

Allocator Agent – Frequently Asked Questions

Jurisdiction Determination

Q. Describe how the software determines whether an address is inside or outside of the city limits (via zip code, geocode, postal address, etc.).

A. Premium Pro Allocator and Agent use a 3rd party GIS geo-coding product. The software incorporates the USPS zip+4 product to determine if an address falls within a legitimate address range. The geocoding is based on the most precise coordinates that are available for the address.

Q. How often is the standardizing data used by the software updated?

A. Quarterly

Q. Describe how the software responds to an “invalid” address (i.e. an address it cannot identify the location/jurisdiction). Are several response codes available to help determine why it was not found?

A. The software does not have a problem identifying the location or jurisdiction for an address. To be specific, a matching address range cannot be found based on the information provided. If a close matching address is not found, the software will either return the message "no close match" or "no candidates found". The software will not return a matching address in either situation. To maximize the possibility of a close match, your company (when integrating the Agent product with you own systems) needs to prevent a user from transmitting an incomplete address (Address, City, State, and Zip Code must be transmitted).

Q. What data is needed as input to the software: street, city, state, zip, county?

A. You should always provide street number, street name, state and zip code. A match may be made if you omit the city or zip code (you cannot omit both), but you decrease the likelihood when doing so (should there be an error in the city name or zip code). The county information is not needed.

Q. Is jurisdiction determination available for any other states?

A. At this time, we only offer KY in our web based format.

Q. Does your software include a stand-alone look-up mechanism (screen? Web page?) that can be used by business users to enter an address and get the taxing jurisdictions for manually processed policies?

- A. The product is designed to interface with your in house system or in house web base solution (such as a company website). There is an input screen that can be accessed from a web browser.

Tax Rate Determination

Q. Are the tax rates for all of Kentucky's premium categories (Casualty, Inland Marine, etc.) included?

A. Yes

Q. Does the software determine and return the applicable tax rates by effective date or does it only return the rates in effect as of the date of the call?

A. If you provide an effective date, it will use that date. Otherwise, it will default to the current date.

Q. What data is needed as input?

A. The effective date of the policy (optional), complete address, password, and ID must be transmitted by you in order for Agent to transmit the address.

Q. Does the software calculate the tax amount, or only return the tax rates?

A. It does not calculate the tax amount, it returns the applicable tax rate.

Q. When are the tax rate updates provided by Kentucky applied? How long does it take those updates to reach the users of the software?

A. The new bulletin is typically issues during the first 2 weeks of April. Agent will be updated at the end of April.

Scenarios

Q. What happens if no taxes apply at all?

A. The county information will be returned, but no tax rate is returned.

Q. Does your software adjust the tax rates to account for premiums that must be reported to both the county and the city? The Kentucky annual bulletin has special instructions for county returns that must be filed with a LGT 142 and for those counties that are "grandfathered".

A. Yes. A effective tax rate is returned that accounts for the additional tax due.

Technical

Q. Is the software set up as a Web Service?

A. Yes

Q. What online protocols are supported?

A. Soap 1.1, Soap 1.2, HTTP Get, HTTP Post (Please access the website for more information).

Q. What is the average online response time?

A. That is dependent on the number of hits per second. At this time, our response time is unaffected by the amount of traffic we are receiving. We can send you stress tests for your review that simulate 20 or 100 simultaneous users.

Q. Is it the intent of your company to pursue certification with the state of Kentucky for tax jurisdiction determination? What is your timeline to apply for/obtain certification?

A. Yes. The state of Kentucky has began to process applications. We are currently working on our data set per the requirements of the verification manual and expect to have it completed by May.

Q. What is the price of the software based on: is it transaction count, policy count, etc.?

A. Transaction count

Q. What Production support is included with the software?

A. TriTech provides support during normal business hours: M-F, 8 to 5 CST. We will answer any questions about the application or make any necessary corrections. We don't provide support to help you integrate the results we transmit with your in house systems or websites.

