1. Hello my name is Michael Tinker I work for the US Geological Survey, National Geospatial Technical Operations Center. We make the National Map, which a suite of products including the US Topo Maps. I work in the *Hydro Partner Support* group, supporting the *Hydro theme* of the National Map—the *National Hydrography Dataset*, or NHD. The NHD is a geospatial dataset model of all the surface water in the United States.
2. This presentation is about hydrographic addressing to the NHD, and in particular our web-based addressing tool, called HydroAdd.  
     
   HydroAdd tool allows you, in a web browser, to address your data to the NHD. That is, to make the features in your data coincident to the NHD and in the process, conflate, or spatially join some of the attributes of the NHD features to your data features.  
     
   *Addressing* your data allows you to track your own geospatial assets in relation to the NHD. *Addressed data* also allows for some special kinds of geospatial analysis. In this presentation I am going to show you a couple simple use cases of addressed data, and then I will do a slide demo the HydroAdd tool.  
     
   But first, we need to get into the weeds just a little bit. I need to define “hydrographic addressing” a bit more in detail.
3. First: A stretch of river in the NHD is called a reach. A reach in the NHD data has a unique 14 digit code—the reachcode.
4. Second: Reaches in the NHD have measures-- 0 is downstream and 100 upstream. No matter how short or how long a reach is it will always have 100 measures. You could think of the measure as percentage upstream.
5. So now that you have a reachcode and a measure for every single reach in the NHD you have a *network addressing system.* You can describe any point in the NHD uniquely with the reach code & measure. For example, this USGS stream gauge sits at this reachcode ending in 0421 and measure 19.8. It's easy to understand and to describe.  
     
   Now you could always use geographic addresses such as a latitude and a longitude, but when you're doing hydro addressing it is very useful to have a network address.   
     
   One of the main benefits of a hydro address is that it allows for *migration* of data addressed to earlier versions of the NHD, such as the 1-100,000-scale NHDPlus. More on that in the demo.
6. There are other terms that you will hear for addressing such as; linear referencing, indexing, linking-- these all mean same thing. The term “Event” describes the actual addressed data. For example, we talk about point events and line events.Okay, now are ready to illustrate a few use cases of addressed data.
7. Address data allows modeling and analysis. In this map, the width of the flowlines depends on its flow, in cubic feet per second. This model is possible *because* we addressed USGS stream gauges. Many of these stream gauges have many decades of flow and velocity data. They also have associated upstream drainage areas. We can use fancy regression equations to calculate flow and velocity on ungauged streams. In fact, we have modeled flow and velocity for every single flowline in the NHDPlus HR. This is a powerful set of attributes. This allows you to do many kinds of modeling analysis, such as time of travel studies for point source pollutants.
8. Address data is useful for tracking assets.   
     
   The National Wild and Scenic Rivers System was created by Congress in 1968 to preserve certain rivers with outstanding natural, cultural, or recreational values. Obviously wild and scenic rivers are national assets.  
     
   USGS puts Wild and Scenic Rivers on the US Topo product. This little section of the US Topo shows the Yukon Charley Rivers National Preserve in eastern Alaska.   
     
   As you can see, address data doesn't have to be points--it can be lines too. An addressed line event has a start point and end point. The intervening geometry is exactly coincident with the underlying NHD flow features.
9. Addressed data is useful for reporting.   
     
   Section 303(d) of the 1972 CWA, requires states to identify to the EPA waters where current pollution control technologies alone cannot meet the water quality standards set for that waterbody.   
     
   This little map shows the 303d-listed streams in Central California. Every stream on this map is addressed to the NHD with the start reachode/measure, end reachcode/ measure, and the intervening geometry is exactly aligned with the underlying NHD reaches.
10. So the last point I'd like to make is that you can address anything to the NHD: point source pollution, biological sampling sites, invasive species in riparian zones, recreational stretches of rivers. You can address pretty much any hydro observation to the NHD. The question then becomes: “*What is your business need? Which brings us to HydroAdd.*
11. HydroAdd allows you
    1. To address your data to the NHD. And, to edit, maintain, migrate, and synchronize it.
    2. It works in a web browser, so teams can collaborate on addressing projects
    3. HydroAdd has national extent, so no limitations, no downloads of staged data. You can work anywhere in the US that has NHD.
    4. It's very easy to use UI, its flexible, and yet powerful.

