ArcGIS Runtime: Network Analysis

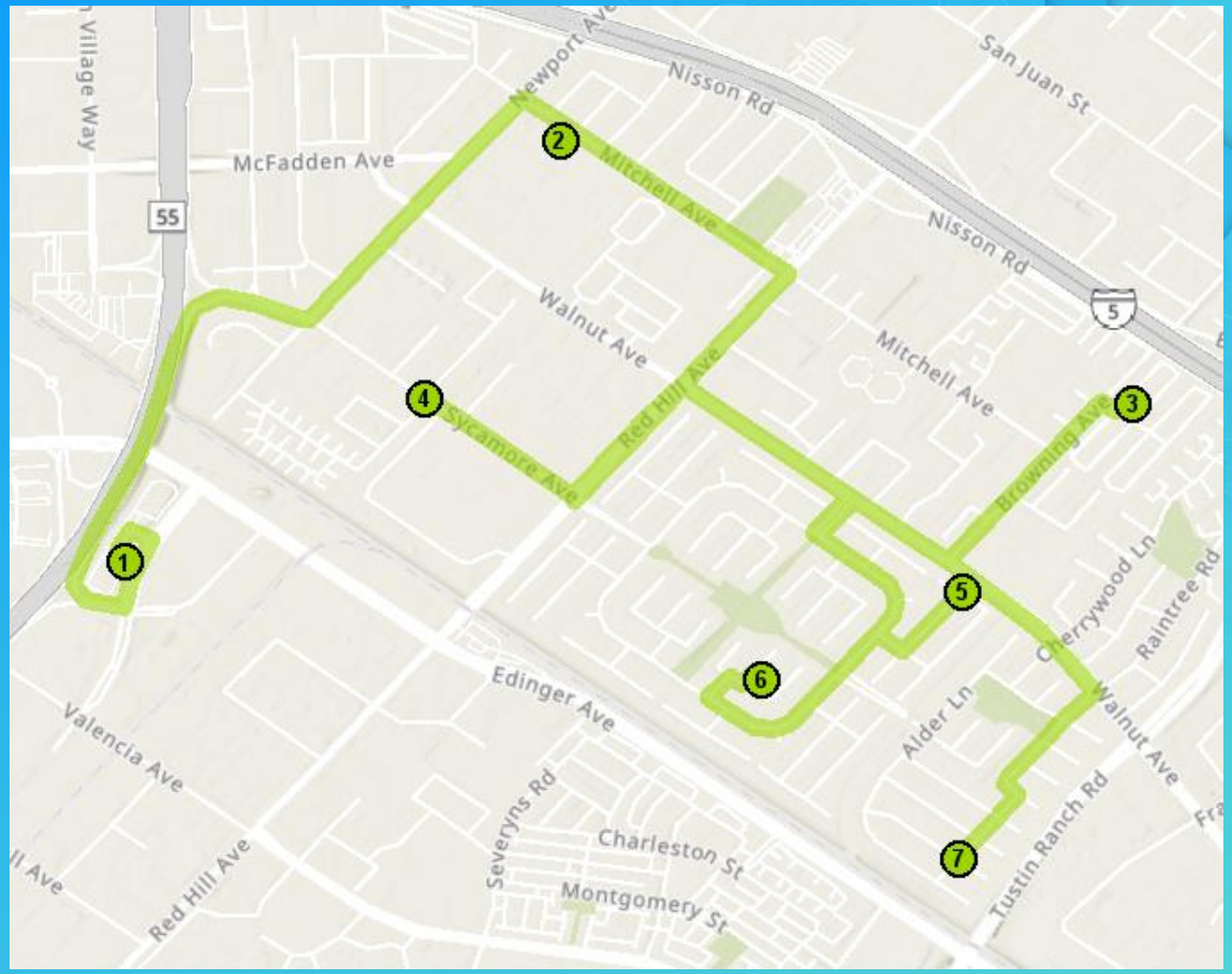
Doug Sterling, Frank Kish

Overview

- What are the network analysis tasks
- What functionality is present in network analysis tasks
- What functionality is common to network analysis tasks
- What are the network analysis objects, what do they do
- How things work
- What are the differences between online and local tasks

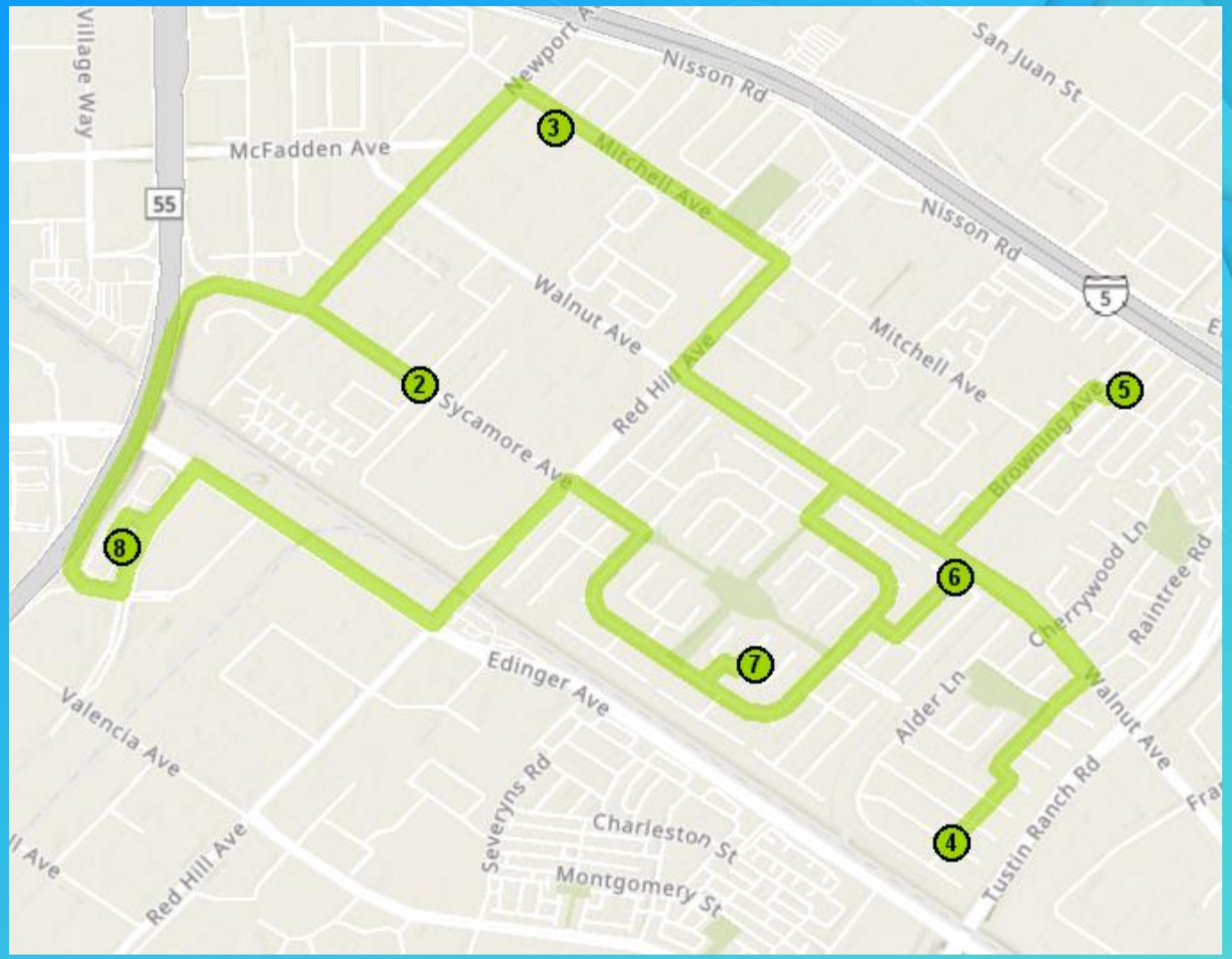
Route Task

- Can generate a route between a set of stops
- Can generate driving directions
- Common use cases:
  - Navigation
  - Route optimization