PREMIUM Pro Allocator Agent offers a robust, scalable, and easy-to-integrate online solution that enables businesses to become location intelligent. This allows you to provide accurate premium allocation data with no interruption to the critical point of sale or policy/agent management system. PREMIUM Pro Allocator Agent is a web-based policy allocation solution specifically designed to allow insurance companies to more accurately pinpoint policy locations in real-time, and to assign the correct premium tax rates to each policy at the point of sale.

As a Microsoft.NET-based web service, PREMIUM Pro Allocator Agent can be quickly and easily integrated into your company's existing website or policy/agent management system. With a Company ID and Password (provided by TriTech Software) and a few lines of code, any insurance company can immediately begin to take advantage of TriTech Software's time-tested and industry-proven experience in the field of premium tax.

Process Flow:

1. Add reference to Allocator Agent web service (this will differ depending on the system used. Can be any system capable of communicating with web services, including .NET, Java, COBOL, ASP, SOAP, XML, or virtually any modern language).
2. Connect to the Allocator Agent web service and supply the source address and policy information.
3. Receive and process XML-format allocation results.



Agent utilizes Pitney Bowes MapInfo software which returns a matching address and plots the address to a digital map of Kentucky. The digital map is created from boundary information supplied by the KY Commonwealth Office of Technology. Tax information is supplied by the KY Department of Insurance LGT Tax Schedule and Bulletin.

Before you can utilize the Allocator Agent Web Service, you must obtain a password and id from your TriTech representative.

The Allocator Agent Web Service may be accessed at the following web address:

<https://agent.tritechsoft.com/AllocatorAgent/>

When you pass a valid address and premium type code (line of business) to Agent, the service will return a matching address, tax jurisdiction, and tax rate information.

For a formal definition of the web service, click on the link for Service Description. The WSDL will be displayed.

To test an address, click the link for AllocateAddress. Complete the test form to allocate an address. The html page will also list:

1. A sample Soap 1.1 request and response.
2. A sample Soap 1.2 request and response.
3. A sample HTTP get request and response.
4. A sample HTTP post request and response.

ENTERING AN ADDRESS

To allocate an address using the test form (Valid element names are bolded):

1. Enter the **CompanyID** and **Password**.
2. Enter a full address (**SourceAddress**). The **SourceAddress** is street address (**SourceStreet**), city name (**SourceCity**), and zip code (**SourceZipCode**). Do not enter a P.O. Box, Rural Route, or Highway Contract address. These addresses are not physical addresses and will not be evaluated.
3. Type in **LineOfBusiness** for the policy. Valid codes are:
 - C - Casualty
 - F – Fire and Allied Perils
 - H – Health
 - I – Inland Marine
 - L – Life
 - M – Motor Vehicle
 - O – Other
4. **EffectiveDate**: This is optional. If omitted, the **EffectiveDate** for a policy is assumed to be today's date. The **EffectiveDate** can affect the tax rate assigned to a policy.

The test form is merely a means to create a request in xml. When you implement Allocator Agent, you may use HTML/SOAP request and response methods to send and

receive the requests in xml format. The address will be processed and returned in XML format.

XML ELEMENTS RETURNED

Valid element names are bolded:

1. **SourceAddress** - The address you typed in.
 - **Street**
 - **City**
 - **State**
 - **ZipCode**
2. **LineOfBusiness** – The original line of business code you entered will be returned. See Entering an Address on page 1 for the 7 valid codes.
3. **MatchedAddress** –An address is returned from the address database, if a close match to your source address can be found.
 - **Street**
 - **City**
 - **State**
 - **ZipCode**

If an address was not returned, it is due to the following reasons:

