relative to intersection, off or on transportation network, etc.

Table 12: RelativeLocation Element Description

Element	Description
---------	-------------

onStreet	Mandatory (string). The name of the street on which the feature is located; if the relative location is not on the street, then this is the nearest street location.
atStreet	Mandatory (string). The name of the cross street nearest the location.
distanceFromIntersection	Optional (distanceType). The distance in feet of the feature from the intersection. An attribute may be designated for either feet or meters. [See example in Transit Point Event distanceFromOrigin field.]
placementRelIntersection	Optional (placementRelIntersection_cd). The relative location of the location from the intersection. Enumerated type values include: [nearside; farside; midBlock; at; between; farsideMidBlock; nearsideMidBlock; opposite]
busPositionBay	Optional (integer). This is a number that designates the platform, bus bay, ferry berth or other access area that may not be defined in a TransitStop element.
alongLocation	Optional (alongLocation_cd). The location along a boarding area where passengers board or alight the transit vehicle. Valid enumerated values refer to the position based on the direction of travel: [right, left, both]
isOffStreet	Optional (boolean). The isOffStreet describes if the location is off the street. “true” implies that the location is not on a public road link; “false” assumes the location is along the street.

An example XML fragment of relativeLocation may appear as follows:

```
<relativeLocation>
  <onStreet>PARK AVE</onStreet>
  <atStreet>NEWTON PL</atStreet>
</relativeLocation>
```

AddressStructure

Definition: A description of the postal address of a place. Most of the elements in the address reference the US Post Office (USPS) Addressing Standard, USPS Publication 28, Postal Addressing Standards, August 1995 (updated 2006) which may be found at: <http://pe.usps.gov/text/pub28/welcome.htm>. Additional guidance on addressing guidance may be found at the USPS site.

Table 13: AddressStructure Element Descriptions

Element	Description
---------	-------------

addressID	Optional (identifier) UNIQUE. An internal index that references the address.
recordDate	Optional (date). The date of record insertion in the file or database.
addressNumber	Mandatory (string). The address number. This is a part of the complete address. See recommendation in USPS Publication 28 [http://pe.usps.com/cpim/ftp/pubs/pub28/pub28.pdf] ..
directionPrefix	Optional (string) The direction prefix for an address. Use one of the codes included in USPS. See Postal Service Standard Suffix Abbreviation [http://pe.usps.com/cpim/ftp/pubs/pub28/pub28.pdf] . Example: North as in East 5 th Avenue
typePrefix	Optional (string). The address prefix (other than direction prefix) for an address. Use one of the codes included in USPS. See Postal Service Standard Suffix Abbreviation [http://pe.usps.com/cpim/ftp/pubs/pub28/pub28.pdf] .
streetName	Mandatory (string). name of the street only (the prefix and suffix fields are designated by other fields)
typeSuffix	Mandatory (string). The address type suffix. Use one of the codes included in USPS. See Postal Service Standard Suffix Abbreviation [http://pe.usps.com/cpim/ftp/pubs/pub28/pub28.pdf] .
directionSuffix	Optional (string). The direction suffix for an address. Use one of the codes included in USPS. See Postal Service Standard Suffix Abbreviation [http://pe.usps.com/cpim/ftp/pubs/pub28/pub28.pdf] .
completeName	Mandatory (string). Complete post office address.
unitType	Optional (string). A unit, office or apartment.
unitDesignation	Optional (string). The number of the unit, office or apartment
secondLine	Optional (string). The 2nd line of an address if applicable.

postalCommunity	Mandatory (string). The city or community (e.g., Flatbush).
postalState	Mandatory (string). The state. The USPS Pub. 28 includes a complete list of FIPS codes.
postalCode	Mandatory (string). The postal code as designated by the USPS. See USPS zip+4 lookup site: http://zip4.usps.com/zip4/welcome.jsp
addressSegStatus	Optional (string). A classification for the status of the segment. Specific enumerated values are not designated in the SDP.

A typical SDP XML fragment for the NYSDOT headquarter address might look like --

```
<headquarterAddress>
  <addressNumber>50</addressNumber>
  <streetName>Wolf</streetName>
  <typeSuffix>RD</typeSuffix>
  <completeName>50 Wolf Road, Albany, NY 12205</completeName>
  <postalCommunity>Albany</postalCommunity>
  <postalState>NY</postalState>
  <postalCode>12205</postalCode>
</headquarterAddress>
```

Unique Identifiers

The identifiers listed in Table 13 should be unique within their primary or identifying element. The primary and related elements are included in the table.

