

Bike Arlington Web services

Counters

Description

Counters are the access locations of a trail available for the use of bikers and pedestrians of Arlington Community.

This web service is intended for the retrieval of all the trails, access locations and the traffic data of bikes and pedestrians.

Common Terms Used:

Count: The total number of bicycle riders/pedestrians who accessed the trail location at a given point of time.

Mode: The type of transportation used by people while on the trail. There are two options available, B: Bikers, P: Pedestrians.

Direction: There are two types of directions available, inbound direction and outbound direction.

Interval: While requesting counter data there are 3 types of intervals you can give to retrieve the data, m: by the minute, h: by the hour, d: by the day.

Description of Methods

The following methods describe the type of methods available to retrieve the data:

1. GetAllRegions – Returns all the regions for which counters are available.
2. GetAllCounters - Returns all the counters by region.
3. GetCounter - Returns specific details related to a counter.
4. GetAllTrails - Returns all the trails.
5. GetCountInDateRange – Returns bike/pedestrians data count based on date range.
6. GetCountInDateList – Returns number of bikes/pedestrians on a given date within specific period of time.
7. getMaxDates – Returns the date of the last available data.
8. getMinDates -- Returns the first date since the data is available for the counter.

Service Link to access counters data:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=methodName>

METHOD 1: GetRegions

Description: Return all regions of the dashboard

Link: <http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetRegions>

Method 2: GetAllCounters

Description: Returns all counters (access locations) by region.

Request Fields: RegionID.

Example:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetAllCounters>

To Request by region pass RegionID as parameter:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetAllCounters®ionid=1>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counters>

    <counter id="10">
        <name>CC Connector</name>
        <description/>
        <trail_id/>
        <trail_name/>
        <latitude>38.857702</latitude>
        <longitude>-77.047373</longitude>
        <region>
            <name>Arlington</name>
            <id>1</id>
        </region>
    </counter>

    <counter id="3">
        <name>Custis Bon Air Park</name>
        <description/>
        <trail_id>2</trail_id>
        <trail_name>Custis</trail_name>
        <latitude>38.879199</latitude>
        <longitude>-77.138420</longitude>
        <region>
```

```

        <name>Arlington</name>
        <id>1</id>
    </region>

</counter>

<counter id="4">
    <name>Custis Rosslyn</name>
    <description/>
    <trail_id>2</trail_id>
    <trail_name>Custis</trail_name>
    <latitude>38.897191</latitude>
    <longitude>-77.083031</longitude>
    <region>
        <name>Arlington</name>
        <id>1</id>
    </region>

</counter>

<counter id="5">
    <name>Four Mile Run (piezo)</name>
    <description/>
    <trail_id>3</trail_id>
    <trail_name>Four Mile Run</trail_name>
    <latitude>38.843262</latitude>
    <longitude>-77.080860</longitude>
    <region>
        <name>Arlington</name>
        <id>1</id>
    </region>

</counter>

</counters>

```

Description of response fields:

- **id** – Id of the counter.
- **name** – Name of the counter.
- **description** – description of the counter.
- **trail_id** – Id of the trail.
- **trail_name** – name of the trail.
- **latitude** – geographical latitude of the counter.
- **longitude** – geographical longitude of the counter.
- **Region** – Region the counter belongs to.

Method 3: GetCounter.

Description: Returns specific details to a counter like trail name, direction and modes of transportation

Request Fields: CounterID.

Example:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCounter&counterid=1>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counters>

    <counter id="1">
        <name>W&OD East Falls Church</name>
        <description/>
        <trail_id>1</trail_id>
        <trail_name>W&OD</trail_name>
        <latitude>38.887806</latitude>
        <longitude>-77.163889</longitude>
        <directions>O,I</directions>
        <in_direction>EB</in_direction>
        <out_direction>WB</out_direction>
        <modes>B,P</modes>
        <region>
            <name>Arlington</name>
            <id>1</id>
        </region>
    </counter>

</counters>
```

Description of response fields:

- **id** – Id of the counter.
- **name** – Name of the counter.
- **description** – description of the counter.
- **trail_id** – Id of the trail.
- **trail_name** – name of the trail.
- **latitude** – geographical latitude of the counter.
- **longitude** – geographical longitude of the counter.
- **directions** – O: outbound, I : Inbound
- **in_direction** – Inbound direction is East Bound.
- **out_direction** – out bound direction is West Bound.
- **modes** – B: bikes, P: pedestrians.
- **Region** – Region the counter belongs to.

Method 4: GetAllTrails

Description: Returns all the trails.

Request Fields: None.

Example:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetAllTrails>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<trails>

    <trail id="1">
        <name>Custis</name>
    </trail>

</trails>
```

Description of response fields:

- **id** – Id of the trail.
- **name** – Name of the trail

Method 5: GetCountInDateRange

Description: – Returns bike/pedestrians data count based on date range.

Request Fields:

- CounterID – Number Required Field
- startDate – mm/dd/yyyy
- endDate – mm/dd/yyyy
- direction – I, O (I: Inbound, O: outbound), Empty for both.
- mode – B, P (B: bike, P: pedestrian), empty for both.
- startTime – HH:MM format

- endtime – HH:MM format
- interval – h (by the hourly), m(by the minutes), d(by the day).

Example with interval as d:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateRange&counterid=1&startDate=12/1/2011&endDate=12/04/2011&direction=I&mode=B&interval=d>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counts counter="1" endDate="12/4/2011" startDate="12/1/2011">
    <count count="1" date="12/01/2011" direction="I" mode="B"/>
    <count count="0" date="12/02/2011" direction="I" mode="B"/>
    <count count="2" date="12/03/2011" direction="I" mode="B"/>
    <count count="0" date="12/04/2011" direction="I" mode="B"/>
</counts>
```

Example with interval as h:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateRange&counterid=1&startDate=12/1/2011&endDate=12/01/2011&direction=I&mode=B&startTime=0:00&endTime=5:00&interval=h>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counts counter="1" endDate="12/4/2011" startDate="12/1/2011">
    <count count="0" date="12/01/2011" direction="I" hour="0" mode="B"/>
    <count count="1" date="12/01/2011" direction="I" hour="1" mode="B"/>
    <count count="0" date="12/01/2011" direction="I" hour="2" mode="B"/>
    <count count="0" date="12/01/2011" direction="I" hour="3" mode="B"/>
    <count count="0" date="12/01/2011" direction="I" hour="4" mode="B"/>
    <count count="0" date="12/01/2011" direction="I" hour="5" mode="B"/>
</counts>
```

Example with interval as m:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateRange&counterid=1&startDate=12/1/2011&endDate=12/01/2011&direction=I&mode=B&startTime=0:00&endTime=2:00&interval=m>

Response example:

```
<counts counter="1" endDate="12/1/2011" startDate="12/1/2011">

  <count count="0" date="12/01/2011" direction="I" hour="0" minute="0"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="0" minute="15"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="0" minute="30"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="0" minute="45"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="1" minute="0"
mode="B" />

  <count count="1" date="12/01/2011" direction="I" hour="1" minute="15"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="1" minute="30"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="1" minute="45"
mode="B" />