*NOTE: You must have AGOL to use HydroAdd. AGOL is the hosting service. We’ll talk about that in just a moment.*

1. Here is the basic workflow for working with HydroAdd.   
   I’ll show you a little bit of each of these steps as we go through the slide demo.
2. **Part 1:** Prepare you Data.
3. Download the schema from HydroAdd tool. The download is a geodatabse with two empty feature classes.
4. After you download the schema, prep your *data* in ArcGIS Pro.   
   I always put the downloaded gdb in my ArcGIS Pro Project, so that way I always have an untouched copy. When I am ready to make a new *event* layer, I just drag one of the empty feature classes from the downloaded gdb to my ArcPro *project gdb* and rename it.
5. Note that HydroAdd is agnostic to extra fields. So as long as you have the *core* HydroAdd schema here, you can add as many other extra fields as you want.  
   After you add any extra fields…append your data into this empty schema.
6. **Part 2:** Now you are ready to publish you map to AGOL.
7. Prep your map. Get the layers and symbology the way you want. This is how it will look in HydroAdd.   
   Then publish to AGOL.
8. When you publish your map to AGOL, you must share it into an AGOL group. This is how AGOL manages sharing and permissions.  
   As you can see, I already shared some maps to this group. And I already have six people in this group who will be *collaborating* with me on this project.
9. **Part 3:** is the easy part: Add the AGOL service to HydroAdd
10. Copy web service URL from AGOL…
11. Back at HydroAdd now, **click** the *add new service* button…
12. Paste the URL that you just copied from AGOL…
13. …and the data instantly appears in the HydroAdd map.  
    All the services you’ve ever **added** are listed in your profile, in the near-left pane, and also in the *services widget*, top right in the map.
14. In **Part 4**, you are ready to start using HydroAdd.  
    So let’s take a look at some of these functions.
15. So here it looks like *make a new point event*.   
    To begin, select the service layer that you want to edit.   
    You see I've selected the service layer up top in the editing pane….  
    And you can see my editing crosshair in the map on an NHD flowline…
16. Click once. A red halo appears in the map where you clicked.   
    Also notice in the *editing pane*… there is now a *reachcode* and a *measure*. That’s the *hydro address*!  
     Now I can't save anything until I have a unique identifier, that’s what we call the *Source ID*
17. But once I enter a SourceID, top right of the editing pane…the save icon lights up, and I can save.
18. The point appears in the map and the Source ID label is also displayed in the map.   
    At the same time, the editing pane clears…you are ready to make the *next* new point event.
19. Making new *line events* is similar. Click once on a flow line to make the *start* point…
20. …click a second time to make the end point; HydroAdd traces the flow lines between the points.   
    The line trace is red. Both start point and endpoint have hydro addresses: ReachCodes and measures.
21. As with points you must have a SourceID in order to save. Enter the SourceID to save…
22. And now you can see the line event with SourceID label in the map.
23. With line events, you have the option to “*Use Full Reach*” that is, to make line events that span the entire ReachCode from measure 100 to measure 0 with just one click….To do this turn on *Use Full Reach*
24. Position the editing crosshair….Click once
25. ...the red trace appears in the map…Enter a Source ID to save…
26. …and the line event appears in the map, spanning the entire ReachCode.