Table 14: Unique Identifiers, Primary and Referencing Keys

Primary Key	Identifying Element	Referencing Element
addressed_id	AddressStructure	<ul style="list-style-type: none"> headquarterAddress in AgencyStructure address in TransitStopStructure
agencyID_id	AgencyStructure	<ul style="list-style-type: none"> agencyID in Portal, DayType,
locationID_id	LocationStructure	<ul style="list-style-type: none"> origin, destination (in PatternStructure) beginLocation, endLocation (in TripStructure) locationID (in TransitPointEventStructure and TripTimeStructure) generalizedLocation (in LocationStructure)
noteID_id	NoteStructure	<ul style="list-style-type: none"> noteList in TripStructure notes in TripTimeStructure
patternID_id	PatternStructure	<ul style="list-style-type: none"> patternID in RouteGrouping.patternSet patternID in TripStructure
pickNo_id	ScheduleVersion attribute structure	Not applicable
revisionNumber	ScheduleRevisionStructure	<ul style="list-style-type: none"> revisionNo in ScheduleCalendarDate.xsd

routeDepotVersionID	RouteDepotVersionStructure	<ul style="list-style-type: none"> • routeDepotVersionID in ScheduleCalendar Date.xsd
routeID_id	RouteStructure	<ul style="list-style-type: none"> • routeID in RouteDepotVersionStructure • routeGroupingID in RouteGroupingStructure • routeID in RouteGroupingStructure.tripSet and RouteGroupingStructure.patternSet • routeID in PatternStructure • routeID in TripStructure • fromRouteID and toRouteID in EventConnectionStructure
scheduleVersionID_id	SDP Document	<ul style="list-style-type: none"> • scheduleVersionID in ScheduleCalendarDate.xsd
seqNo_id	TransitPointEventStructure	Unique within a element list per SDP Document (element is not used to reference elements outside of structure)
stopID_id	TransitStopStructure	<ul style="list-style-type: none"> • stopID in TransitPointEventStructure • stopID in PlantComponentStructure
trackNo_id	TrackStructure	<ul style="list-style-type: none"> • trackNo in TransitPointEventStructure • trackNo in TransitPathEventStructure • trackNo in PlantComponentStructure • trackNo in TrackAssociationStructure
tripID_id	TripStructure	<ul style="list-style-type: none"> • blockTripList in BlockStructure • tripID in BlockTimeStructure • fromTripID and toTripID in EventConnectionStructure • tripID in RouteGrouping.tripSet

Data Type Descriptions

There are several data types (simple or complex types) that are described in the SDP Schema. The IDENTIFIER, Telephone and distanceType, found in the mandatory high level elements are described in Table 14.

Table 15: Data Type Descriptions

Type	Description
Description	(string) A data structure that requires one or more characters. A description is inserted in an element to ensure that the element is not left blank or null.
IDENTIFIER	(string) Identifying nested elements. See Table 1 for list of identifiers. A special notation of _id is used to describe a type referencing the data type IDENTIFIER. For example, agencyID_id. The XML Schema includes KEY and KEYREF declarations to manage referential integrity.

TELEPHONE	(string with a minimum of 10 characters) May be used to list the characters of a telephone number. A minimum of ten (10) characters should be included in the value.
distanceType	(float with attribute of units) The measure of distance and type of units used by the distance value. The units value is optional and if not included is assumed to be feet. The simple type units_cd enumerated values consist of [feet, meters].

Schedule Calendar XML Schema

Definition: The schedule calendar is a calendar of dates (scheduleCalendarDate) on which a transit agency assigns the type of service scheduled to be delivered (within its schedule day).

The scheduleCalendarDate is assigned a type of service using a code called a day type (dayType_cd). A day type enumeration includes most general service type codes as well as special days such as Veterans Day, Martin Luther King Day, Thanksgiving, Day After Thanksgiving, New Years Eve Day, and more. Altogether there are 56 values for dayType_cd. When a special day type value is not included in the list, you should complete a high level element called DayType. More details on how to use the DayType element are included in the Guidance document set (see <http://ngtsip.pbworks.com/library>).