- The address is a PO Box.
 - The address is a Rural Route or Highway Contract Route.
 - A similar address could not be found. Either the address is invalid or a similar address could not be found in the database.
4. **CensusBlockID** – The Census Block ID of the **MatchedAddress** contains relevant US Census information.
 5. **CountyFIPSCode** – The county FIPS code is derived from the **CensusBlockID**. The first 2 digits identify the state jurisdiction and the last 3 digits identify the county jurisdiction of the **MatchedAddress**.
 6. **CensusCountyName** – The county of the **MatchedAddress** is determined by plotting address coordinates to a Kentucky map. **CensusCountyName** is always returned; *it does not* indicate that the county taxes the policy.
 7. **CensusCountyCode** – If **CensusCountyName** has a corresponding tax code in the KY Annual Insurance Bulletin, it will be returned. This *does not indicate* the county taxes the policy. If there is no tax code, the number 9998 will be returned.
 8. **MatchQuality** – In order for a **MatchedAddress** to be returned, **MatchQuality** must be rated as a close match. Pitney Bowes MapInfo uses its own algorithm to determine what a close match is. The street or house number of a **MatchedAddress** must be identical to the number in the **SourceAddress** to be considered a close match. Other elements of the address are evaluated to

determine if the **MatchedAddress** is a good candidate for the **SourceAddress** provided.

- Single Match. 1 close match was returned that was better than all other addresses.
- Multiple Matches. 2 or more closely matching addresses are determined to be suitable candidates. One of these addresses will be returned.
- No Close Match. An address closely corresponding to the source address was not found. No **MatchedAddress** will be returned.
- No Candidates. No address corresponding to the source address was found. No **MatchedAddress** will be returned.
- PO Box or Rural Route. The **SourceAddress** or **MatchedAddress** was determined to be a PO Box, Rural Route, or HC Route. No **MatchedAddress** will be returned.

9. **MatchCode** –

- 0 indicates a single match.
- 1 indicates multiple matches.
- 2 indicates no close match.
- 3 indicates no matches.
- 4 indicates a PO Box, RR, or HC Route.

10. **MatchIdentifier** –Indicates which parts of a **MatchedAddress** are an exact match to its **SourceAddress**. The **MatchIdentifier** contains 10 characters that represent different parts of the address. If Agent is able to match part of an address, a character will appear in the match identifier at the position corresponding to that part of the address. If not, a dash will appear instead. The **MatchIdentifier** positions are as follows:

Geocode Type

The first character in the **MatchIdentifier** is always: S – Street Type

Positional Accuracy

The second character indicates the positional accuracy of the candidate’s point.

Number	Positional Accuracy
8	A single point represents the address
7	Interpolated points along a street segment/parcel
6	Centroid of a site or building with a dedicated zip code
5	Interpolated points along a street segment
4	Street centroid
3	ZIP + 4 centroid
2	ZIP + 2 centroid
1	ZIP centroid
0	No coordinates available

Address Position

These positions describe the matches in the different elements in the address.

Position	Letter Code	Address Position
1	H	House Number
2	P	Street Prefix
3	N	Street Name
4	T	Street Type
5	S	Street Suffix
6	C	City Name
7	Z	Zip Code

Example: The **MatchIdentifier** S5HPN-S-Z is interpreted as follows:

- The point is a street type known as an interpolated street segment.
- The house number, street prefix, street name, street suffix and zip code of the **MatchedAddress** are identical to the **SourceAddress**.
- The street type and city name in the **MatchedAddress** are not identical to the **SourceAddress**. This could be due to spelling or it may be completely different.

11. **CityCode** –This corresponds to the City Code that appears on the KY Dept of Insurance Local Government Premium Tax Schedule. When a **MatchedAddress** is returned, Agent plots coordinates to a map to determine the taxing jurisdiction.
 - If the address falls within the boundaries of a city that taxes policies, the **CityCode** for the city is returned.
 - If the address falls outside a city that taxes or inside a city that does not tax, the address is assigned to the county. The county **CityCode** is returned.
 - If a **CityCode** is not returned in the above 2 steps; a code of 9998 will be returned. This indicates that the policy is not taxable.
12. **CityName** –The name of the city or county corresponding to the **CityCode** will be returned. **CityName** may or may not match the city found in the **MatchedAddress**. This is the actual physical location, not the name listed in the physical address. The word “NONTAXABLE” will be returned if a **CityCode** of 9998 is returned.
13. **CityOrCounty** – If the **CityCode** and **CityName** returned are for a city, the word *City* will be returned. If the code and name are for a county, the word *County* will be returned. If the policy is nontaxable, nothing will be returned.
14. **TaxCode** –**TaxCode** is returned for any municipality or county that has one. It is listed on Kentucky Local Government Premium Tax Schedule and the Local Government Premium Tax Code Descriptions. The maximum length for this element is 10.
15. **CityTaxRates** – The tax rates for each line of business will be returned for the **CityCode**. If the **CityCode** is 9998, no rate will be returned. These rates are from the Kentucky Local Government Premium Tax Schedule.