27. One more thing to know about making new line events is that HydroAdd always splits lines at ReachCodes. So if your *start* and *end* points *span* multiple ReachCodes, then HydroAdd splits that into multiple features at the ReachCodes.
    1. So you can see my start point in the map…
28. …and there’s the end point and the trace in red
    1. Notice there are dots where the ReachCodes change
29. Save it. My SourceID is “U2”…
30. …and there’s the final result. I just made four line events with the same SourceID.
31. You can prove this in the Search/Select tab. Just search your service for SourceID = “U2”. You can see there are four results.
32. Okay lets talk about Editing.  
    To edit a *point*, simply click directly on any point event in the map …
33. …and the Editing Pane opens. Notice this point already has a ReachCode. *But* its need to be synchronized. It’s just floating in white space nearby this confluence. I'm going to update the point to sit on the tributary rather than the main stem.
34. There’s the red halo. In the editing pane the ReachCode is updated to the correct reach. Also notice top of the editing pane, the Source ID has already been populated, so I can save.
35. …and the point is now updated.
36. *Line editing* is similar. This line event here in orange was originally addressed to the *medium* res NHDPlusV2. You can see that ***hi res*** NHD flowline in blue behind it has a greater sinuosity.   
    As with points, click the line event in the map to select it…
37. …and the editing pane opens. The line event is highlighted in cyan in the map.   
    In the editing pane, you can see it already has *start point*, an *end point* *and a SourceID*.
38. So to update this line event, I’ll click the new start point …
39. …and then the new end point. There’s the trace in red.
40. It's already got a source ID so I can save...
41. …and the line is updated.
42. Now Creating and editing features one at a time is all very well, but the *real* benefit of using HydroAdd is that it has an *editing queue.*
    1. You can see the name of an editing queue I've made in the *far-left* pane called “303d lines Shamokin Mtns”.
    2. In the *near-left* pane you can see the editing queue itself.
    3. It has 40 features. You can see the number of features in your queue and the number of rows per page at the bottom of the queue.
    4. In the map, the features in a queue are always highlighted in cyan.
43. Mouse over a row in the queue to highlight that feature in the map in yellow.
44. Click a row in the queue to *zoom* to the feature in the map.   
    BTW, The blue row in the queue is always most recently clicked row.
45. When you mouse over a row, the edit and delete buttons appear in the row.   
    Click the edit button…
46. … and the editing pane appears. I turned on “use full reach” function to update this line.
47. After saving, you are returned to your spot in the queue. Notice how the row now has a thumbs up. It's approved! HydroAdd assumes that because I went to the trouble to edit this line that its in the right place, and therefore approved. You can see there are 2 approved features in this queue, but rest of the features in the queue are thumbs down.
48. The checkboxes at the top of the editing queue allow you to view *approved*, *unapproved*, or *both*.
49. A typical workflow would be to view only the *unapproved* features, then whittle away at the queue until you’ve *approved all* the features. Just put your headphones on and go to work. A seasoned editor can go through 500 features in a day.  
    We covered *Making New Events, Editing,* and the *Editing Queue*.
50. I want to show you now the *Batch QC function* for *large data sets*. Say you have 10,000 point events. No one has time to manually QC 10,000 points. Instead, you would build a *Batch QC* job.