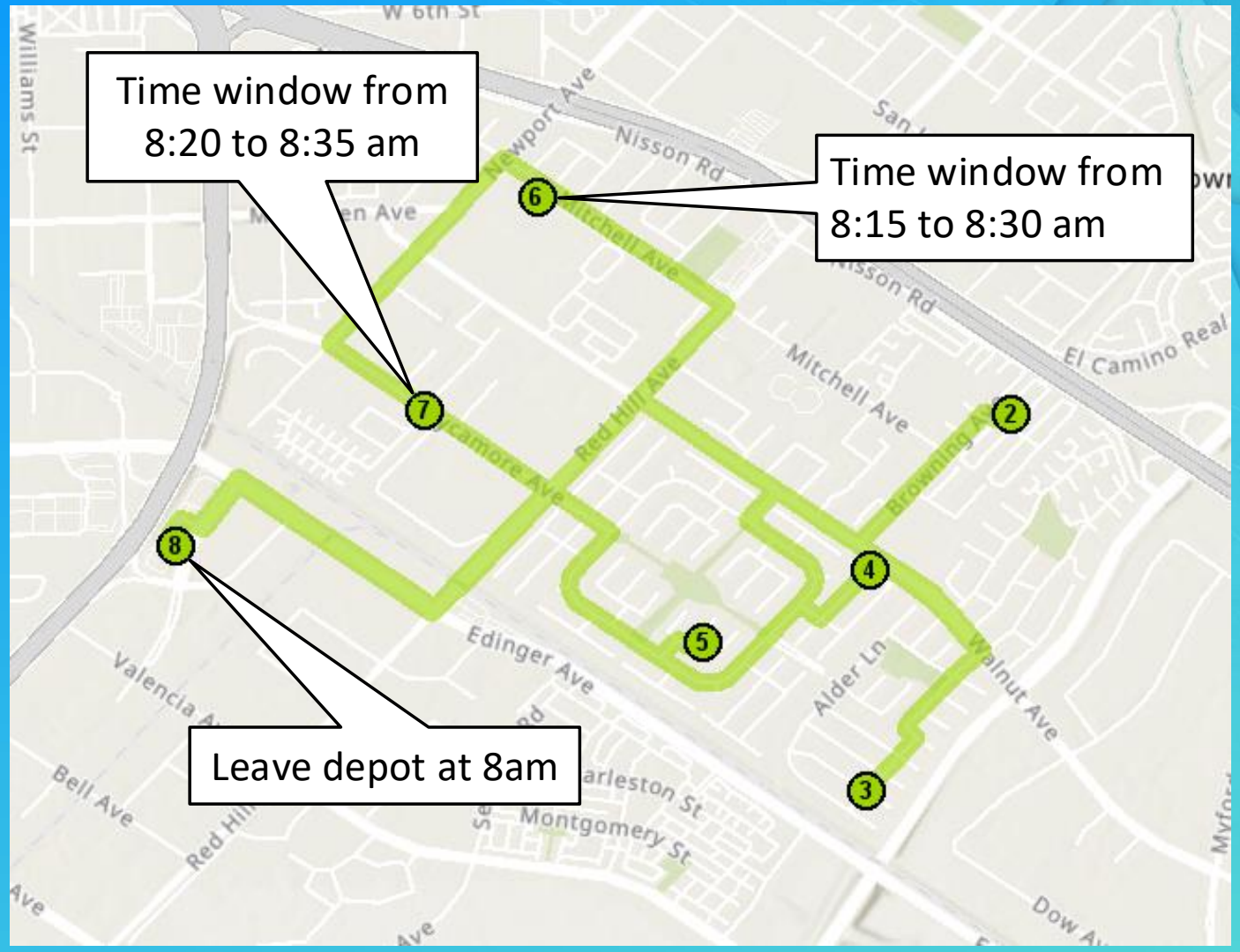
Route Task

- Can find an optimal route between stops
- Can preserve first or last stop for starting and ending at a depot



Route Task

- Route stops can have time windows that will be honored when finding a route
- Stops 6 and 7 need to be visited when returning to Stop 8 because of time windows



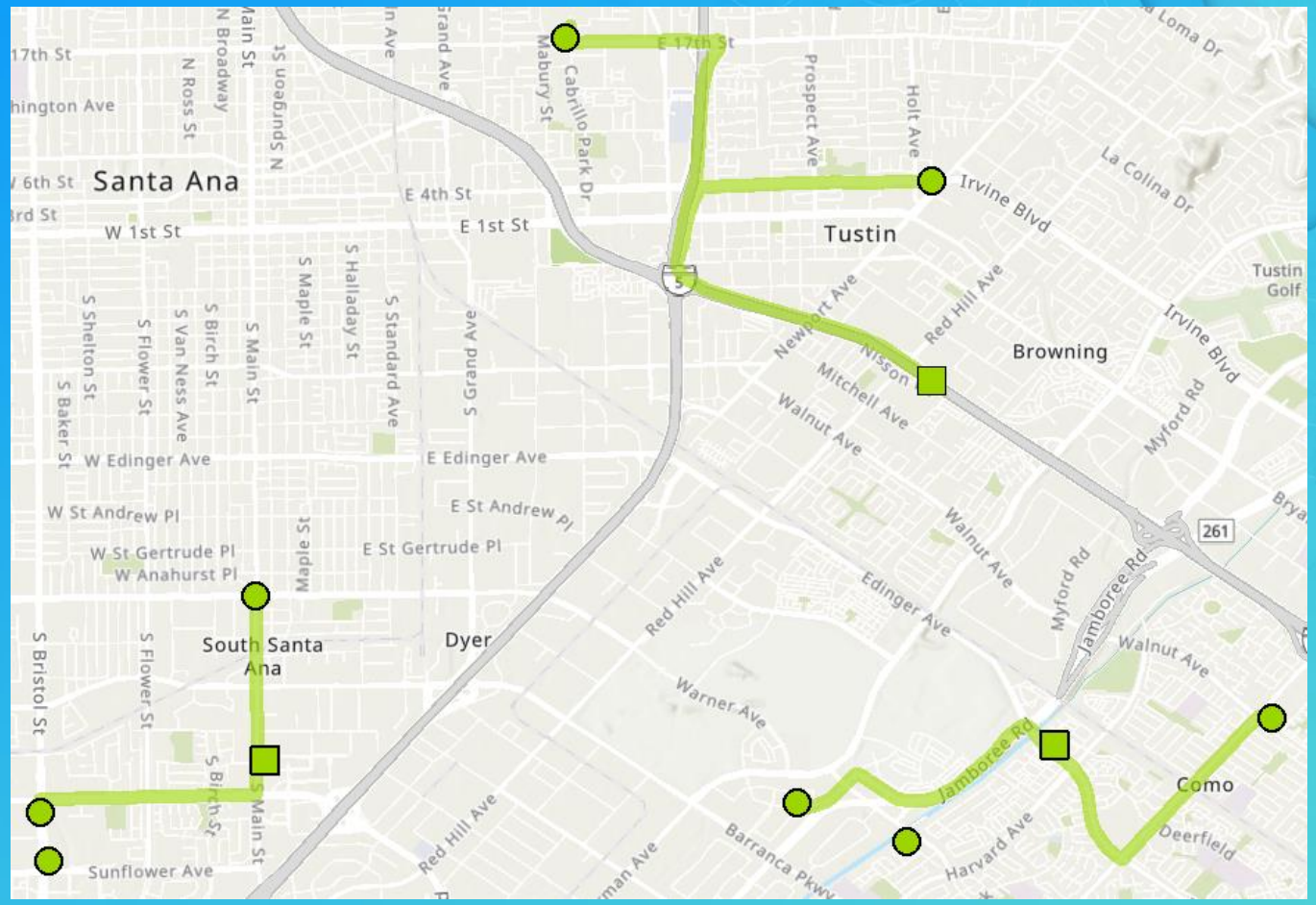
Route Task

Demo

- Demonstration of the route task

Closest Facility Task

- Finds Routes between incidents and facilities
- Common use cases:
  - Find closest business
  - Find five closest fire stations
  - Find closest ambulance