- **Casualty**
 - **Fire**
 - **Health**
 - **Inland Marine**
 - **Life**
 - **Motor Vehicle**
 - **Other**
 - **MinTax** (Minimum Tax will be returned if applicable. A minimum tax cannot be assessed on a policy by policy basis. It can only be assessed on total premiums)
 - **FlatFee** (A flat fee cannot be assessed on a policy by policy basis. It can only be assessed on total premiums)
16. **CountyCreditApplies** – If premiums taxable by a city are also taxable by a county, *true* will be returned. Otherwise *false* will be returned. This applies to cities with a **TaxCode** of *A* or *B*.
17. **CountyCode** – If **CountyCreditApplies** is *true*, the county's **CityCode** will be returned.
18. **CountyName** – If **CountyCreditApplies** is *true*, the county's **CityName** will be returned.
19. **CountyTaxRates** – If **CountyCreditApplies** is *true*, the county's **CityTaxRates** will be returned.
- **Casualty**
 - **Fire**
 - **Health**
 - **Inland Marine**
 - **Life**
 - **Motor Vehicle**
 - **Other**
 - **MinTax** (Minimum Tax will be returned if applicable. A minimum tax cannot be assessed on a policy by policy basis. It can only be assessed on total premium),
 - **FlatFee** (A flat fee cannot be assessed on a policy by policy basis. It can only be assessed on total premium).
20. **EffectiveDate** – Returns the **EffectiveDate** you entered. Otherwise the date is the current date. Tax rates returned are from this period.
21. **EffectiveTaxRate** – An **EffectiveTaxRate** is returned for the **LineOfBusiness** you specified. If you do not specify a **LineOfBusiness**, the **EffectiveTaxRate** is zero. The **EffectiveTaxRate** rate may be multiplied by the premium amount to determine the total tax amount to be passed to the insured.
- If **CountyCreditApplies** is *false*, the **EffectiveTaxRate** is pulled from **CityTaxRates**,
 - If **CountyCreditApplies** is *true*, the **EffectiveTaxRate** is pulled from **CityTaxRates** or **CountyTaxRates** (whichever is higher).

- If **CountyCreditApplies** is *true*, but the city name is Springfield, Hickman, or Lawrenceburg, the effective tax rate will be the sum of the **CityTaxRates** and **CountyTaxRates**.
22. **EffectiveCityRate** – The **EffectiveCityRate** is returned for the **LineOfBusiness** you specified. This is pulled from **CityTaxRates**. The **EffectiveCityRate** can be applied to premiums to determine the tax payable to the city or county.
- If **CountyCreditApplies** is *false*, the **EffectiveCityRate** is the same as the **EffectiveTaxRate**. may be multiplied by the premiums to determine
 - If **CountyCreditApplies** is *true*, the **EffectiveCityRate** is the same as the **CityTaxRates**.
23. **EffectiveCountyRate** – This is the **EffectiveTaxRate** less the **EffectiveCityRate**. The **EffectiveCountyRate** can be applied to premiums to determine the tax due to the county ONLY when both the county and city tax the same premiums.
24. **ErrorDetails** –
- If no error occurs, **Description** remains blank and **Number** returned is 0.
 - Error **Description** and **Number** for errors are:
 - Geocoder Server error, 1000
 - No CompanyID was specified, 1601
 - No Password was specified, 1602
 - Invalid CompanyID, 1603
 - Incorrect Password, 1604
 - Account suspended, 1605
 - Tax rate not available for this date, 1606
 - Incorrect Premium Type Code, 16071
 - Premium Type Code Missing, 16072
 - Unspecified Error, 1600