  <count count="0" date="12/01/2011" direction="I" hour="2" minute="0"
mode="B" />

</count>
```

Description of response fields:

- **id** – Id of the counter.
- **startDate** – start date of the request.
- **endDate** – end date of the request.
- **count** – Number of bikes or pedestrians.
- **date** – day of the count.
- **directions** – O: outbound, I: Inbound

- **modes** – B: bikes, P: pedestrians.
- **hour** – hour of the day.
- **minute** – minute of the hour.

Method 6: GetCountInDateList

Description: – Returns number of bikes/pedestrians on a given date within specific period of time.

Request Fields:

- CounterID – Number Required Field
- dates – mm/dd/yyyy
- direction – I, O (I: Inbound, O: outbound), Empty for both.
- mode – B, P (B: bike, P: pedestrian), empty for both.
- startTime – HH:MM format
- endTime – HH:MM format
- interval – h (data for every hourly), m(data for every minutes), d(data for the whole day).

Example with interval as d:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateList&counterid=1&Dates=12/1/2011&direction=I&mode=B&interval=d>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counts counter="1" dates="12/1/2011">
    <count count="242" date="12/01/2011" direction="I" mode="B"/>
</counts>
```

Example with interval as h:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateList&counterid=1&Dates=12/1/2011&direction=I&mode=B&startTime=0:00&endTime=5:00&interval=h>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<counts counter="1" Dates="12/1/2011">
  <count count="0" date="12/01/2011" direction="I" hour="0" mode="B"/>
  <count count="1" date="12/01/2011" direction="I" hour="1" mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="2" mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="3" mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="4" mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="5" mode="B"/>
</counts>
```

Example with interval as m:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetCountInDateRange&counterid=1&Dates=12/1/2011&direction=I&mode=B&startTime=0:00&endTime=2:00&interval=m>

Response example:

```
<counts counter="1" dates="12/1/2011">
  <count count="0" date="12/01/2011" direction="I" hour="0" minute="0"
mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="0" minute="15"
mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="0" minute="30"
mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="0" minute="45"
mode="B"/>
  <count count="0" date="12/01/2011" direction="I" hour="1" minute="0"
mode="B"/>
  <count count="1" date="12/01/2011" direction="I" hour="1" minute="15"
mode="B"/>
```

```
<count count="0" date="12/01/2011" direction="I" hour="1" minute="30"
mode="B"/>

<count count="0" date="12/01/2011" direction="I" hour="1" minute="45"
mode="B"/>

<count count="0" date="12/01/2011" direction="I" hour="2" minute="0"
mode="B"/>

</count>
```

Description of response fields:

- **id** – Id of the counter.
- **Dates** –date of the request.
- **count** – Number of bikes or pedestrians.
- **date** – day of the count.
- **directions** – O: outbound, I : Inbound
- **modes** – B: bikes, P: pedestrians.
- **hour** – hour of the day.
- **minute** – minute of the hour.

Method 7: getMaxDates

Description: Returns the date of the last available data.

Request Fields: None.

Example:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetMaxDates>

Response example:

```
<wddxPacket version='1.0'>
<header/>
  <data>
    <struct>
      <var name='STARTDATE'>
        <string>12/31/2011</string>
      </var>
      <var name='ENDDATE'>
        <string>12/31/2011</string>
      </var>
    </struct>
  </data>
</wddxPacket>
```

```
    </struct>
  </data>
</wddxPacket>
```

Method 8: getMinDates

Description: Returns the first date since the data is available for the counter.

Request Fields: CounterID.

Example:

<http://webservices.commuterpage.com/counters.cfc?wsdl&method=GetMinDates&CounterID=1>

Response example:

```
<wddxPacket version='1.0'><header/>
  <data>
    <struct>
      <var name='STARTDATE'>
        <string>07/09/2011</string>
      </var>
      <var name='ENDDATE'>
        <string>07/09/2011</string>
      </var>
    </struct>
  </data>
</wddxPacket>
```

Weather Data

Description

This web service is intended for the retrieval of weather data on a given date or range of dates. The type of weather data options available vary between temperature, type of weather like rain, snow, and thunder to dew point, pressure, humidity and fog.

Common Terms Used

Code: Refers to a weather condition.

Function: Refers to a calculating means such as Sum, Max, Min, Avg.

Daylimiter: Refers to the type of days data to be retrieved, such as, all Sundays in a given date range.

Units: Refers to the measuring unit of the weather conditions returned, such as, centigrade or Fahrenheit.

Mode: Refers to the calculated or summary of data.

Description of Methods

The following methods describe the type of methods available to retrieve the data:

1. GetCodes: Returns types of weather codes available.
2. GetData: Returns the weather data based on the requested weather code and the date.
3. GetDates: Returns the dates of weather data based on the requested weather value and the date.

Service Link to access counters data:

<http://webservices.commuterpage.com/weatherdata.cfc?wsdl&method=methodName>

Method 1: GetCodes

Description: Returns types of weather codes available.

Request Fields: None.

Example:

<http://webservices.commuterpage.com/weatherdata.cfc?wsdl&method=GetCodes>

Response example:

```
<weather_codes>

  <code name="dewpoint" binary="No" cumulative="No" units="F" />

  <code name="fog" binary="Yes" cumulative="No" units="" />

  <code name="hail" binary="Yes" cumulative="No" units="" />

</weather_codes>
```

Description of response fields:

- **name:** name of the weather code.
- **binary:** the response is independent. Yes or No
- **cumulative:** if the response is a cumulative value. Yes or No. Cumulative sum of hourly data.
- **units:** degrees in which the weather value is measured.

Method 2: GetData

Description: Returns the weather data based on the requested weather code and the date.

Request Fields:

- **dates:** string value, required field. Is a range of dates to retrieve the weather data, acceptable format yyyyymmdd-yyyyymmdd
- **code:** string value, required field. Supply the available weather codes from the getcodes response.
- **function:** string value, optional. Function to perform on the data (avg, sum, min, max); defaults to avg.
- **hours:** string value, optional. Hours in 24-hour format (0-23); may be comma-delimited list, hyphen-separated range, or special keywords "all" or "daytime"
- **dayLimiter:** numeric value, optional. Day of the week. Sun to Sat – 1 to 7 respectively, weekends – 8, sat & sun – 9, All - 0. Defaults to All - 0.

Example:

<http://webservices.commuterpage.com/weatherdata.cfc?wsdl&method=GetData&dates=20110301-20110305&code=temperature>

Response example:

```
<weather_data dates="20110301-20110305" hours="all" sample_frequency="all">
```

```
  <dataset>
```

```
    <date value="20110301">
```

```
      <data mode="calculated" function="avg" code="temperature" value="39.4"
units="F" />
```

```
    </date>
```

```
    <date value="20110302">
```

```
      <data mode="calculated" function="avg" code="temperature" value="46.4"
units="F" />
```

```
    </date>
```

```
    <date value="20110303">
```

```
      <data mode="calculated" function="avg" code="temperature" value="33.1"
units="F" />
```