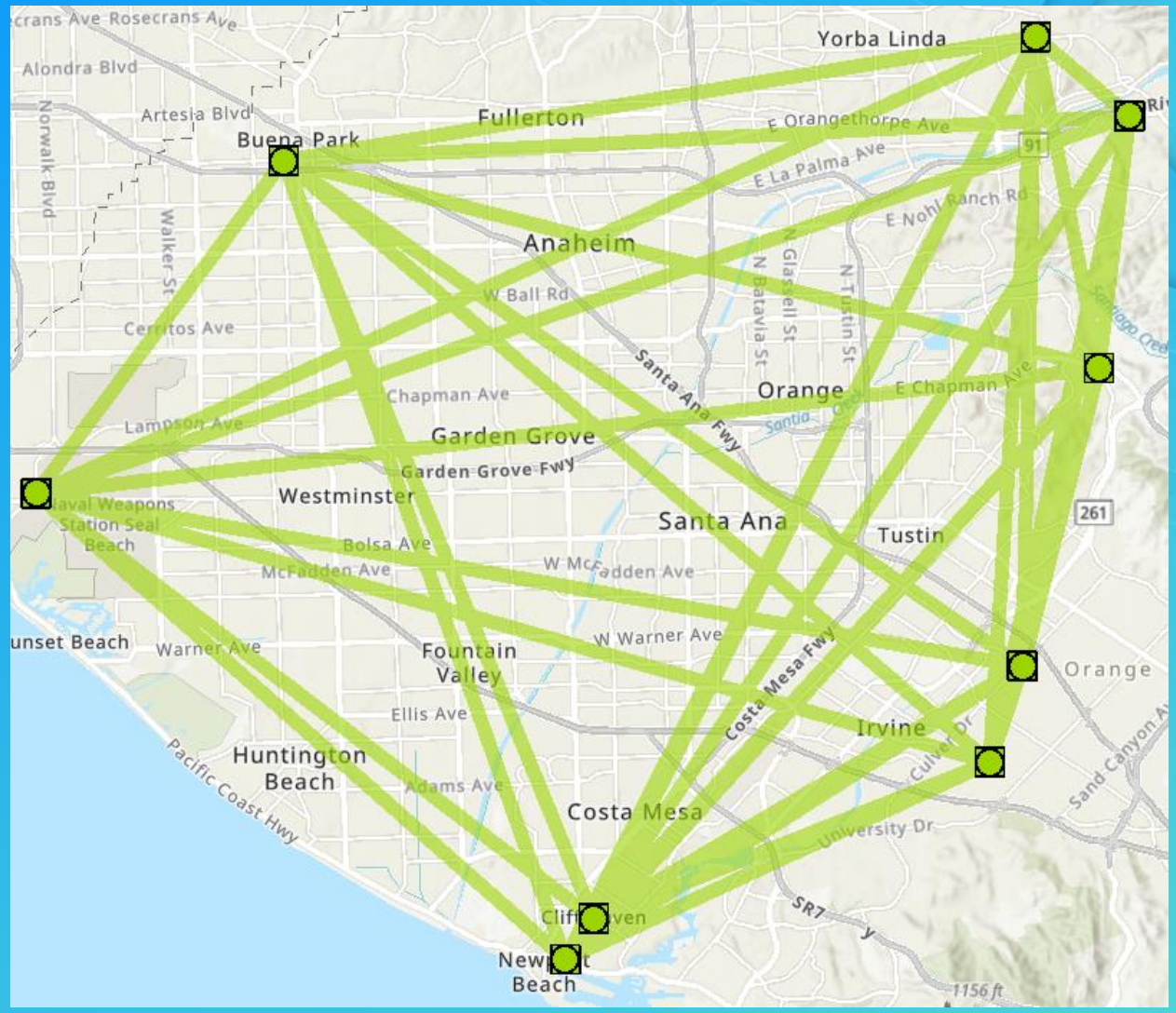


Closest Facility Task

- Routes are constrained by a number to find and also a cutoff
  - “Find up to three stores in no more than five minutes of driving time”
- Routes can go from incidents to facilities or from facilities to incidents
- Routes can arrive or depart at a time of day

Closest Facility Task

- Can be used to generate a matrix of route costs
- Use the Closest Facility Task if you need a matrix of costs
- Common use cases:
  - Generate a matrix of time and distance costs for some other optimization logic