AllocateAddress

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
CompanyID:	<input type="text"/>
Password:	<input type="text"/>
SourceStreet:	<input type="text"/>
SourceCity:	<input type="text"/>
SourceState:	<input type="text"/>
SourceZipCode:	<input type="text"/>
LineOfBusiness:	<input type="text"/>
EffectiveDate:	<input type="text"/>

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The [placeholders](#) shown need to be replaced with actual values.

```
POST /allocatoragent/service.asmx HTTP/1.1
Host: agent.tritechsoft.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "https://agent.tritechsoft.com/AllocatorAgent/AllocateAddress"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AllocateAddress xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
      <CompanyID>string</CompanyID>
      <Password>string</Password>
      <SourceStreet>string</SourceStreet>
      <SourceCity>string</SourceCity>
      <SourceState>string</SourceState>
      <SourceZipCode>string</SourceZipCode>
      <LineOfBusiness>string</LineOfBusiness>
      <EffectiveDate>string</EffectiveDate>
    </AllocateAddress>
  </soap:Body>
</soap:Envelope>
```

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AllocateAddressResponse
xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
      <AllocateAddressResult>
        <SourceAddress>
          <Street>string</Street>
          <City>string</City>
          <State>string</State>
          <ZipCode>string</ZipCode>
        </SourceAddress>
        <LineOfBusiness>string</LineOfBusiness>
        <MatchedAddress>
          <Street>string</Street>
          <City>string</City>
          <State>string</State>
          <ZipCode>string</ZipCode>
        </MatchedAddress>
        <CensusBlockID>string</CensusBlockID>
        <CountyFipsCode>string</CountyFipsCode>
        <CensusCountyName>string</CensusCountyName>
        <CensusCountyCode>string</CensusCountyCode>
        <MatchQuality>
          <MatchCode>short</MatchCode>
          <MatchDescription>string</MatchDescription>
          <MatchIdentifier>string</MatchIdentifier>
        </MatchQuality>
        <CityCode>string</CityCode>
        <CityName>string</CityName>
        <CityOrCounty>string</CityOrCounty>
        <TaxCode>string</TaxCode>
        <CityTaxRates>
          <Casualty>double</Casualty>
          <Fire>double</Fire>
          <Health>double</Health>
          <InlandMarine>double</InlandMarine>
          <Life>double</Life>
          <MotorVehicle>double</MotorVehicle>
          <Other>double</Other>
          <MinTax>double</MinTax>
          <FlatFee>double</FlatFee>
        </CityTaxRates>
        <CountyCreditApplies>boolean</CountyCreditApplies>
        <CountyCode>string</CountyCode>
        <CountyName>string</CountyName>
        <CountyTaxRates>
          <Casualty>double</Casualty>
          <Fire>double</Fire>
          <Health>double</Health>
```

```

    <InlandMarine>double</InlandMarine>
    <Life>double</Life>
    <MotorVehicle>double</MotorVehicle>
    <Other>double</Other>
    <MinTax>double</MinTax>
    <FlatFee>double</FlatFee>
  </CountyTaxRates>
  <EffectiveDate>string</EffectiveDate>
  <EffectiveTaxRate>double</EffectiveTaxRate>
  <EffectiveCityRate>double</EffectiveCityRate>
  <EffectiveCountyRate>double</EffectiveCountyRate>
  <ErrorDetails>
    <Number>int</Number>
    <Description>string</Description>
    <Source>string</Source>
  </ErrorDetails>
</AllocateAddressResult>
</AllocateAddressResponse>
</soap:Body>
</soap:Envelope>

```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The [placeholders](#) shown need to be replaced with actual values.