```
    </date>
```

```
    <date value="20110304">
```

```
      <data mode="calculated" function="avg" code="temperature" value="39.7"
units="F" />
```

```
    </date>
```

```
    <date value="20110305">
```

```
      <data mode="calculated" function="avg" code="temperature" value="49.3"
units="F" />
```

```
    </date>
```

```
  </dataset>
```

</weather_codes>

Description of response fields:

- **mode:** name of the calculated mode. Either calculated or summary.
Calculated –calculated value of hourly data.
Summary – calculated value of a day's data.
- **function:** type of function used to calculate the value.
- **code:** type of weather code.
- **value:** value of the weather data code requested.
- **units:** measuring unit of the value returned.

Method 3: GetDates

Description: Returns the dates of weather data based on the requested weather value and the date.

Request Fields:

- **dates:** string value, required field. Is a range of dates to retrieve the weather data, acceptable format yyyyymmdd-yyyyymmdd
- **code:** string value, required field. Supply the available weather codes from the getcodes response.
- **Value:** string value, required field. value may be single value or hyphen-separated range. Example: If value is 70-75 and code is temperature, returns all the dates with a temperature between 70-75 degree.
- **function:** string value, optional. Function to perform on the data (avg, sum, min, max); defaults to avg.
- **hours:** string value, optional. Hours in 24-hour format (0-23); may be comma-delimited list, hyphen-separated range, or special keywords "all" or "daytime"
- **dayLimiter:** numeric value, optional. Day of the week. Sun to Sat – 1 to 7 respectively, weekends – 8, sat & sun – 9, All - 0. Defaults to All - 0.

Example:

<http://webservices.commuterpage.com/weatherdata.cfc?wsdl&method=GetDates&dates=20110301-20110315&code=temperature&value=70-75>

Response example:

```
<weather_data dates="20110301-20110315" code="tempi" value="40-45">
  <dataset>
    <date value="20110308">
      <data mode="calculated" function="avg" code="temperature" value="42.1"
units="F" />
    </date>
    <date value="20110309">
      <data mode="calculated" function="avg" code="temperature" value="42.0"
units="F" />
    </date>
    <date value="20110311">
      <data mode="calculated" function="avg" code="temperature" value="44.3"
units="F" />
    </date>
    <date value="20110314">
      <data mode="calculated" function="avg" code="temperature" value="44.9"
units="F" />
    </date>
    <date value="20110315">
      <data mode="calculated" function="avg" code="temperature" value="44.1"
units="F" />
    </date>
  </dataset>
</weather_codes>
```

Description of response fields:

- **mode:** name of the calculated mode. Either calculated or summary.
Calculated – calculated value of hourly data.
Summary – calculated value of a day's data.
- **function:** type of function used to calculate the value.
- **code:** type of weather code.
- **value:** value of the weather data code requested.
- **units:** measuring unit of the value returned.

Get News

Description

This web service provides all the latest news, updates and facts about Arlington. It also provides real-time schedules information of Art bus, Metro Bus, Metro Rail and George Bus and all the service related information about the respective transportations.

Description of Methods

The following methods describe the type of methods available to retrieve the data:

1. GetNews: Returns latest commuter news.
2. GetEvents: Returns all current events related to Arlington Commuter Page.
3. GetFacts: Returns latest facts related to Arlington Commuter Page.
4. GetArtAlerts: Returns latest information on real-time service disruptions and immediate emergency announcements for the Arlington Transit bus system.
5. GetGEORGEAlerts: Returns latest information on real-time service disruptions and immediate emergency announcements for the Falls Church GEORGE bus system.
6. GetMetroRailAlerts: Returns latest information on real-time service information on Metro Rail system.
7. GetMetroBusAlerts: Returns latest information on real-time service information for Metro Bus system.

Service Link to access counters data:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=methodName>

Method 1: GetNews

Description: Returns latest commuter news.

Request Fields:

- startDate: string value, optional. Is a start dates to retrieve the news data, acceptable format mm/dd/yyyy. Defaults to All.
- endDate: string value, optional. Is a end dates to retrieve the news data, acceptable format mm/dd/yyyy. Defaults to All.
- Topic: string value, optional. Can be comma delimited or | delimited. Defaults to home.
Available topics: home, atp ,cpage, bikearl, gtfs, carshare, ART, air, visit, crail, local, carpool, telework, paratran, taxi, store, direct, traffic, freepubs, metro, weather, mobilestor, walk, metrobus, carfree, cfd

- returncount: numeric value, optional field. Number of responses to return. Defaults to All.

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=Getnews>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>
    <title>CommuterPage.com Commuter News</title>

<link>http://webservices.commuterpage.com/services/news.cfc?method=GetNews</link
>
    <description>CommuterPage.com Commuter News</description>
    <language>en-us</language>
    <image>
      <title>CommuterPage.com Commuter News</title>
      <url>http://www.commuterpage.com/RSS/cplogo.gif</url>
      <link>http://www.commuterpage.com</link>
      <width>120</width>
      <height>50</height>
    </image>

    <copyright>Copyright 2012 Arlington County Commuter Services</copyright>
    <lastBuildDate>Tue, 01 May 2012 15:39:58 GMT</lastBuildDate>
    <docs>http://backend.userland.com/rss</docs>
    <generator>Arlington County Commuter Services /
CommuterPage.com</generator>
    <ttl>10</ttl>

    <item>
      <guid>20293</guid>
      <title>Super NoVa public meetings scheduled for May</title>
      <description>available Description.</description>
      <link>http://www.commuterpage.com/eventdetail.cfm?eventID=1564</link>
      <citation>April 26, 2012, Arlington County Commuter Services</citation>
      <pubDate>Thu, 26 Apr 2012 04:00:00 GMT</pubDate>
      <category>ART and Transit News</category>
```

```
<category>ATP News</category>
<category>BikeArlington News</category>
<category>Car-Free Diet News</category>
<category>Commuter Rail News</category>
<category>CommuterPage.com</category>
<category>Display on home page</category>
<category>Local Transit News</category>
<category>Metro System News</category>
<category>Traffic News</category>
<category>WalkArlington News</category>
</item>
</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the news item.
- **Title:** headline of the news title.
- **Description:** detailed news description.
- **Link:** referring link for the news.
- **Citation:** News published source.
- **pubDate:** News published date.
- **Category:** encloses all the related categories involved.

Method 2: GetEvents

Description: Returns all current events related to Arlington Commuter Page.

Request Fields:

- **startDate:** string value, optional. Is a start dates to retrieve the events data, acceptable format mm/dd/yyyy. Defaults to current date.
- **endDate:** string value, optional. Is a end dates to retrieve the events data, acceptable format mm/dd/yyyy. Defaults to current date.
- **Topic:** string value, optional. Can be comma delimited or | delimited. Defaults to home.

Available topics: home, atp ,cpage, bikearl, gtfs, carshare, ART, air, visit, crail, local, carpool, telework, paratran, taxi, store, direct, traffic, freepubs, metro, weather, mobilestor, walk, metrobus, carfree, cfd

- returncount: numeric value, optional field. Number of responses to return. Defaults to All.

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetEvents>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>
    <title>CommuterPage Events</title>

    <link>http://webservices.commuterpage.com/services/news.cfc?method=GetEvents</link>
    <description>Displays all current events related to Arlington Commuter Page</description>
    <lastBuildDate>Tue, 01 May 2012 16:57:48 GMT</lastBuildDate>

    <item>

      <guid>1562</guid>
      <title>Public Meeting: Super NoVA Transit/TDM Vision Plan</title>
      <description>Triangle, Virginia</description>
      <link>http://www.supernovatransitvision.com/</link>
      <link_target>0</link_target>
      <pubDate>Wed, 25 Apr 2012 04:00:00 GMT</pubDate>

    </item>

  </channel>
</rss>
```

Description of response fields:

- **guid**: identification number for the event item.
- **Title**: headline of the event title.
- **Description**: detailed event description.