Table 16: Schedule CalendarDateElement Descriptions

Element	Description
calendarDate	Mandatory. (date) A calendar date. If each dayType is associated with one and only one calendarDate, then this field is unique throughout the document. If there are trips in routes where a dayType will differ on the same date, then the routeDepotVersionID must be completed to distinguish the particular routes that provide the dayType service on that date. The combination of the calendarDate, revisionNumber and scheduleVersionID must be unique.
dayType	Mandatory (dayType_cd) A dayType_cd that is assigned to operate on the calendar date. See enumeration list for complete set of enumeration values.
placementDate	Mandatory (date) [yyyy-mm-dd] The date when this record was placed or generated.
activationDate	Optional. (date) [yyyy-mm-dd] The date when this dayType is valid
deactivationDate	Optional. (date) [yyyy-mm-dd] The date when this dayType becomes deactivated. If unknown, use 9999-12-31.

routeDepotVersionID	Optional. (routeDepotVersionID) When the day type differs for various routes, this field is included to reference the routeID that is valid.
revisionNo	Mandatory. (revisionNumber_id) The revisionNumber_id (from ScheduleRevision) of the route schedule assigned to this date. The identifier should match one in a registered SDP Document.
scheduleVersionID	Mandatory. (scheduleVersionID) The scheduleVersionID of the route schedule assigned to this date. The identifier should match one in a registered SDP Document.
agencyID	Optional. (agencyID_id) The agency name associated with this schedule date. The identifier should match a registered agency that submitted an SDP Document.

An example XML fragment of ScheduleCalendarDate may appear as follows:

```

<ScheduleCalendar>
  <scheduleCalendarDate>
    ...Add more scheduleCalendarDate elements here
    <calendarDate>2007-01-01</calendarDate>
    <dayType>sun</dayType>
    <placementDate>2006-12-26</placementDate>
    <activationDate>2006-9-4</activationDate>
    <deactivationDate>2007-01-02</deactivationDate>
    <revisionNo>0</revisionNo>
    <scheduleVersionID>306</scheduleVersionID>
  </scheduleCalendarDate>
</ScheduleCalendar>

```

Day Type Codes

Abbreviation	Code	Value	Description
sun	<xsd:enumeration value="sun"/>	1	Sunday
mon	<xsd:enumeration value="mon"/>	2	Monday
tue	<xsd:enumeration value="tue"/>	3	Tuesday
wed	<xsd:enumeration value="wed"/>	4	Wednesday
thu	<xsd:enumeration value="thu"/>	5	Thursday
fri	<xsd:enumeration value="fri"/>	6	Friday
sat	<xsd:enumeration value="sat"/>	7	Saturday
holiday	<xsd:enumeration value="holiday"/>	8	Holiday
weekday	<xsd:enumeration value="weekday"/>	9	Weekday (assume school open)
weekend	<xsd:enumeration value="weekend"/>	10	Weekend
weekday-closed	<xsd:enumeration value="weekday-closed"/>	11	Weekday, school closed
mon-wed	<xsd:enumeration value="mon-wed"/>	12	Monday, Tues, Wednesday
tu-th	<xsd:enumeration value="tu-th"/>	13	Tuesday to Thursday
mon-sat	<xsd:enumeration value="mon-sat"/>	14	Monday to Saturday
mon-thu	<xsd:enumeration value="mon-thu"/>	15	Monday to Thursday
daily	<xsd:enumeration value="daily"/>	16	Daily
sat-sun-hol	<xsd:enumeration value="sat-sun-hol"/>	17	Sat/Sun/Holiday
new-year-day	<xsd:enumeration value="new-year-day"/>	26	New Year's Day
president-day	<xsd:enumeration value="president-day"/>	27	President's Day
st-pat-day	<xsd:enumeration value="st-pat-day"/>	28	St. Patrick's Day
good-fri	<xsd:enumeration value="good-fri"/>	29	Good Friday
mem-day	<xsd:enumeration value="mem-day"/>	30	Memorial Day
ind-day	<xsd:enumeration value="ind-day"/>	31	Independence Day
lab-day	<xsd:enumeration value="lab-day"/>	32	Labor Day
col-day	<xsd:enumeration value="col-day"/>	33	Columbus Day
thanks	<xsd:enumeration value="thanks"/>	34	Thanksgiving
chris	<xsd:enumeration value="xmas"/>	35	Christmas
mlk-day	<xsd:enumeration value="mlk-day"/>	36	Martin Luther King Day
lincoln-day	<xsd:enumeration value="lincoln-day"/>	37	Lincoln's Birthday
washington-day	<xsd:enumeration value="washington-day"/>	38	Washington's Birthday

