MANAGE YOUR OUTSIDE PLANT WITH NETTERRAIN OSP

JAN DURNHOFER
GRAPHICAL NETWORKS

TABLE OF CONTENTS

04 INTRODUCTION

11 MANHOLES, CONDUITS, DUCTS

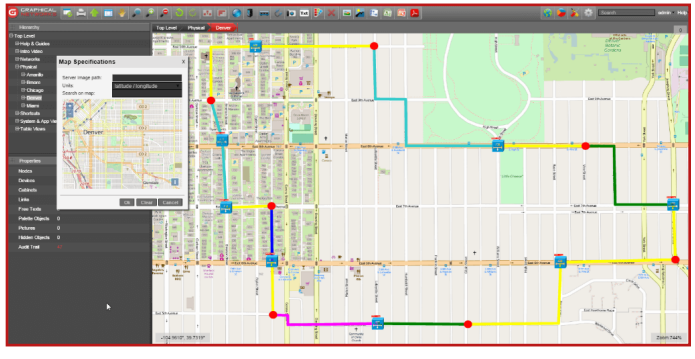
19 MAPPING FIBER STRANDS IN A SPLICE BOX

29 TRACING CIRCUITS

39 OUTSIDE PLANT SOFTWARE OPTIONS: COMPARISON/COST

INTRODUCTION

Need an easier way to document and map your fiber backbone? netTerrain OSP, our outside plant management platform, is used by our customers to manage their inside and outside plants — from fiber splices to manholes to GIS-enabled street views (with 6-decimal precision) all the way to fiber patch panels inside buildings.

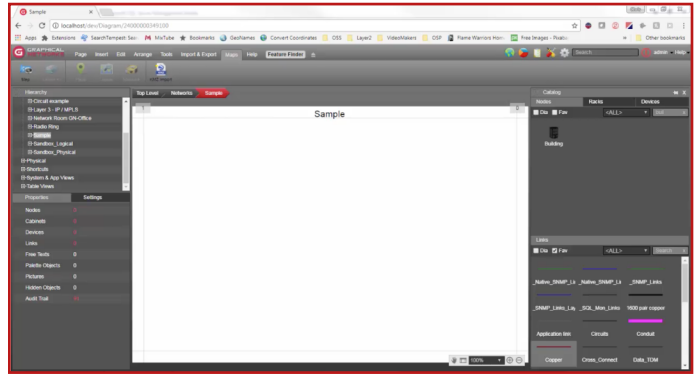


This eBook takes a look at what you can map with our fiber plant software and how easily you can work with it.

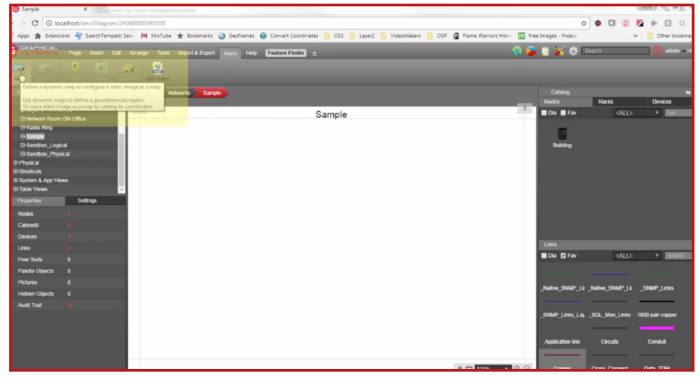
In this introduction to the eBook, we will:

-Create a dynamic map from scratch — in netTerrain, using OpenStreetMap (the tool we use for rendering dynamic maps)

-Show you how easy it is to navigate around the maps you create



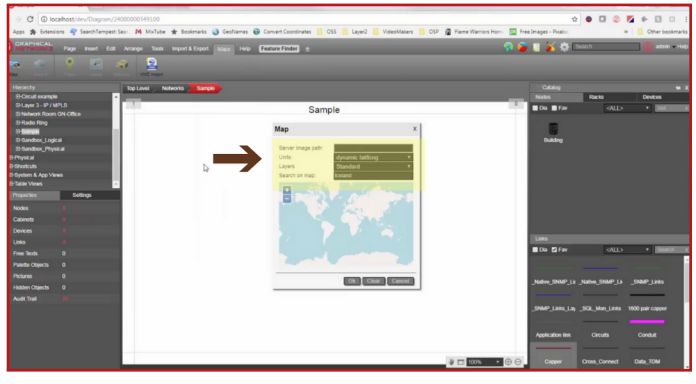
We'll start with a completely empty diagram (see above image) and add a dynamic map to it.