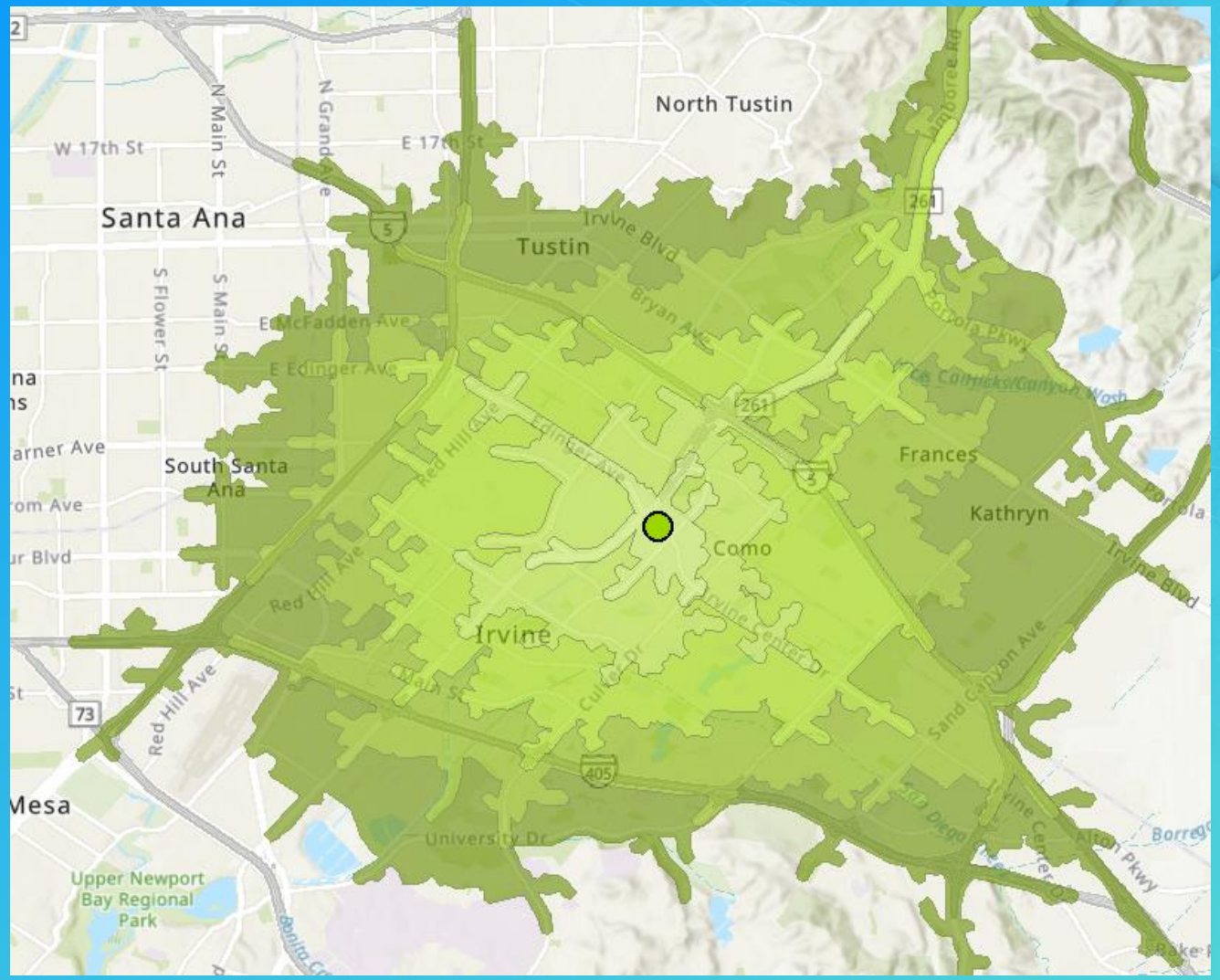
Closest Facility Task

Demo

- Demonstration of the closest facility task

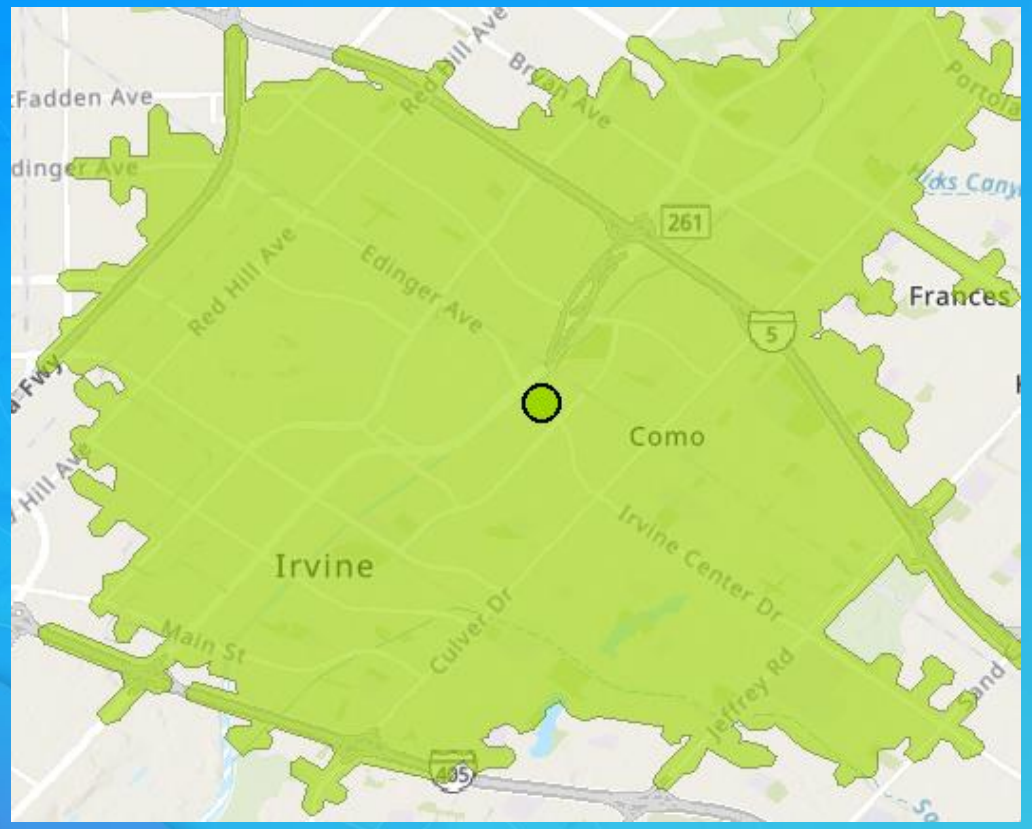
Service Area Task

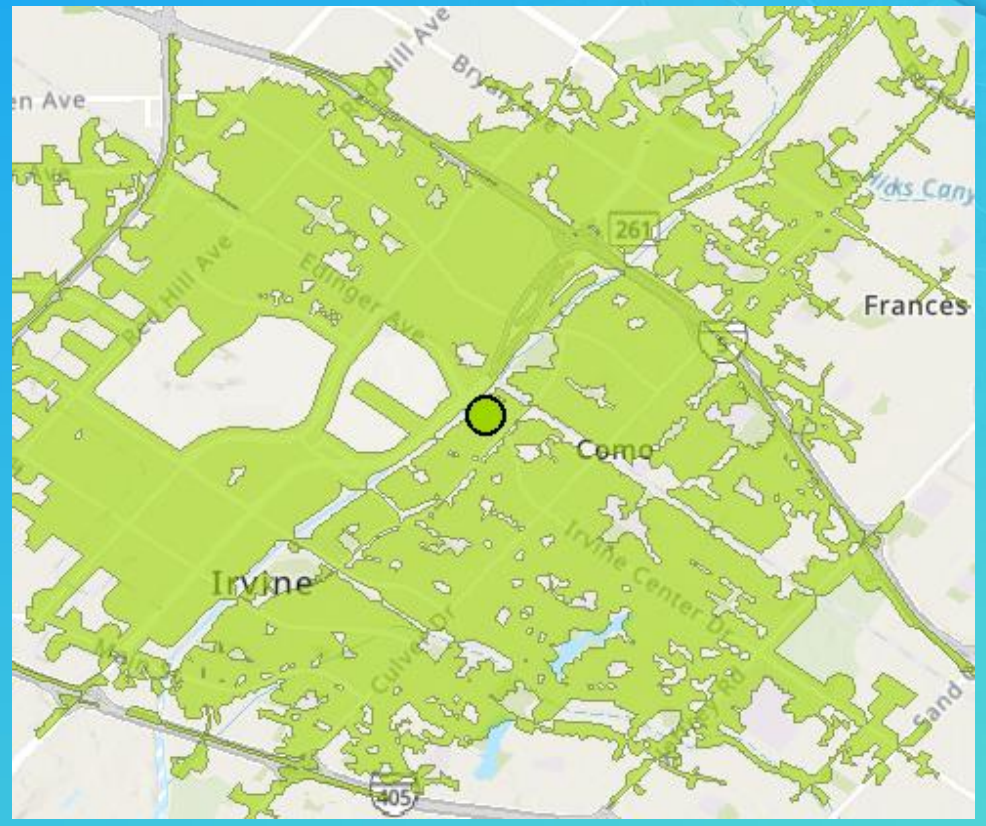
- Find the area that can be reached by a facility
- Here we find the area that is reached in ten minutes
- Common use cases:
  - Visualization of coverage
  - Load polygons in to spatial database and perform spatial queries instead of creating routes



Service Area Task

- Variable levels of detail coming in version 100.3
  - Can make holes for areas where edges are not traversed





Service Area Task

- Lines can also be output
- Lines accumulate multiple cost values
  - Time, distance, etc

Service Area Task

Demo

- Demonstration of the service area task

Common Task Functionality

Programming patterns

- All tasks work with online services as well as local data
- One programming pattern for both online services and local data
- Only task constructors know about online versus local

Common Task Functionality

Time

- All network analysis tasks are time aware
- Route task
  - Route has a start time
  - Stops have time windows for when the route should reach them
- Closest facility task
  - You can specify a departure or arrival time
- Service area task
  - You can specify a departure or arrival time

Common Task Functionality

Barriers

- All network analysis tasks support barriers
- Point barriers
- Polyline barriers
- Polygon barriers