```

POST /allocatoragent/service.asmx HTTP/1.1
Host: agent.tritechsoft.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <AllocateAddress xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
      <CompanyID>string</CompanyID>
      <Password>string</Password>
      <SourceStreet>string</SourceStreet>
      <SourceCity>string</SourceCity>
      <SourceState>string</SourceState>
      <SourceZipCode>string</SourceZipCode>
      <LineOfBusiness>string</LineOfBusiness>
      <EffectiveDate>string</EffectiveDate>
    </AllocateAddress>
  </soap12:Body>
</soap12:Envelope>
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">

```

```
<soap12:Body>
  <AllocateAddressResponse
xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
  <AllocateAddressResult>
    <SourceAddress>
      <Street>string</Street>
      <City>string</City>
      <State>string</State>
      <ZipCode>string</ZipCode>
    </SourceAddress>
    <LineOfBusiness>string</LineOfBusiness>
    <MatchedAddress>
      <Street>string</Street>
      <City>string</City>
      <State>string</State>
      <ZipCode>string</ZipCode>
    </MatchedAddress>
    <CensusBlockID>string</CensusBlockID>
    <CountyFipsCode>string</CountyFipsCode>
    <CensusCountyName>string</CensusCountyName>
    <CensusCountyCode>string</CensusCountyCode>
    <MatchQuality>
      <MatchCode>short</MatchCode>
      <MatchDescription>string</MatchDescription>
      <MatchIdentifier>string</MatchIdentifier>
    </MatchQuality>
    <CityCode>string</CityCode>
    <CityName>string</CityName>
    <CityOrCounty>string</CityOrCounty>
    <TaxCode>string</TaxCode>
    <CityTaxRates>
      <Casualty>double</Casualty>
      <Fire>double</Fire>
      <Health>double</Health>
      <InlandMarine>double</InlandMarine>
      <Life>double</Life>
      <MotorVehicle>double</MotorVehicle>
      <Other>double</Other>
      <MinTax>double</MinTax>
      <FlatFee>double</FlatFee>
    </CityTaxRates>
    <CountyCreditApplies>boolean</CountyCreditApplies>
    <CountyCode>string</CountyCode>
    <CountyName>string</CountyName>
    <CountyTaxRates>
      <Casualty>double</Casualty>
      <Fire>double</Fire>
      <Health>double</Health>
      <InlandMarine>double</InlandMarine>
      <Life>double</Life>
      <MotorVehicle>double</MotorVehicle>
      <Other>double</Other>
      <MinTax>double</MinTax>
      <FlatFee>double</FlatFee>
    </CountyTaxRates>
    <EffectiveDate>string</EffectiveDate>
```

```

    <EffectiveTaxRate>double</EffectiveTaxRate>
    <EffectiveCityRate>double</EffectiveCityRate>
    <EffectiveCountyRate>double</EffectiveCountyRate>
    <ErrorDetails>
      <Number>int</Number>
      <Description>string</Description>
      <Source>string</Source>
    </ErrorDetails>
  </AllocateAddressResult>
</AllocateAddressResponse>
</soap12:Body>
</soap12:Envelope>

```

HTTP GET

The following is a sample HTTP GET request and response. The [placeholders](#) shown need to be replaced with actual values.

```

GET
/allocatoragent/service.asmx/AllocateAddress?CompanyID=string&Password=string
&SourceStreet=string&SourceCity=string&SourceState=string&SourceZipCode=string
&LineOfBusiness=string&EffectiveDate=string HTTP/1.1
Host: agent.tritechsoft.com
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<CallResults xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
  <SourceAddress>
    <Street>string</Street>
    <City>string</City>
    <State>string</State>
    <ZipCode>string</ZipCode>
  </SourceAddress>
  <LineOfBusiness>string</LineOfBusiness>
  <MatchedAddress>
    <Street>string</Street>
    <City>string</City>
    <State>string</State>
    <ZipCode>string</ZipCode>
  </MatchedAddress>
  <CensusBlockID>string</CensusBlockID>
  <CountyFipsCode>string</CountyFipsCode>
  <CensusCountyName>string</CensusCountyName>
  <CensusCountyCode>string</CensusCountyCode>
  <MatchQuality>
    <MatchCode>short</MatchCode>
    <MatchDescription>string</MatchDescription>
    <MatchIdentifier>string</MatchIdentifier>
  </MatchQuality>
  <CityCode>string</CityCode>
  <CityName>string</CityName>
  <CityOrCounty>string</CityOrCounty>
  <TaxCode>string</TaxCode>
  <CityTaxRates>

```