- **Link:** referring link for the event.
- **pubDate:** event published date.

Method 3: GetFacts

Description: Returns latest facts related to Arlington Commuter Page.

Request Fields:

- **Date:** string value, optional. Is a dates to retrieve the fact of the day, acceptable format mm/dd/yyyy. Defaults to current date.
- **Topic:** string value, optional. Can be comma delimited or | delimited. Defaults to all.
- **returncount:** numeric value, optional field. Number of responses to return. Defaults to All.

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetFacts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>
    <title>CommuterPage Events</title>

    <link>http://webservices.commuterpage.com/services/news.cfc?method=GetFacts</link>
    <description>Displays all current facts related to Arlington Commuter Page</description>
    <lastBuildDate>Tue, 01 May 2012 16:57:48 GMT</lastBuildDate>

    <item>

      <guid>1562</guid>
      <title>Public Meeting: Super NoVA Transit/TDM Vision Plan</title>
```

```
<description>Triangle, Virginia</description>
<link>http://www.supernovatransitvision.com/</link>
<link_target>0</link_target>
<pubDate>Wed, 25 Apr 2012 04:00:00 GMT</pubDate>

</item>

</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the fact item.
- **Title:** headline of the fact.
- **Description:** detailed fact description.
- **Link:** referring link for the fact.
- **pubDate:** fact published date.

Method 4: GetARTAlerts

Description: Returns the latest information on real-time service disruptions and immediate emergency announcements for the Arlington Transit bus system.

Request Fields:

- **StartDate:** string value, optional. Is the start date to retrieve art alerts, acceptable format mm/dd/yyyy.
- **EndDate:** string value, optional. Is the end date to retrieve art alerts, acceptable format mm/dd/yyyy.
- **returncount:** numeric value, optional field. Number of responses to return. Defaults to All.

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetArtAlerts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<rss version="2.0">
<channel>
  <title>ART Disruptions</title>
  <link>http://webservices.commuterpage.com/services/news.cfc?method=GetARTAlerts</link>
  <description>ART Alert provides the latest information on real-time service disruptions and immediate emergency announcements for the Arlington Transit bus system.</description>
  <language>en-us</language>
  <image>
    <title>ART Alert Logo</title>
    <url>http://www.commuterpage.com/art/graphics/ArtAlert1.gif</url>
    <link>http://www.arlingtontransit.com</link>
    <width>109</width>
    <height>36</height>
  </image>
  <copyright>Copyright 2012 Arlington Transit</copyright>
  <lastBuildDate>Tue, 08 May 2012 15:13:36 GMT</lastBuildDate>
  <docs>http://backend.userland.com/rss</docs>
  <generator>Arlington Transit / CommuterPage.com</generator>
  <ttl>15</ttl>

  <item>
    <guid>3441</guid>
    <title>ART 41 - Delays Affecting Every Bus on the Route</title>
    <description>Buses serving ART 41 are running 8 Minutes Slower due to Traffic Congestion.</description>
    <link>http://www.arlingtontransit.com/pages/service-alerts/alert-details/?alertid=3441</link>
    <pubDate>2012-05-08 10:27:00.0</pubDate>
    <urgent>0</urgent>
  </item>

</channel>
</rss>

```

Description of response fields:

- **guid**: identification number for the alert item.
- **Title**: headline of the alert item.
- **Description**: detailed art service description.
- **Link**: referring link for the alert.
- **pubDate**: alert published date.

Method 5: GetGEORGEAlerts

Description: Returns the latest information on real-time service disruptions and immediate emergency announcements for the Falls Church GEORGE bus system.

Request Fields:

- StartDate: string value, optional. Is the start date to retrieve art alerts, acceptable format mm/dd/yyyy.
- EndDate: string value, optional. Is the end date to retrieve art alerts, acceptable format mm/dd/yyyy.
- returncount: numeric value, optional field. Number of responses to return. Defaults to All.

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetGEORGEAlerts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>
    <title>GEORGE Disruptions</title>

    <link>http://webservices.commuterpage.com/services/news.cfc?method=GetG
EORGEAlerts</link>
    <description>GEORGE Alert provides the latest information on real-
time service disruptions and immediate emergency announcements for the Falls Church
GEORGE bus system.</description>
    <language>en-us</language>
    <image>

      <title>GEORGE Alert Logo</title>

    <url>http://www.commuterpage.com/art/graphics/GeorgeLogo.gif</url>
```



```
<link>http://www.fallschurchva.gov/Content/CultureRecreation/GEORGEmain.a
spx</link>
    <width>119</width>
    <height>36</height>
</image>

<copyright>Copyright 2012 City of Falls Church, VA</copyright>
<lastBuildDate>Tue, 08 May 2012 16:21:04 GMT</lastBuildDate>
<docs>http://backend.userland.com/rss</docs>
<generator>CommuterPage.com</generator>
<ttl>15</ttl>

<item>

    <title>GEORGE is operating normally</title>
    <description>GEORGE is operating normally</description>
    <pubDate>Tue, 08 May 2012 16:21:04 GMT</pubDate>

</item>

</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the alert item.
- **Title:** headline of the alert item.
- **Description:** detailed service description.
- **Link:** referring link for the alert.
- **pubDate:** alert published date.
- **urgent:**

Method 6: GetMetroRailAlerts

Description: Returns the latest information on real-time metro rail disruptions.

Request Fields: None

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetMetroRailAlerts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
  <title>Washington Metropolitan Area Transit Authority Alerts</title>
  <link>http://www.wmata.com/</link>
  <description>Washington Metropolitan Area Transit Authority
Alerts</description>
  <language>en-us</language>
  <generator>eAlert Messaging System http://www.ealert.com/</generator>
  <webMaster>hostmaster@mis-sciences.com (MIS Hostmaster)</webMaster>
  <ttl>15</ttl>

  <item>
    <title>Org/ Blu Line:Trains are single tracking btwn Federal Center
& Eastern Market due to scheduled track work. ...More...</title>
    <link>http://www.wmata.com/</link>
    <pubDate>Tue, 08 May 2012 15:13:31 GMT</pubDate>
    <description>Org/ Blu Line:Trains are single tracking btwn Federal
Center & Eastern Market due to scheduled track work. Expect delays in both
directions.</description>
  </item>

</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the alert item.
- **Title:** headline of the alert item.
- **Description:** detailed rail service description.
- **Link:** referring link for the alert.
- **pubDate:** alert published date.

Method 7: GetMetroBusAlerts

Description: Returns the latest information on real-time metro bus disruptions.

Request Fields: None

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetMetroBusAlerts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
  <title>Washington Metropolitan Area Transit Authority Alerts</title>
  <link>http://www.wmata.com/</link>
  <description>Washington Metropolitan Area Transit Authority
Alerts</description>
  <language>en-us</language>
  <generator>eAlert Messaging System http://www.ealert.com/</generator>
  <webMaster>hostmaster@mis-sciences.com (MIS Hostmaster)</webMaster>
  <ttl>15</ttl>

  <item>
    <title>3A, 3B: Westbound buses detour at Lee Hwy & Old
Dominion Dr to resume regular route at Wakefield St & Old Dominion Dr due to
construction. ...More...</title>
    <link>http://www.wmata.com/</link>
    <pubDate>Tue, 08 May 2012 14:32:57 GMT</pubDate>
    <description>3A, 3B: Westbound buses detour at Lee Hwy & Old
Dominion Dr to resume regular route at Wakefield St & Old Dominion Dr due to
construction.</description>
  </item>
</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the alert item.
- **Title:** headline of the alert item.
- **Description:** detailed bus service description.
- **Link:** referring link for the alert.
- **pubDate:** alert published date.

Method 7: GetAllAlerts

Description: Returns the latest information on real-time metro bus disruptions.

Request Fields: None

Example:

<http://webservices.commuterpage.com/news.cfc?wsdl&method=GetAllAlerts>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
  <title>Washington Metropolitan Area Transit Authority Alerts</title>
  <link>http://www.wmata.com/</link>
  <description>Washington Metropolitan Area Transit Authority
Alerts</description>
  <language>en-us</language>
  <generator>eAlert Messaging System http://www.ealert.com/</generator>
  <webMaster>hostmaster@mis-sciences.com (MIS Hostmaster)</webMaster>
  <ttl>15</ttl>

  <item>
    <title>3A, 3B: Westbound buses detour at Lee Hwy & Old
Dominion Dr to resume regular route at Wakefield St & Old Dominion Dr due to
construction. ...More...</title>
    <link>http://www.wmata.com/</link>
    <pubDate>Tue, 08 May 2012 14:32:57 GMT</pubDate>
    <description>3A, 3B: Westbound buses detour at Lee Hwy & Old
Dominion Dr to resume regular route at Wakefield St & Old Dominion Dr due to
construction.</description>
  </item>
</channel>
</rss>
```