You can see some of Batch QC jobs I've already run listed in the near-left pane.

1. You can run Batch QC against an existing *editing* *queue* …
2. or you can select an entire *service*.

Be careful when you run Batch QC against an entire service. A whole service might have very many features. You don’t want to accidentally kick off a Batch QC against a service that has 64,000 events! That would take a *very long time* to run! We have not *yet* developed a way to *stop* a Batch QC once it’s started.

1. The most important parameters of **Points** Batch QC: the S*nap Tolerance* and *QC tolerance*.
2. The *snap tolerance* is just a buffer around your point. If there are flowlines in your snap tolerance, the point will *always* snap to the nearest flowline within that buffer.
3. The *QC tolerance* sets the *auto approve* threshold. In this example, the snap tolerance is 100 meters. A point will snap to the nearest flowline in the 100 meter tolerance *no matter what*. But the *QC tolerance* is set to 50 meters…so, if the point moves *farther* than 50 meters, it's not automatically approved.
   1. If the snap tolerance is set high, for example, 100 meters or more, and there are multiple flowlines in the snap tolerance, your point could snap to the wrong flowline. Setting the QC Tolerance to be just a bit *less than* the Snap Tolerance helps catch instances where a point moves very far and may have snapped to the wrong flowline. Again, if the point moves *outside* the QC tolerance, then its not approved, and you would need to go back and manually QC the unapproved features later.
   2. BTW, Batch QC results are greatly improved if your data already have a ReachCode or a G N I S NAME. So, if your point event already has a ReachCode, and there is a flowline with matching ReachCode in your snap tolerance, Batch QC will snap to that flowline, and skip the flowlines that don’t match. Likewise for a matching GNIS NAME.
   3. ~~If your data already have~~ *~~both~~* ~~ReachCode~~ *~~and~~* ~~NAME, ReachCode always takes priority. If you have neither, Batch QC simply snaps to the~~ *~~nearest~~* ~~flowline in the snap tolerance. And that is when~~ *~~QC Tolerance~~* ~~is most important—in cases where your data~~ *~~do not have a match to~~* ~~an existing ReachCodes or NAME.~~
4. So here are the results of that *points* Batch QC. The near-left pane shows the Batch QC queue. The *yellow* line you see in the map indicates the snap distance. This point has a thumbs up because it’s within *both* Snap tolerance and QC tolerance.
5. Here’s another one. In *this* case, the point snapped because it's ***in*** the Snap Tolerance, but it's *not approved* because it moved farther than 50 meters. So, thumbs down!
6. HydroAdd has Batch QC for lines too. It functions in a similar way to *points* Batch QC, but it’s just a bit more complex.
   1. So lets discuss Lines Batch QC in detail.
   2. You can see the *lines QC Jobs pane* lists your previous Batch QC jobs, same as points QC.
   3. In the map here, there are two line events: one is red, one is blue.
7. Looking at these two line events from a queue, we can highlight them.
   1. the SourceIDs are “one”…
8. …and SourceID “two”.
   1. Imagine these two line events were originally addressed to the *medium-res NHDPlus*, so they already have ReachCodes.
   2. But looking at these events in the map, you can see these are not aligned with the *high-res* NHD.
   3. They need to be updated, or migrated. So, lets kick off a new Batch QC…
9. As with points Batch QC, you can run Lines QC against a queue…
10. …or an entire service. I’m just going to use a queue.
11. I’m giving it a name here: “Two Lines 100m”
12. I enter the Snap Tolerance: 100 meters.
    1. The Snap Tolerance for *lines* refers to *both* the start *and* end points. *Both* the *start and end points* must be in Snap Tolerance, or the e*ntire line* is ignored as *out of snap tolerance*. But if both start *and* end points are in the snap tolerance, then Lines Batch QC will snap the two points to the nearest flowlines and then *trace* the flowline to draw a new line event between the two points.
13. Then enter the QC Tolerance: I’m also setting this to 100 meters.
    1. The QC Tolerance uses similar logic as Points QC, but *both* start and end points must be in QC Tolerance for the entire line to be approved. For example, if *only* the start point is outside QC tolerance then the *entire line* is out of QC Tolerance.
14. Lines Batch QC has an additional QC parameter, the *Length Change Tolerance.* 
    1. The *Snap* and *QC Tolerances* are in meters, but the *Length Change Tolerance* is a *percentage*. You’re setting a QC threshold on the *percent changed* of the *new line length* from the *original line length.* This is because when a Batch QC draws new line events, the *new* line is almost always a different length than the original line. It’s usually shorter. *~~But sometimes longer; it just depends on the sinuosity of the lines.~~*