```

    <Casualty>double</Casualty>
    <Fire>double</Fire>
    <Health>double</Health>
    <InlandMarine>double</InlandMarine>
    <Life>double</Life>
    <MotorVehicle>double</MotorVehicle>
    <Other>double</Other>
    <MinTax>double</MinTax>
    <FlatFee>double</FlatFee>
  </CityTaxRates>
  <CountyCreditApplies>boolean</CountyCreditApplies>
  <CountyCode>string</CountyCode>
  <CountyName>string</CountyName>
  <CountyTaxRates>
    <Casualty>double</Casualty>
    <Fire>double</Fire>
    <Health>double</Health>
    <InlandMarine>double</InlandMarine>
    <Life>double</Life>
    <MotorVehicle>double</MotorVehicle>
    <Other>double</Other>
    <MinTax>double</MinTax>
    <FlatFee>double</FlatFee>
  </CountyTaxRates>
  <EffectiveDate>string</EffectiveDate>
  <EffectiveTaxRate>double</EffectiveTaxRate>
  <EffectiveCityRate>double</EffectiveCityRate>
  <EffectiveCountyRate>double</EffectiveCountyRate>
  <ErrorDetails>
    <Number>int</Number>
    <Description>string</Description>
    <Source>string</Source>
  </ErrorDetails>
</CALloResults>

```

HTTP POST

The following is a sample HTTP POST request and response. The [placeholders](#) shown need to be replaced with actual values.

```

POST /allocatoragent/service.asmx/AllocateAddress HTTP/1.1
Host: agent.tritechsoft.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

CompanyID=string&Password=string&SourceStreet=string&SourceCity=string&Source
State=string&SourceZipCode=string&LineOfBusiness=string&EffectiveDate=string
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<CALloResults xmlns="https://agent.tritechsoft.com/AllocatorAgent/">
  <SourceAddress>
    <Street>string</Street>
    <City>string</City>

```



```
<State>string</State>
  <ZipCode>string</ZipCode>
</SourceAddress>
<LineOfBusiness>string</LineOfBusiness>
<MatchedAddress>
  <Street>string</Street>
  <City>string</City>
  <State>string</State>
  <ZipCode>string</ZipCode>
</MatchedAddress>
<CensusBlockID>string</CensusBlockID>
<CountyFipsCode>string</CountyFipsCode>
<CensusCountyName>string</CensusCountyName>
<CensusCountyCode>string</CensusCountyCode>
<MatchQuality>
  <MatchCode>short</MatchCode>
  <MatchDescription>string</MatchDescription>
  <MatchIdentifier>string</MatchIdentifier>
</MatchQuality>
<CityCode>string</CityCode>
<CityName>string</CityName>
<CityOrCounty>string</CityOrCounty>
<TaxCode>string</TaxCode>
<CityTaxRates>
  <Casualty>double</Casualty>
  <Fire>double</Fire>
  <Health>double</Health>
  <InlandMarine>double</InlandMarine>
  <Life>double</Life>
  <MotorVehicle>double</MotorVehicle>
  <Other>double</Other>
  <MinTax>double</MinTax>
  <FlatFee>double</FlatFee>
</CityTaxRates>
<CountyCreditApplies>boolean</CountyCreditApplies>
<CountyCode>string</CountyCode>
<CountyName>string</CountyName>
<CountyTaxRates>
  <Casualty>double</Casualty>
  <Fire>double</Fire>
  <Health>double</Health>
  <InlandMarine>double</InlandMarine>
  <Life>double</Life>
  <MotorVehicle>double</MotorVehicle>
  <Other>double</Other>
  <MinTax>double</MinTax>
  <FlatFee>double</FlatFee>
</CountyTaxRates>
<EffectiveDate>string</EffectiveDate>
<EffectiveTaxRate>double</EffectiveTaxRate>
<EffectiveCityRate>double</EffectiveCityRate>
<EffectiveCountyRate>double</EffectiveCountyRate>
<ErrorDetails>
  <Number>int</Number>
  <Description>string</Description>
  <Source>string</Source>
```

```
</ErrorDetails>  
</CAlloResults>
```