Common Functionality

Demo

- Demonstration of barriers

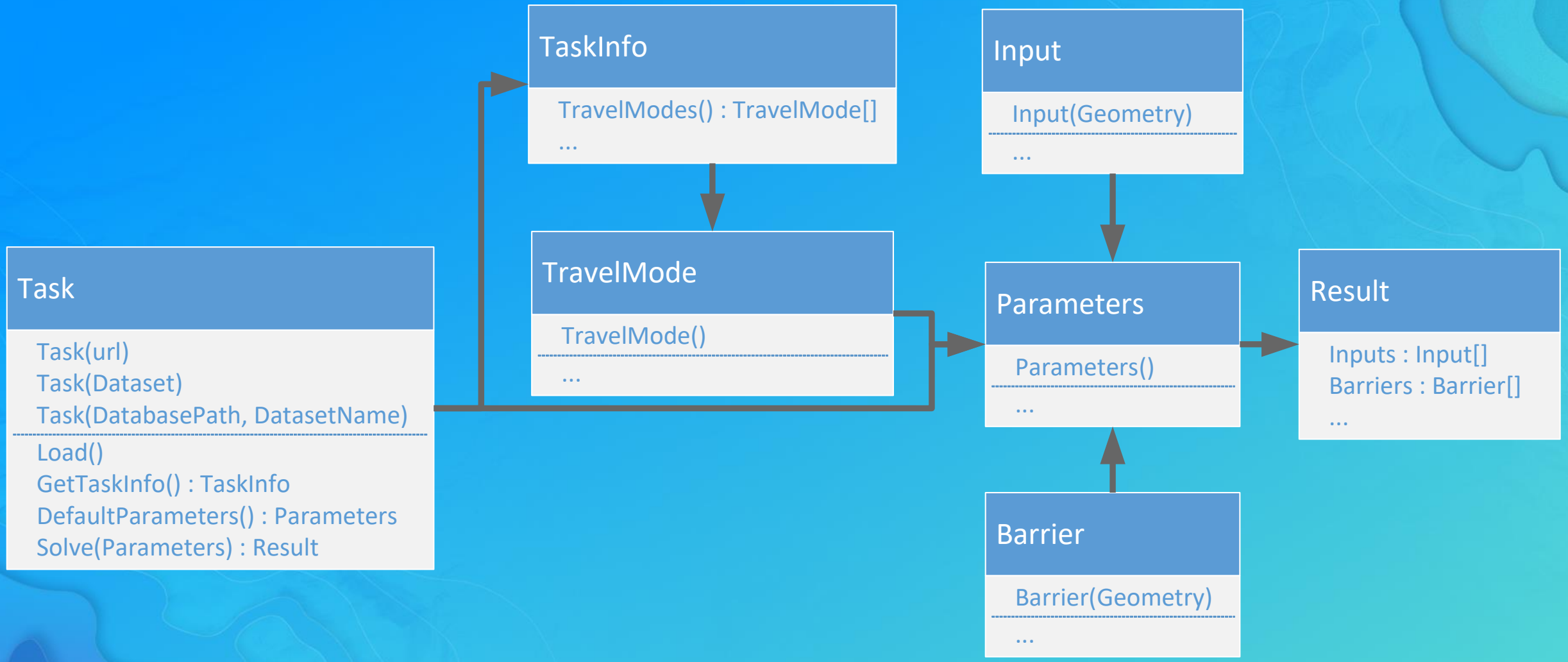
The Simplest Task Sample

Make a Service Area Polygon in C#

```
// Create and load task object
ServiceAreaTask serviceAreaTask = await ServiceAreaTask.CreateAsync(urlToService);

// Create default parameters object
var serviceAreaParams = await serviceAreaTask.CreateDefaultParametersAsync();

// Set Properties on parameters object
serviceAreaParams.SetFacilities(new [] {ServiceAreaFacility(MapPoint(-117.195,34.056, SpatialReferences.Wgs84))});

// Solve the service area
ServiceAreaResult serviceAreaResult = await serviceAreaTask.SolveServiceAreaAsync(serviceAreaParams);

// Get results
IEnumerable<ServiceAreaPolygon> serviceAreaPolygons = serviceAreaResult.GetResultPolygons(0);
```

Network Analyst task pattern

Start at the left, move to the right



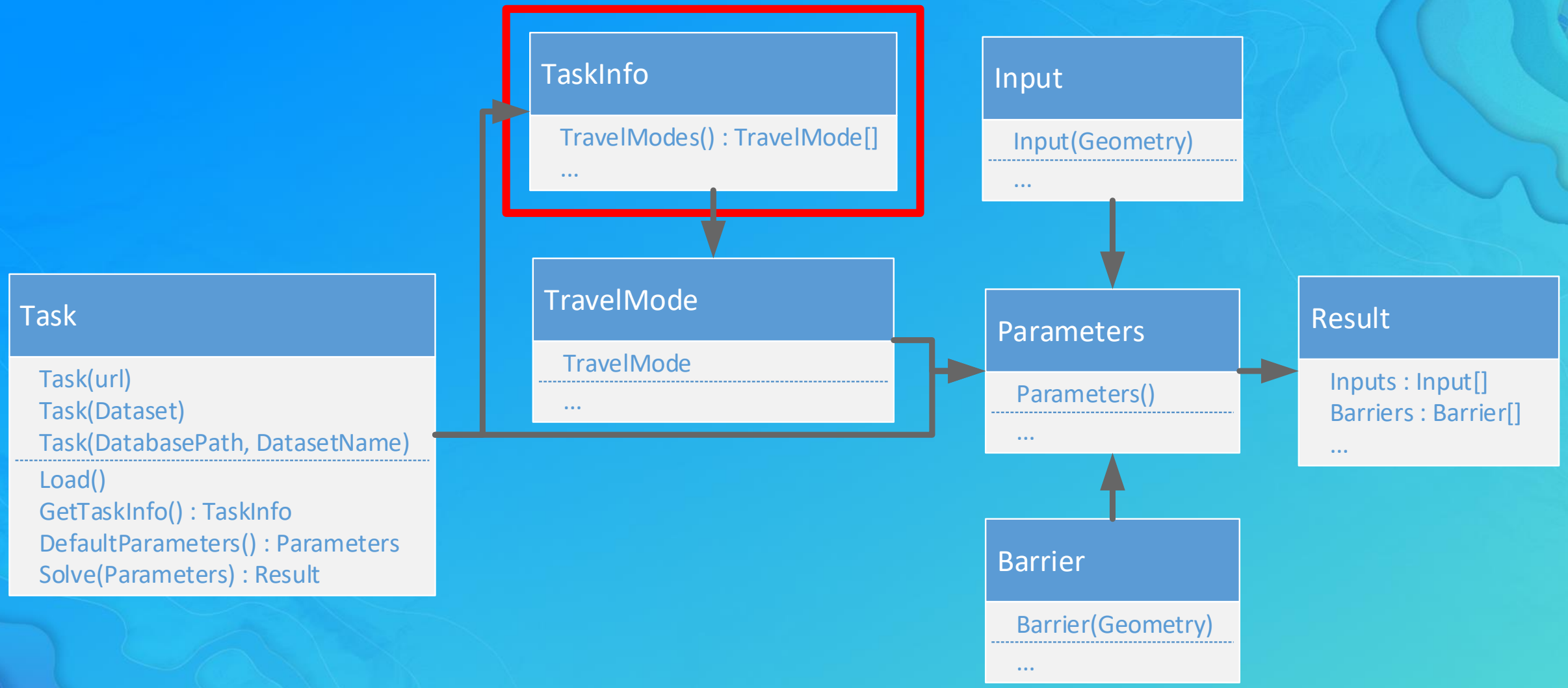
Tasks

Route Task, Closest Facility Task, Service Area Task

- Objects that perform network analysis operations
- Tasks can be created as needed or created and reused
- Tasks must be loaded before being used
  - Loading is asynchronous, SDKs handle this in construction functions
- Task solving takes place on an internal thread
- Tasks can solve multiple problems at once
  - Before we return flow of control to the calling application we clone your inputs
- Tasks are for online services or local data and this cannot be changed once they are loaded
  - Task constructors for SDKs apart from Qt force you to pick online/local in the constructor