    2. The ***length change tolerance*** is the *amount of length change* you are *willing to accept* for a new line to be auto approved.
    3. I am going to show you the results now so this will make more sense.
15. Here are the results of this Lines Batch QC job
    1. The Batch QC *queue* is the near left pane.
16. Turning off the NHD service just for a moment, so the features are easier to see.
    1. The *original* features are always cyan in the map.
    2. *Results, or new features,* are always red.
17. Notice these odd, short, little segments here…
18. …and here. These are the kind of thing that *Length Change Tolerance* is designed to catch
19. Recall that were two original line events: The SourceID are: “one”…
20. …and there is SourceID “two”.
21. But look again at SourceID “one” in the Batch QC queue.Notice the original has 3 results under it:
22. First result is *not approved*, you can see the “thumbs down” in the row of the queue.
23. Second result *IS* approved, thumbs up.
24. And third result is also *not approved*.
    1. So why did we get three results on SourceID “one”?
    2. The first part of the answer is that HydroAdd always splits lines at ReachCodes. These results span three ReachCodes, so we got three features.
    3. But the second part of the answer is that Batch QC didn’t find any matching ReachCodes in the Snap Tolerance of 100 m, so the end points just snapped to the nearest flowlines, which in this case are tributaries to the main stem.
25. Let’s zoom in to one of the end points. The original line is cyan, the Batch QC result is red. I drew an arrow where the end point of the original line snapped to the nearest flowline—on a tributary to the man stem.
26. And zooming in to the *other* end point….the original line is highlighted yellow this time….but the same thing happened on this side. The *end point* of the original snapped to the nearest flowline—a tributary to the main stem.
    1. This *resulting* segment was *not approved* because its too short. It’s outside the *length change tolerance*
27. If you want to see *how far* outside tolerance it is, click the details button.
28. This is the *details view*. Its zooms you to the result feature and highlights it in yellow. As always, the original is cyan.
    1. Details view displays all kinds of interesting things, such as how far the end points moved, if the ReachCodes were matching or not, as well as **the actual** *length change percentage*.
    2. In this case, Details View is telling us the *length change percentage* was 83.6%, which is way above the 50% tolerance we set. So that’s why this little segment was not approved.
29. The approved feature, in the middle, is exactly in the right place.
    1. So our length change tolerance of 50% did what it was supposed to do—it found the short little segments that we didn’t want.
    2. And when we go to examine all the *UNapproved* items during the *manual QC pass*, we would just *delete* these unapproved short little segments.
30. Another thing we *could* have done was to run the Batch QC again, but to *increase* the snap tolerance to 200 meters.
    1. Remember, our original lines already had ReachCodes, because they were addressed to the medium-red NHDPlus.
    2. When we run this same data again at 200 m snap tolerance, Batch QC finds the matching ReachCodes.
    3. And you can see the results look better here—those little segments are gone.
31. And that brings me to my final slide.
    1. HydroAdd is a tool for QC’ing your addressed data, but it’s also a *migration* tool because Batch QC can use an existing ReachCode or NAME. If you have 10,000 points events that you’ve already addressed to the NHDPlusV2, you can **migrate** those *medium-res* events to matching ReachCodes on the *hi-res* NHD.
32. Okay, quick summary:
    1. Because HydroAdd is web-based tool, project collaboration is easy.
    2. You can use HydroAdd to migrate or sync existing events, if they have ReachCodes or G N I S NAME
    3. It has seamless national extent of WBD/NHD  
       No need to checkout or download staged products from USGS
    4. It has a simple, but powerful, UI
    5. ArcGIS Online as hosting platform is easy to use
    6. In general, web-based tools make it easier for our developers to roll-out upgrades
33. But there are some limitations you should be aware of.
34. Web-based tools are inherently slower. Batch QC jobs can take a long time. Large jobs with many thousands of features can take hours to run.
35. Also, users Batch QC jobs are staged. Our addressing web API is a serial processor, so if there are multiple Batch QC jobs, they are staged to run in succession.
36. Finally, AGOL as a hosting platform has some limitations. I love AGOL—it’s very easy to make editable web feature services and set group permissions. But publishing Batch QC results to AGOL is very slow. And while it's easy to develop against AGOL, we are still having to chase the most recent upgrades and changes that Esri makes to AGOL
37. And of course, AGOL is not free.