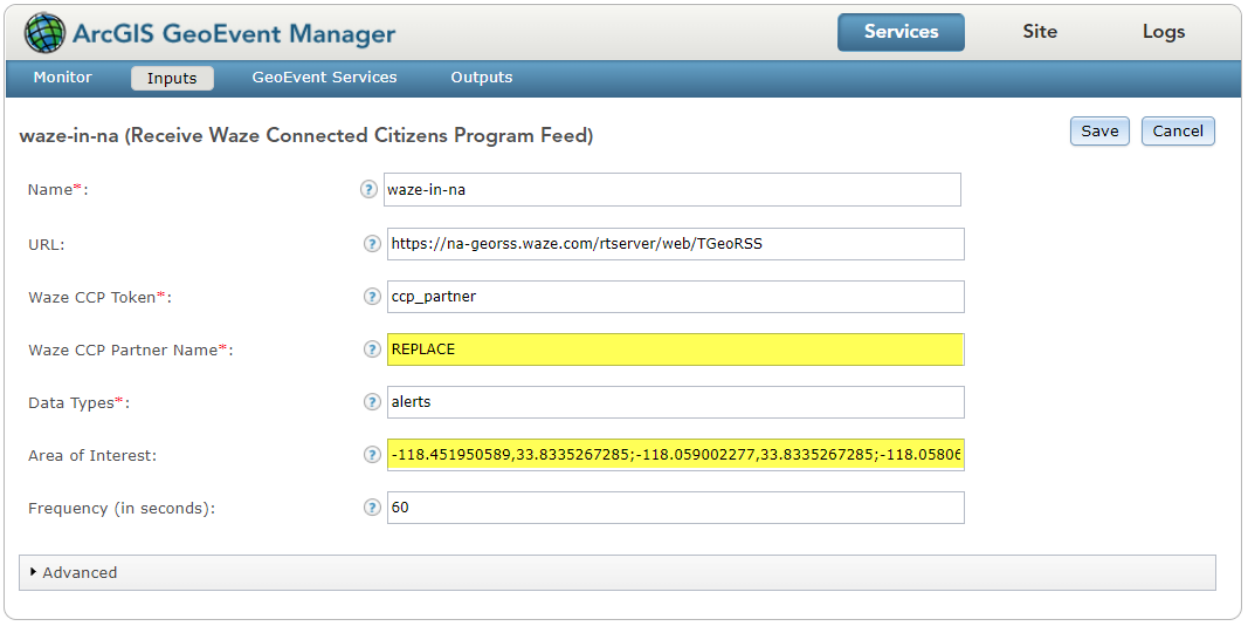
GeoEvent Server / Waze / Cityworks Integration

The solution detailed below consists of three key components, the Waze Connector, the Event Volume Control Processor, and the Message Formatter Adapter. The first two items are add-on components available on the GeoEvent Gallery and GitHub respectively. The third component is included with GeoEvent Server but is not utilized by the default / preconfigured connectors. To use that adapter, you would need to create a new connector and attach it with the appropriate transport.

To simplify the deployment process, we have included the necessary compiled jar files for the add-on components, and provided a configuration file that has prebuilt the Message Formatter outputs. Unless already available in your system, please import the provided components and then load the configuration.

The screenshots below highlight the components of the service that you will need to modify:

Replace the Waze CCP Partner Name and Area of Interest with your approved values

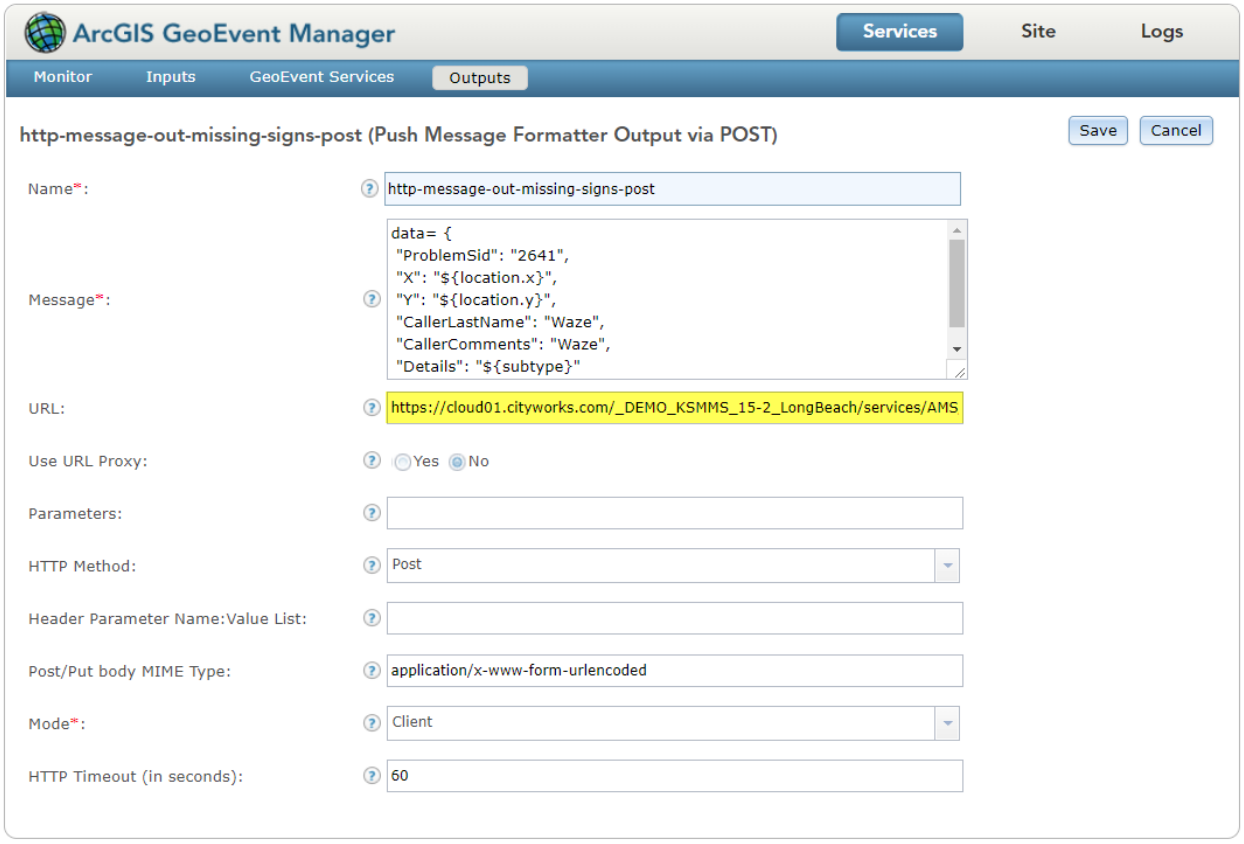


Edit each of the provided outputs replacing the URL parameter with the services endpoint of your Cityworks server including a token in the following format:

https://cloud01.cityworks.com/MY_CITYWORKS_SERVER/services/AMS/ServiceRequest/create?token=eyJFbXBsb3llZVNpZCI6ODEwMywiSXNzdWVkVGltZSI6MTUwOTU0NjUwNzczNCwiTG9naW5OYW1lIjoiU1RSRUVUUyIsIlNpZ25hdHVyZSI6InhKekY4MnpWR2h4dnJobjd1MlVzcUJhbStiZ3dtQ1Fyd3EwOFVkZlVnK2s9IiwiVG9rZW4iOiJzalJqZTVTV1RBSVhEUlpVZU0ycXo4L1VJb2wzUWdJVGNnbjNIdm1YV2NFPSJ9

To generate that token, you will need to make an Authentication request from a browser directly to your Cityworks server's API. Here's an example of that request (taken from the Cityworks documentation):

```
http://localhost/Cityworks/Services/General/Authentication/Authenticate?data=
{"LoginName":"jdoe","Password":"jdoe"}
```



You can include additional service request types by creating copies of the outputs and modifying the ProblemSid parameter of the JSON Message block.

```
data= {
"ProblemSid": "2641",
"X": "${location.x}",
"Y": "${location.y}",
"CallerLastName": "Waze",
"CallerComments": "Waze",
"Details": "${subtype}"
}
```

By default, the service will accept any matching alert with a reliability of 2 or higher. If the volume of new reports is too high you can set the threshold higher (up to a maximum score of 10).

While not implemented in the provided solution, you could further reduce the number of provided service requests by connecting to the Cityworks Feature Service, create buffers around those features which can be written to a Stream Service, then used as Synchronized/Dynamic GeoFences to filter for new requests (e.g. not capturing service requests for potholes in the same area as open requests).