Network Analyst task pattern

Task Info



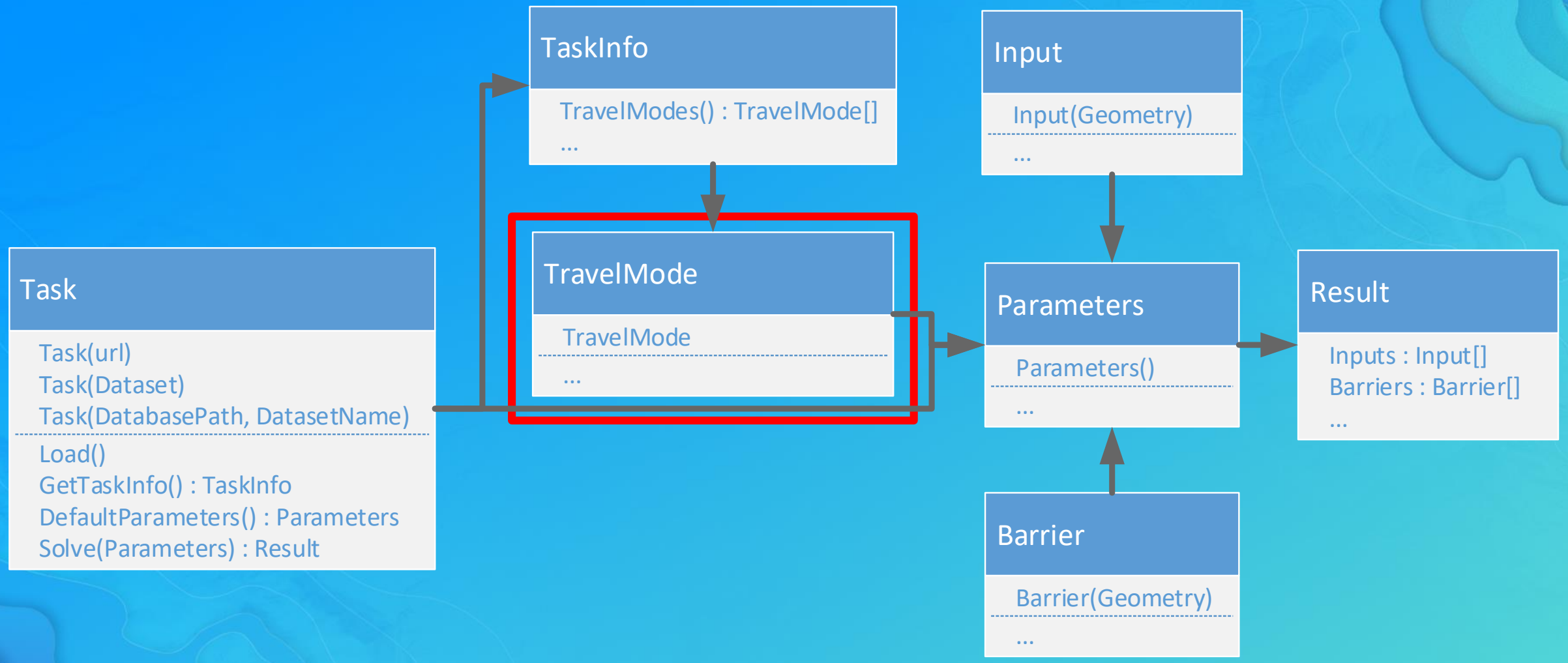
Task Info

Route Task Info, Closest Facility Task Info, Service Area Task Info

- Are obtained from loaded tasks, cannot be created
- The task info object describes the default state of a task
  - Describe the prebuilt travel modes that are supported by the service/dataset
  - Describe the available cost attributes and restriction attributes that are supported by the service/dataset
  - Describes default spatial reference of the task
  - Task specific defaults
    - Directions language, start time, travel from or to facilities, etc

Network Analyst task pattern

Travel Mode

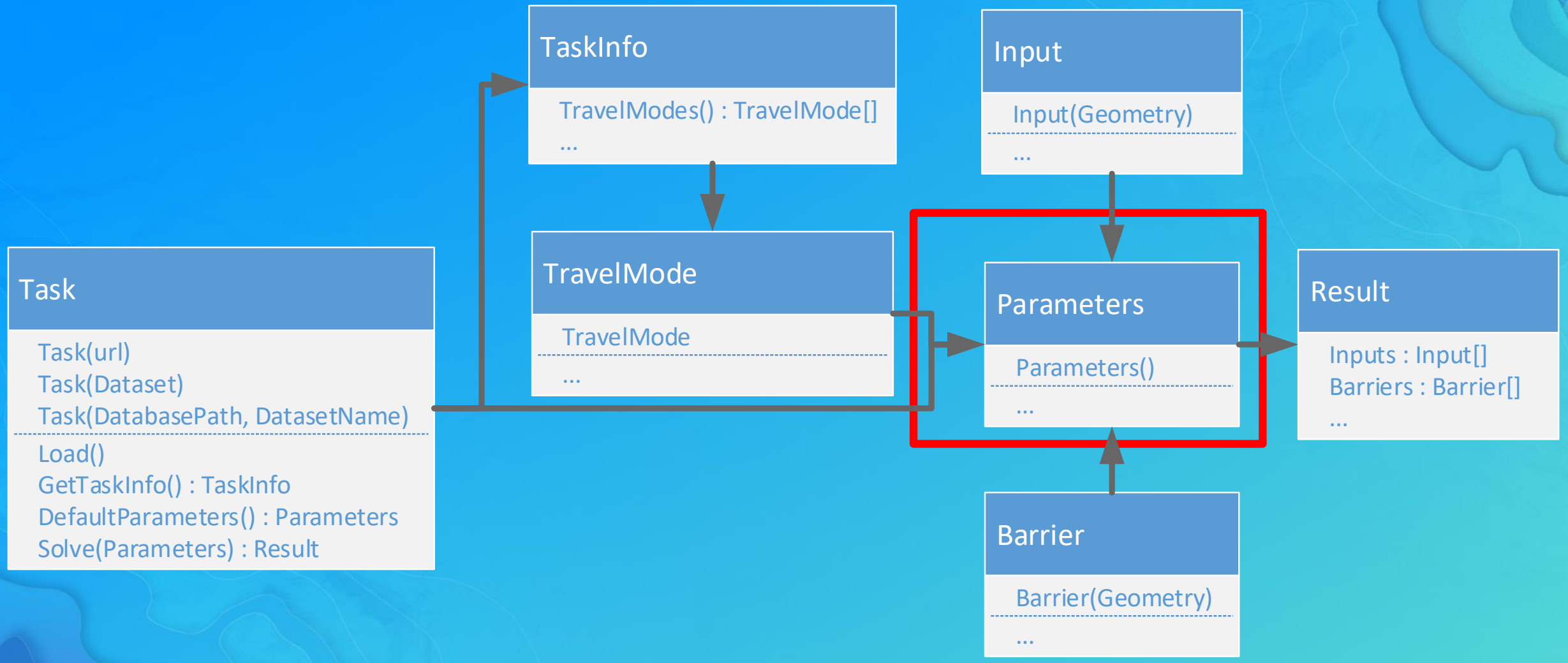


Travel Mode

- An object that represents a means of transportation
  - Driving a car, driving a truck, walking
  - Police car, fire truck
- Contains:
  - Impedance attribute use when traversing a transportation network
  - Time and distance attributes
  - Restriction attributes
- Describe where we can locate inputs
- Created in desktop, part of the network when it is exported to a MMPK
  - Best practice: create them in desktop and consume them in Runtime SDKs