Description of response fields:

- **guld:** identification number for the alert item.
- **Title:** headline of the alert item.
- **Description:** detailed bus service description.
- **Link:** referring link for the alert.
- **pubDate:** alert published date.

BusRailSchedules

Description

This web services is used to retrieve all the operators, routes, stops and schedule information for all the both bus and rail.

Common Terms Used

Operator: Operator refers to the type of transportation service. For ex: Metro, ART, CUE etc.,

Route: Route refers to the path taken by a bus/train between two points of locations.

Stop: Stop refers to the point of location a bus/train stops to pick up or drop off.

Platform: Platform refers to the different locations at the same stop. Some stops can have multiple platforms. For Ex: Crystal City Metro is considered one stop with different platforms to pick up and drop off.

Schedule: Schedule refers to the time of day at which a bus/train is scheduled to arrive.

Description of Methods

The following methods describe the type of methods available to retrieve the data:

1. GetOperator: Returns all the operator types available.
2. GetAllRoutes: Returns all the routes available.
3. GetRoute: Returns route details for a given route Id.
4. GetAllStops: Returns all the stops available on a route.
5. GetStop: Returns stop location (latitude and longitude) and stop details.
6. GetAllMapStops: Returns all the stops available on a route along with direction.
7. GetARTRealtimeStops: Returns all the real time art stops on a route.
8. GetSchedule: Returns all the available operator schedules on a route along with stops.
9. GetAllDays: Returns all the days and the associated ID of the day.
10. GetDaysForRoute: Returns all the days the route operates.
11. GetDirectionsForRoute: Returns the direction a route operates.
12. GetRouteDay: Returns the current day of a route.
13. GetRoutesAtStop: Returns all types of bus operators at the stop along with all the routes and direction.
14. GetStopSchedules: Returns all the schedules at a given stop.
15. GetHolidays: Returns all the holidays of the year.

Service Link to access counters data:

<http://webservices.commuterpage.com/busrailschedule.cfc?wsdl&method=methodName>

Method 1: GetOperator

Description: Returns all the operator types available. For ex: ART, CUE, DASH etc.,

Request Fields:

- **OperatorID:** Numeric value, optional. Retrieves operator details based on ID. If not passed returns all operators.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetOperator&operatorId=3>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<operators_all>

<operator abbr="ART" icon="http://www.commuterpage.com/icons/art_icon.bmp" id="3"
logo="http://www.commuterpage.com/icons/art_logo.bmp" name="ART - Arlington
Transit" phone="703-228-RIDE" url="www.arlingtontransit.com"/>

</operators_all>
```

Description of response fields:

- **operator:**
 - **Abbr:** abbreviation for the operator name.
 - **Id:** operator id.
 - **Icon:** operator icon.
 - **Logo:** operator logo.
 - **Name:** full name of the operator.
 - **Phone:** contact number for the operator.
 - **URL:** operator url.

Method 2: GetAllRoutes

Description: Returns all the routes available.

Request Fields:

- OperatorID: Numeric value, optional. Is used to find the routes served by the operator. If not passed returns all the routes by all the operators.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetAllRoutes&operatorID=1>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<routes_all>
```

```
<route>
  <routeid>119</routeid>
  <schedcodeid>113</schedcodeid>
  <schedrouteid>17G-L</schedrouteid>
  <name>Kings Park Express</name>
  <direction1>Southbound</direction1>
```

```
  <direction2>Northbound</direction2>
  <description>Serves George Mason University, Burke Center Park & Ride,
  Burke Center VRE station, Kings Park West, Olde Forge, Burke, Kings Park, Danbury
  Forest, Pentagon station</description>
  <effdate>9/29/2002</effdate>
  <expdate>12/31/3000</expdate>
  <asymmetric>1</asymmetric>
  <sort>17</sort>
```

```
<bustrain>1</bustrain>
<maplat/>
<maplon/>
<mapzoomlevel/>
<mappolyline/>
<maplevels/>
<operatorid>1</operatorid>
<operatorname>Metrobus</operatorname>
```

```
<operatorurl>www.wmata.com</operatorurl>
<operatorabbr>Metrobus</operatorabbr>
```



```
<operatorphone>202-228-1234</operatorphone>  
</route>
```

```
</routes_all>
```

Description of response fields:

- **Route:** detailed information for each route.
 - **RouteID:** identification number for the route.
 - **SchedcodeId:** Identification numbers for the schedule, changes each time the schedule changes.
 - **Schedroutecode:** scheduled route code.
 - **Direction:** directions in which the route operates.
 - **Description:** description of the route (explains start location and end location along with major intersections).
 - **Effdate:** effective start date of the route in operation.
 - **Expdate:** end date of the route in operation (defaults to 12/31/3000 which means does not have a set end date).
 - **Asymmetric:** If 0, the route follows a dedicated path and has dedicated stops in both directions. If 1, the route can either have a different path or different stops in each direction.
 - **Sort:** sorting number for the route.
 - **Bustrain:** differentiates between a bus and a train. 1: bus, 2: train.
 - **Maplat:** map latitude. Usually empty for a route. Since route is the path followed by an operator.
 - **Maplon:** map longitude. Usually empty for route. Since route is the path followed by an operator.
 - **Mapzoomlevel:** zoom level on the map. Defines the resolution of the current view.
 - **Mappolyline:** polyline for the route.
 - **Maplevels:** encoded level on the map.
 - **OperatorId:** identification number for the operator on the route.
 - **Operatorname:** full name of the operator.
 - **Operatorurl:** url link for the operator.
 - **Operatorabbr:** abbreviation of the operator on the route.
 - **Operatorphone:** contact number of the operator.

Method 3: GetRoute

Description: Returns route details for a given route Id.

Request Fields:

- RouteID: Numeric value, required field.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetRoute&routeID=85>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<route>
  <routeid>119</routeid>
  <schedcodeid>113</schedcodeid>
  <schedrouteid>17G-L</schedrouteid>
  <name>Kings Park Express</name>
  <direction1>Southbound</direction1>

  <direction2>Northbound</direction2>
  <description>Serves George Mason University, Burke Center Park & Ride,
  Burke Center VRE station, Kings Park West, Olde Forge, Burke, Kings Park, Danbury
  Forest, Pentagon station</description>
  <effdate>9/29/2002</effdate>
  <expdate>12/31/3000</expdate>
  <asymmetric>1</asymmetric>
  <sort>17</sort>

  <bustrain>1</bustrain>
  <maplat/>
  <maplon/>
  <mapzoomlevel/>
  <mappolyline/>
  <maplevels/>
  <operatorid>1</operatorid>
  <operatorname>Metrobus</operatorname>

  <operatorurl>www.wmata.com</operatorurl>
  <operatorabbr>Metrobus</operatorabbr>
  <operatorphone>202-228-1234</operatorphone>
</route>
```

Description of response fields:

- **route:** detailed information for each route.
 - **RouteID:** identification number for the route.
 - **SchedcodeId:** Identification numbers for the schedule, changes each time the schedule changes.
 - **Schedroutecode:** scheduled route code.
 - **Direction:** directions in which the route operates.
 - **Description:** description of the route (explains start location and end location along with major intersections).
 - **Effdate:** effective start date of the route in operation.
 - **Exptime:** end date of the route in operation (defaults to 12/31/3000 which means does not have a set end date).
 - **Asymmetric:** If 0, the route follows a dedicated path and has dedicated stops in both directions. If 1, the route can either have a different path or different stops in each direction.
 - **Sort:** sorting number for the route.
 - **Bustrain:** differentiates between a bus and a train. 1: bus, 2: train.
 - **Maplat:** map latitude. Usually empty for a route. Since route is the path followed by an operator.
 - **Maplon:** map longitude. Usually empty for route. Since route is the path followed by an operator.
 - **Mapzoomlevel:** zoom level on the map. Defines the resolution of the current view.
 - **Mappolyline:** polyline for the route.
 - **Maplevels:** encoded level on the map.
 - **OperatorId:** identification number for the operator on the route.
 - **Operatorname:** full name of the operator.
 - **Operatorurl:** url link for the operator.
 - **Operatorabbr:** abbreviation of the operator on the route.
 - **Operatorphone:** contact number of the operator.

Method 4: GetAllStops

Description: Returns all the stops and their Ids on a route.

Request Fields:

- RouteID: Numeric value, required field.
- DayID: Numeric value, optional.
- DirectionID: Numeric value, optional.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetAllStops&routeID=85>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<stops_all dayid="-2" directionid="0" routeid="85">
  <stops>
    <stop abbr="W Falls Ch" id="11" lat="38.9007184185" lon="-77.188911438" name="W Falls Church Metro" sequence="10"/>
    <stop abbr="Westpark" id="62" lat="38.9319391789" lon="-77.2297668457" name="Tysons-Westpark" sequence="20"/>
    <stop abbr="IntlGreens" id="402" lat="38.923451" lon="-77.226602" name="International & Greensboro" sequence="30"/>
    <stop abbr="Westpark" id="62" lat="38.9319391789" lon="-77.2297668457" name="Tysons-Westpark" sequence="40"/>
    <stop abbr="W Falls Ch" id="11" lat="38.9007184185" lon="-77.188911438" name="W Falls Church Metro" sequence="50"/>
  </stops>
</stops_all>
```

Description of response fields:

- **DayID:** identification number for the day. Can get DayID's using GetAllDays method.
- **DirectionID:** identification number for the direction. Varies based on route, refer to GetRoute or GetAllRoutes methods.
- **RouteID:** identification number for the route.
- **Abbr:** abbreviation of the stop name.
- **Id:** identification number for the stop.
- **Lat:** stop latitude.
- **Lon:** stop longitude.
- **Sequence:** the sequence of the stop on a route.

Method 5: GetStop

Description: Returns stop location (latitude and longitude) and stop details.

Request Fields:

- StopID: Numeric value, required field.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetStop&stopID=11>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<stop_all>
    <stop abbr="W Falls Ch" id="11" lat="38.9007184185" lon="-77.188911438"
name="W Falls Church Metro"/>
</stop_all>
```

Description of response fields:

- **Abbr:** abbreviation of the stop name.
- **Id:** identification number for the stop.
- **Lat:** stop latitude.
- **Lon:** stop longitude.
- **Name:** full name of the stop.

Method 6: GetAllMapStops

Description: Returns all the stops available on a route along with direction

Request Fields:

- RouteID: Numeric value, required field.
- DayID: Numeric value, optional.
- DirectionID: Numeric value, optional.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetAllMapStops&routeID=85>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<stops_all dayid="-2" directionid="0" routeid="85">
  <stops>
    <stop abbr="W Falls Ch" dir="1" id="11" lat="38.9007184185" lon="-77.188911438" name="W Falls Church Metro" sequence="10"/>
    <stop abbr="Westpark" dir="1" id="62" lat="38.9319391789" lon="-77.2297668457" name="Tysons-Westpark" sequence="20"/>
    <stop abbr="IntlGreens" dir="1" id="402" lat="38.923451" lon="-77.226602" name="International & Greensboro" sequence="30"/>
    <stop abbr="Westpark" dir="1" id="62" lat="38.9319391789" lon="-77.2297668457" name="Tysons-Westpark" sequence="40"/>
    <stop abbr="W Falls Ch" dir="1" id="11" lat="38.9007184185" lon="-77.188911438" name="W Falls Church Metro" sequence="50"/>
  </stops>
</stops_all>
```

Description of response fields:

- **DayID:** identification number for the day. Can get DayId's using GetAllDays method.
- **DirectionId:** identification number for the direction. Varies based on route, refer to GetRoute or GetAllRoutes methods.
- **RouteId:** identification number for the route.
- **Abbr:** abbreviation of the stop name.
- **Dir:** direction id of the route. Refer to GetRoute or GetAllRoutes methods to find direction details.
- **Id:** identification number for the stop.
- **Lat:** stop latitude.
- **Lon:** stop longitude.
- **Sequence:** the sequence of the stop on a route.