fri-before-Mem-day	<xsd:enumeration value="fri-before-Mem-day"/>	39	Friday Before Memorial Day
day-before-Ind-day	<xsd:enumeration value="day-before-Ind-day"/>	40	Day before July 4
fri-before-Labor	<xsd:enumeration value="fri-before-Labor"/>	41	Friday before Labor Day
election-day	<xsd:enumeration value="election-day"/>	42	Election Day
vets-day	<xsd:enumeration value="vets-day"/>	43	Veteran's Day
day-before-thanks	<xsd:enumeration value="day-before-thanks"/>	44	Day before Thanksgiving
day-after-thanks	<xsd:enumeration value="day-after-thanks"/>	45	Day after Thanksgiving
4-sat-before-chris	<xsd:enumeration value="4-sat-before-xmas"/>	46	4th Saturday before Christmas
4-sun-before-chris	<xsd:enumeration value="4-sun-before-xmas"/>	47	4th Sunday before Christmas
3-sat-before-chris	<xsd:enumeration value="3-sat-before-xmas"/>	48	3rd Saturday before Christmas
3-sun-before-chris	<xsd:enumeration value="3-sun-before-xmas"/>	49	3rd Sunday before Christmas
2-sat-before-chris	<xsd:enumeration value="2-sat-before-xmas"/>	50	2nd Saturday before Christmas
2-sun-before-chris	<xsd:enumeration value="2-sun-before-xmas"/>	51	2nd Sunday before Christmas
1-sat-before-chris	<xsd:enumeration value="1-sat-before-xmas"/>	52	Saturday before Christmas
1-sun-before-chris	<xsd:enumeration value="1-sun-before-xmas"/>	53	Sunday before Christmas
mon-before-chris	<xsd:enumeration value="mon-before-xmas"/>	54	Monday before Christmas
chris-eve	<xsd:enumeration value="xmas-eve"/>	55	Christmas Eve
new-year-eve	<xsd:enumeration value="new-year-eve"/>	56	New Year's Eve
wed-before-chris	<xsd:enumeration value="wed-before-xmas"/>	57	Wednesday before Christmas
thu-before-chris	<xsd:enumeration value="thu-before-xmas"/>	58	Thursday before Christmas
fri-before-chris	<xsd:enumeration value="fri-before-xmas"/>	59	Friday before Christmas
erev-yomtov-fa-sp	<xsd:enumeration value="erev-yomtov-fa-sp"/>	60	Shabbat / Jewish Holiday Eve (fall/spring)
erev-yomtov-su	<xsd:enumeration value="erev-yomtov-su"/>	61	Shabbat / Jewish Holiday Eve (summer)
erev-yomtov-wi	<xsd:enumeration value="erev-yomtov-wi"/>	62	shabbat / Jewish Holiday Eve (winter)
motzei-fa-sp	<xsd:enumeration value="motzei-fa-sp"/>	63	Motzei Shabbat / Jewish Holiday (fall/spring)
motzei-su	<xsd:enumeration value="motzei-su"/>	64	Motzei shabbat / Jewish Holiday (summer)
motzei-wi	<xsd:enumeration value="motzei-wi"/>	65	Motzei shabbat / Jewish Holiday (winter)

Additional Types			
sp-sun	sp-sun	18	Special Sundays
sp-mon	sp-mon	19	Special Mondays
sp-tue	sp-tue	20	Special Tuesdays
sp-wed	sp-wed	21	Special Wednesdays

sp-thu	sp-thu	22	Special Thursdays
sp-fri	sp-fri	23	Special Fridays
sp-sat	sp-sat	24	Special Saturdays
sp-daily	sp-daily	25	Special Daily

Sample SDP Document

For sample SDP version 2.0 Documents see <http://ngtsip.pbworks.com/w/page/34418345/Final-TSIP-Publications>.