Network Analyst task pattern

Parameters



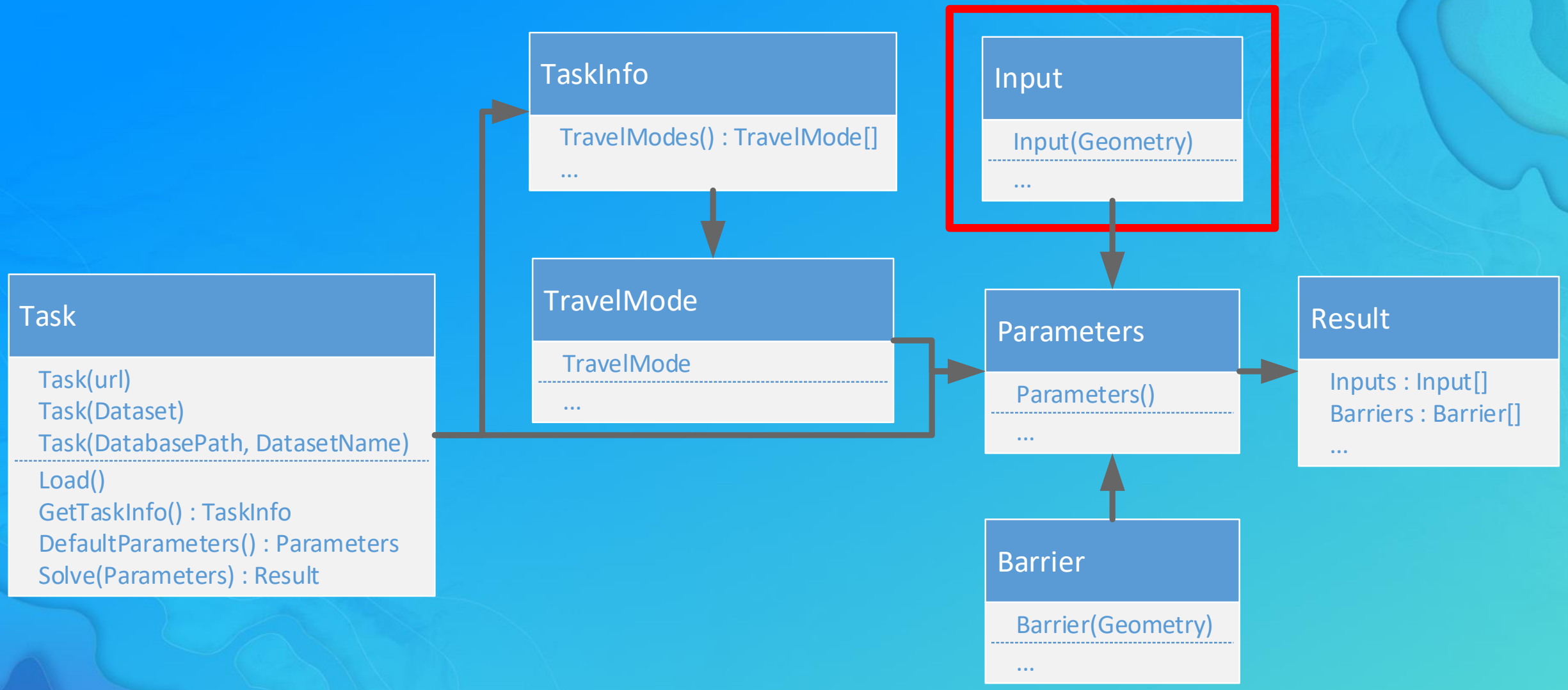
Parameters

Route Parameters, Closest Facility Parameters, Service Area Parameters

- Collection of task settings
  - Directions, route shape type, return barriers from solve, etc.
- Collection of task inputs
  - Barriers, Stops, Facilities, etc
  - Inputs may be objects, may be tables
- Used by task objects when a task is solved
- Client code can create Parameters objects from scratch
- Client code can obtain a default Parameter object from the task object

Network Analyst task pattern

Input



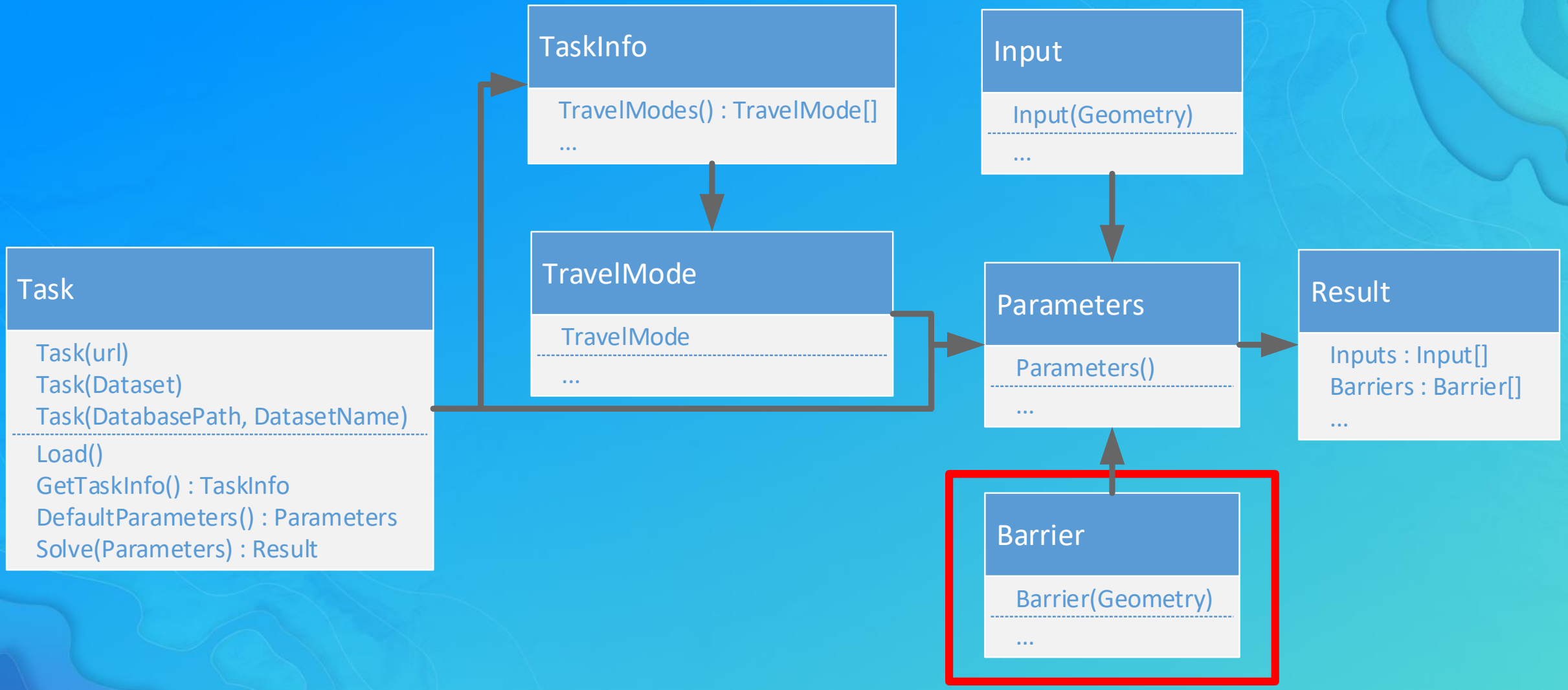
Inputs

Stops, Facilities, Incidents, Service Area Facilities, Point Barriers, Polyline Barriers, Polygon Barriers

- All inputs require a geometry in their constructors
- All can specify which side of the street they must arrive or depart from
- All can have added costs which are used to model the real world
  - Stop service time
  - Facility departure time (aka chute time)
- Inputs have properties specific to their task
- Some properties are populated by tasks
  - Stop sequence, arrival time, departure time, etc
  - Location status, network location