Method 7: GetARTRealtimeStops

Description: Returns all the real time stops available on a route along with direction.

Request Fields:

- RouteID: Numeric value, required field.
- DirectionID: Numeric value, optional.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetARTRealtimeStops&routeID=123>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<stops_all dayid="-2" directionid="1" routeid="943">
  <stops>
    <stop dir="1" dirname="Westbound" id="74001183" lat="38.86285913"
lon="-77.05901953" name="Pentagon Cty Metro, 12th St S, EB @ S Hayes St, FS"
platformNo="74001" platformTag="183" rc="84" sequence="1010" timepoint="1"/>
    <stop dir="1" dirname="Westbound" id="82101543" lat="38.84817863"
lon="-77.08111329" name="24th Road S, WB @ S Glebe Road , FS"
platformNo="82101" platformTag="543" rc="84" sequence="1020" timepoint="1"/>
    <stop dir="1" dirname="Westbound" id="840041088"
lat="38.84825350" lon="-77.08460203" name="S Kenmore Street, NB @ 24th Street S,
NS" platformNo="84004" platformTag="1088" rc="84" sequence="1030" timepoint="0"/>
    <stop dir="1" dirname="Westbound" id="840051085"
lat="38.84971936" lon="-77.08542501" name="S Kenmore Street, NB @ 22nd Street S,
NS" platformNo="84005" platformTag="1085" rc="84" sequence="1040" timepoint="0"/>
  </stops>
</stops_all>
```

Description of response fields:

- **Id**: combination of platform number and platform tag.

- **DirectionId:** identification number for the direction. Direction varies based on route, refer to GetRoute or GetAllRoutes methods.
- **RouteId:** identification number for the route.
- **Abbr:** abbreviation of the stop name.
- **Dir:** direction Id of the route. Refer to GetRoute or GetAllRoutes methods to find direction details.
- **Dirname:** name of the direction.
- **Lat:** stop latitude.
- **Lon:** stop longitude.
- **Name:** full name of the stop.
- **Platformno:** the platform number at the stop. Some stops can have multiple platforms for different buses/trains.
- **Platformtag:** Identification number for the platform. Can be you used to retrieve real time schedule information for a platform.
- **RC:** schedules bus/train number.
- **Sequence:** the sequence of the stop on a route.
- **Timepoint:** Is a binary number, differentiates if a stop has a scheduled time of arrival or not.

Method 8: GetSchedule

Description: Returns all the available operator schedules on a route along with stops.

Request Fields:

- RouteID: Numeric value, optional.
- DepartStopID: Numeric value, optional.
- ArriveStopID: Numeric value, optional.
- DayID: Numeric value, optional.
- DirectionID: Numeric value, optional.
- ReturnCount: Numeric value, optional. Number of schedules to return.
- TimeFormat: Numeric value, optional.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetSchedule&routeID=123>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<schedule arrivestopabbr="N/A" arrivestopname="N/A" dayid="0" dayname="Weekday"
departstopabbr="N/A" departstopname="N/A" directionid="1"
```


directionname="Eastbound" effdate="6/21/2010" operatorabbr="ART"
operatoricon="http://www.commuterpage.com/icons/art_icon.bmp" operatorid="3"
operatorlogo="http://www.commuterpage.com/icons/art_logo.bmp" operatorname="ART
- Arlington Transit" operatorphone="703-228-RIDE"
operatorurl="www.arlingtontransit.com" routecode="84" routedesc="Connects Douglas
Park, Nauck and 24th St. S at S. Glebe Road to the Pentagon City Metro during rush
hours." routeid="943" routename="Douglas Park-Nauck-Pentagon City"
schedcodeid="270"
scheduleURL="http://www.commuterpage.com/schedules/sched.cfm?id=270">

```
<runs>
  <run accessible="1" highlight="0" id="1" notes="" runcode="84">
    <stop abbr="Monroe&16" id="1122" name="Monroe St. & 16th St.
S." time="5:56a"/>
    <stop abbr="WReed&19" id="1289" name="Walter Reed & 19th St
S" time="6:01a"/>
    <stop abbr="S24&Glebe" id="75" name="24th Rd. & S.Glebe"
time="6:04a"/>
    <stop abbr="Pent City" id="469" name="Pentagon City Metro" time="6:15a"/>
  </run>
  <run accessible="1" highlight="0" id="2" notes="" runcode="84">
    <stop abbr="Monroe&16" id="1122" name="Monroe St. & 16th St.
S." time="6:26a"/>
    <stop abbr="WReed&19" id="1289" name="Walter Reed & 19th St
S" time="6:31a"/>
    <stop abbr="S24&Glebe" id="75" name="24th Rd. & S.Glebe"
time="6:34a"/>
    <stop abbr="Pent City" id="469" name="Pentagon City Metro" time="6:45a"/>
  </run>
</runs>
</schedule>
```

Description of response fields:

- **Arrivestopabbr:** arrival stop name abbreviation.
- **Arrivestopname:** full name of arrival stop.
- **DayId:** Identification number for ID. Refer GetAllDays method for day ID's.
- **Dayname:** name of the day.
- **Departstopabbr:** departing stop abbreviation.
- **Departstopname:** departing stop full name.

- **DirectionId:** identification number for the direction. Direction varies based on route; refer to GetRoute or GetAllRoutes methods.

- **Directionname:** name of the direction.
- **EffDate:** effective start date since the route has been active.
- **OperatorId:** identification number for the operator on the route.
- **Operatorname:** full name of the operator.
- **Operatorurl:** url link for the operator.
- **Operatorabbr:** abbreviation of the operator on the route.
- **Operatorphone:** contact number of the operator.
- **RouteId:** identification number for the route.
- **Routename:** brief route name with major intersections or stops.
- **RouteCode:** scheduled bus/train number.
- **Routerdesc:** route description.
- **SchedcodeId:** Identification number for the schedule, changes each time the schedule changes.
- **ScheduleURL:** schedule link. Provides the full schedule for the route code.
- **Runs:** lists all the schedule information for the route code for each trip of the operator.
 - **Run:** individual schedule information for the route code.
 - **Accessible:** if the schedule is accessible.
 - **Highlight:** if the schedule refers to current time.
 - **Id:** sequence number of the trip by the operator.
 - **Notes:** schedule notes.
 - **Runcode:** schedules bus or train number.
 - **Stop:** encloses individual stop description.
 - **Abbr:** stop name abbreviation.
 - **Id:** identification number for the stop.
 - **Name:** name of the stop.
 - **Time:** scheduled time the operator arrives at the stop.