Network Analyst task pattern

Barrier



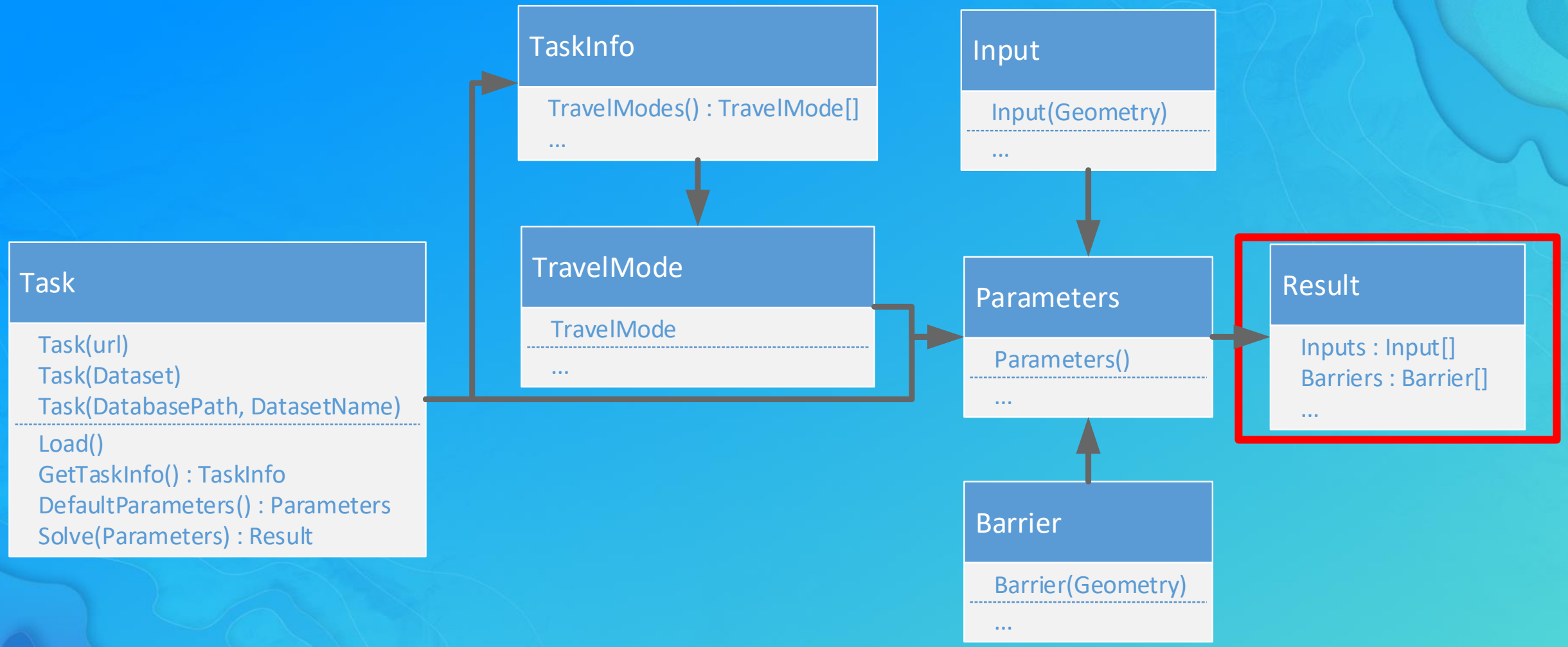
Barriers

Point Barrier, Polyline Barrier, Polygon Barrier

- All barriers require a geometry in their constructors
- Prevent inputs from locating on a part of the network that is blocked by a barrier
- Restrict traversal for a given point, polyline or polygon geometry
  - Optionally point barriers can add a cost to traverse a point
  - Optionally polyline and polygon barriers can scale the cost to traverse an area

Network Analyst task pattern

Result



Results

Route Result, Closest Facility Result, Service Area Result

- Tasks return result objects
- Result objects can contain multiple results
  - Route result can contain multiple routes
  - Service Area contains multiple polygons
  - Closest Facility contains routes
- Service area and Closest Facility task results are enumerated by facility index
  - For these tasks results always return copies of the facilities/incidents

How things work

Locating

- Online tasks: the service locates inputs as it sees fit
- Local tasks will always locate on routable elements in the network
  - Travel mode determines what is routable
- Local tasks will search up to twenty kilometers when locating inputs
  - If something is unlocated then most commonly the input geometry or SR is bad
  - Alternately everything within 20km is restricted due to malformed Travel Mode
  - Bad/incorrect where clauses on the parameters object
- Local tasks locate on edges unless incredibly close to junctions
  - You will locate on a junction if you are within the spatial precision of the projection

How things work

Locating continued

- Local tasks will inform you when the spatially closest network element was not routable
  - Will have a status of “not located on closest”
  - Apps can consume this value
- Curb approach can be flipped if the input is not reachable
  - Stop with curb approach right hand side but the stop is on the left side of a one way street
- Local task result inputs return the distance to their network location in meters

How things work

Locating barriers

- Barrier locating is not influenced by travel mode- it is purely spatial
- Point barriers are located on the nearest element
  - Like inputs can be located up twenty kilometers away
- Polyline barriers apply to all edges and junctions that intersect the polyline
- Polygon barriers apply to all edges and junctions that they contain and cross
- It is acceptable to have unlocated point barriers
- It is acceptable to have polyline, polygon barriers that do not intersect or contain any edges or junctions

How things work

Unlocated inputs

- For the route task if any stop can’t be located:
  - Local task: solve returns an error
  - Online task: solve succeeds if there are two located stops
    - Unlocated stops in result have a status of not located
- For closest facility or service area, if facilities or incidents can’t be located:
  - Local task: solve succeeds
    - Unlocated inputs in result have a status of not located
    - Unlocated inputs have no service area or closest facility routes
  - Online task: solve succeeds unless there is no result which throws an error

How things work

Loading inputs from Tables

- Parameters objects can accept inputs as ArcGIS feature table objects
- Table contents are loaded when the task is solved
- Input objects are internally created and can be returned in the results
- Online routing service with online table of inputs means that the ***service*** needs access to online table
- Input objects have properties populated by field mappings
  - Table schema identical to table schema of an analysis layer in desktop
  - *ONLY load tables with network location fields when the fields were populated against the same network as the MMPK or locations become erratic*

How things work

Memory

- All inputs and results are stored in memory
- NO paging to disk
- Something that works on a desktop can fail on a phone
  - Returning 500k+ service area lines
  - Returning 1000 closest facility routes

How things work

Local task results as inputs to local tasks

- Result objects can return copies of the input objects
- If you solve again then you can bypass spatial queries in locating if you use the inputs from the results
  - Polyline and polygon barriers internally track network elements that are restricted, spatial queries can be time consuming
- After a solve operation point inputs have a populated network location
  - Where exactly they are located
- If you change the travel mode or its settings then we will relocate inputs if the new travel mode interprets their locations as restricted
- Make sure you are solving on the same transportation network!

How things work

Licensing

- Local and online routing, online Closest Facility and online Service Area:
  - You need a lite license
- Local Closest Facility, Local Service Area:
  - You need a standard or advanced license
  - You also need an analysis license
- To use Streetmap Premium in any local task:
  - You need a Streetmap Premium license

How things work

Streetmap Premium data

- Requires its own license
- Built using HERE data
- High quality and reliability for use in network analyst, geocoding
- Many countries have historical traffic profiles
- Has commonly used Travel Modes already defined
- Available for individual states, groups of states or countries
- What Esri uses for our own online services

Differences between online and local tasks

- Error messages can be different
- Locating may be different for online tasks based on service configuration
- Online route task will return partial routes if some Stops can not be reached
- Local Service Area Polygon task does not support Generalized quality level
  - Local Service Area task upgrades your quality level to standard precision
- Local tasks use transportation networks directly, services use a published analysis layer which has more settings
- Default parameters objects can be different between local and online even if the underlying datasets are the same

Differences between online and local tasks

continued

- Online network datasets support Script evaluators, local data does not
- Route results may be different between online and local tasks
- Local tasks do not support live traffic
- Online tasks do not return input distance to the network
- Online tasks do not return when inputs are on the closest element that is not restricted

Questions?

esri

THE SCIENCE OF WHERE