Method 9: GetAllDays

Description: Returns all the days and the associated ID of the day.

Request Fields: None.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetAllDays>

Response Exmample:

```
<?xml version="1.0" encoding="UTF-8"?>
<days_all>
  <day id="1" name="Monday"/>
  <day id="2" name="Tuesday"/>
</days_all>
```

Description of response fields:

- **Day:** is the description of the day.
- **Id:** identification number for the day.
- **Name:** name of the day corresponding to the ID.

Method 10: GetDaysForRoute

Description: Returns all the days the route operates.

Request Field:

- **RouteId:** numeric value, required field.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetDaysForRoute&routeid=123>

Response Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<days_all route="943">
  <day id="0" name="Weekday"/>
</days_all>
```

Description of response fields:

- **Id:** identification number for the day the route operates.
- **Name:** name of the day the route operates.

Method 11: GetDirectionsForRoute

Description: Returns the direction a route operates.

Request Fields:

- **RouteId:** numeric value, required field.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetDirectionsForRoute&routeid=123>

Response Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<directions_all route="943">
    <direction id="1" name="Eastbound"/>
    <direction id="2" name="Westbound"/>
</directions_all>
```

Description of response fields:

- **Id:** identification number for the direction.
- **Name:** name of the direction a route operates.

Method 12: GetRouteDay

Description: Returns the current day of a route.

Request Fields:

- RouteId: Numeric value, required field.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetRouteDay&routeid=123>

Response Example:

```
<wddxPacket version='1.0'>
```

```
<header/>
```

```
  <data>
```

```
    <string>3</string>
```

```
  </data>
```

```
</wddxPacket>
```

Description of response fields:

- **Data:** encloses the current day ID of the route.

Method 13: GetRoutesAtStop

Description: Returns all types of bus operators at the stop along with all the routes and direction.

Request Fields:

- stopID: Numeric value, required.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetRoutesAtStop&stopId=1>

Response Example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<stop abbr="W Falls Ch" id="11" name="W Falls Church Metro">
```

```
  <routes>
```

```
    <operator abbreviation="FC" id="10" name="Fairfax Connector">
```

```
      <route codeid="78" fullname="425" id="844" name="425">
```

```
        <direction id="1" name="Round Trip"/>
```

```
        <direction id="1" name="Round Trip"/>
```

```
      </route>
```

```
      <route codeid="79" fullname="427" id="85" name="427">
```

```
        <direction id="1" name="Round Trip"/>
```

```
        <direction id="1" name="Round Trip"/>
```

```
      </route>
```

```
    </operator>
```

```
  <operator abbreviation="LC" id="9" name="Loudoun County Transit">
```

```
    <route codeid="54" fullname="" id="1102" name="901">
```

```
      <direction id="1" name="AM Eastbound"/>
```

```
    </route>
```

```
<route codeid="54" fullname="" id="1102" name="902">
    <direction id="1" name="AM Eastbound"/>
</route>
</operator>
</routes>
</stop>
```

Description of response fields:

- **Stop:**
 - **Abbr:** abbreviation.
 - **Id:** identification number of the stop.
 - **Name:** name of the stop.
- **Operator:**
 - **Abbreviaton:** abbreviation of the operator name.
 - **Id:** operator Id.
 - **Name:** operator name.
- **Route:**
 - **codeId:** identification number for the schedule, changes each time the schedule changes.
 - **Fullname:** all the operating bus/trains at the stop on the route.
 - **Id:** route Id.
 - **Name:** Name of the bus/train on the route. Name with which the bus/train is identified. For ex: 5A. 5A is the name of the bus.
- **Direction:**
 - **Id:** Id number of the direction in which the bus/train is headed on the route.
 - **Name:** name of the direction in which the bus/train is headed on the route.

Method 14: GetStopSchedules

Description: Returns all the schedules at a given stop.

Request Fields:

- StopId: Numeric value, required.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetStopSchedules&stopId=11>

Response Example:

```
<transit_schedules last_update="5/19/2009 3:52PM">
  <stop id="60" name="Tysons Center" abbr="Tysons Ctr">
    <route id="805" name="Metrobus 15M">
      <direction directionid="1" directionname="Eastbound"
destination="Franconia-Springfield Metro" via="">
        <day id="0" name="weekday">
          <schedule>
            <run time="3:38 PM" />
            <run time="4:41 PM" />
            <run time="5:22 PM" />
            <run time="6:07 PM" />
          </schedule>
        </day>
      </direction>
      <direction directionid="2" directionname="Westbound"
destination="Crystal City Metro" via="">
        <day id="0" name="weekday">
          <schedule>
            <run time="3:38 PM" />
```



```
                <run time="4:41 PM" />
                <run time="5:22 PM" />
                <run time="6:07 PM" />
            </schedule>
        </day>
    </direction>
</route>
</stop>
</transit_schedules>
```

Description of response fields:

- **Stop:**
 - **Abbr:** abbreviation.
 - **Id:** identification number of the stop.
 - **Name:** name of the stop.
- **Route:**
 - **Id:** identification number for the route.
 - **Fullname:** full name of the route.
 - **Name:** Name with which the bus/train is identified. For ex: 5A. 5A is the name of the bus.
- **Direction:**
 - **Id:** Id number of the direction in which the bus/train is headed on the route.
 - **Name:** name of the direction in which the bus/train is headed on the route.
- **Day:**
 - **Id:** identification number for the day.
 - **name:** day of the week the schedule is retrieved for.
- **Schedule:**
 - **Date:** date on which the bus/train arrives at the stop.
 - **Time:** time at which the bus/train arrives at the stop.

Method 15: GetHolidays

Description: Returns all the holidays from the current month up to a year.

Request Fields: None.

Example:

<http://webservices.commuterpage.com/busrailschedules.cfc?wsdl&method=GetHolidays>

Response Example:

```
<?xml version="1.0" encoding="UTF-8"?>

<holidays last_update="01/01/2012 5:04:23 PM" last_update_GMT="Sun, 01 Jan 2012
21:04:23 GMT">

    <holiday holiday_date="01/15/2012" holiday_name="Martin Luther King Day "
id="1"/>

    <holiday holiday_date="02/15/2012" holiday_name="Presidents Day" id="2"/>

    <holiday holiday_date="5/28/2012" holiday_name="Memorial Day" id="3"/>

</holidays>
```

Description of response fields:

- **Holidat_date:** Date of the holiday.
- **Holiday_name:** Name of the holiday.

Additional Information

Data for individual counters can be exported to Microsoft Excel, directly from the Counter Dashboard. After setting parameters and charting results in the dashboard, click the icon for any counter to create a graph of data for that counter. Each graph includes links to print the graph and to export graph data to an Excel file.

Weather data: Powered by Weather Underground (<https://www.wunderground.com/>)

Web Services: More comprehensive data is available in the form of two web services

1. <http://webservices.commuterpage.com/counters.cfc?wsdl>
2. <http://webservices.commuterpage.com/weatherdata.cfc?wsdl>

Application developers are welcome to use this data in their own applications.