To do this, click on the 'maps menu' in your ribbon (as in above image) and you'll see the map button: you can also get to the map button by right-clicking on a diagram. Once you click on the map button, you can choose the specific map you need.

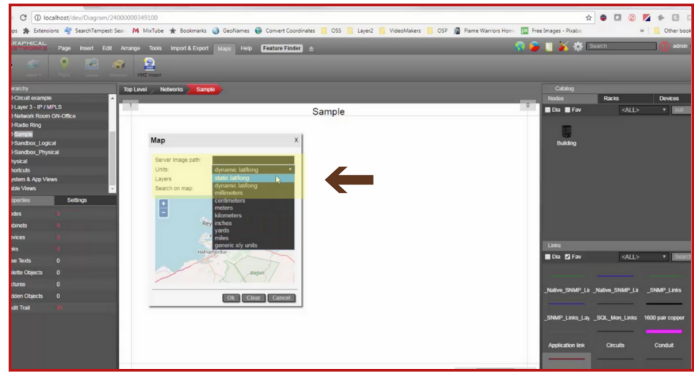
For this demo, we'll work in Iceland (because I've always wanted to visit) — and add some objects into the capital city of Reykjavik. At this point, make sure the 'units' here

are set to a dynamic latitude and longitude — without this set, your map is static (see image above). With a static map, you can still set your coordinates — but it's not going to be a dynamic map created using OpenStreetMap.

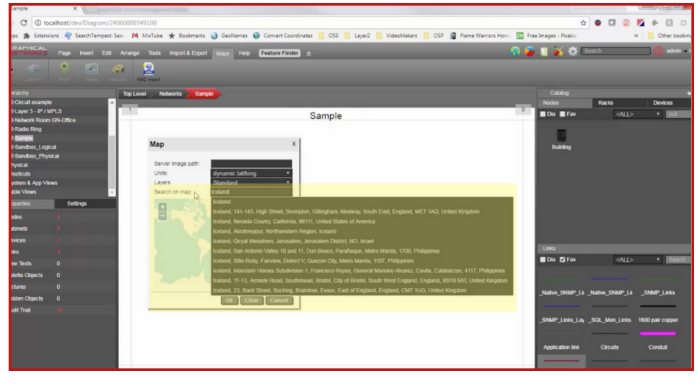


In general with netTerrain OSP, you'll use dynamic network maps (not static) for managing your objects in large geographical areas ('dynamic' refers to OpenStreetMaps (aka GIS maps or geo-referenced maps). netTerrain's dynamic network maps have coordinates attached: you can zoom in and see your locations in great detail.

To give you some background, OpenStreetMap is an open-source mapping technology widely used around the world with millions of users. It's the 'Wikipedia of Maps' as it doesn't require any royalties for usage, you can use as many as you want, and you can even contribute to the map (any changes you contribute will be automatically reflected in netTerrain).



Need to find something specific? You can search by any criteria you put in: your options are endless.



Want to add layers or change your map's appearance? You have three different layers you can use:

Standard Layer

This is our most used one and it has most of the labels

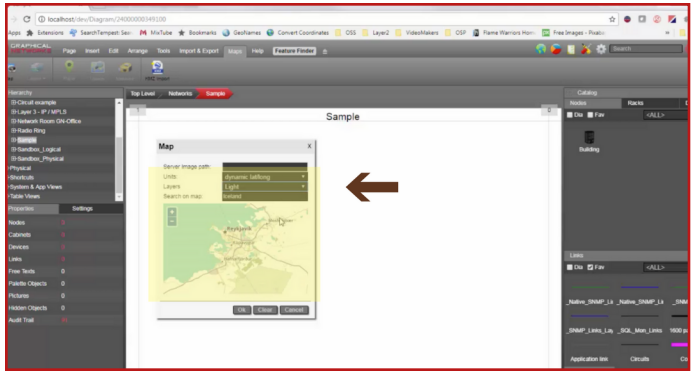
here, such as street names.

Transport Layer

As the name suggests, the transport layer is more geared towards highlighting highways, railways, and so on.

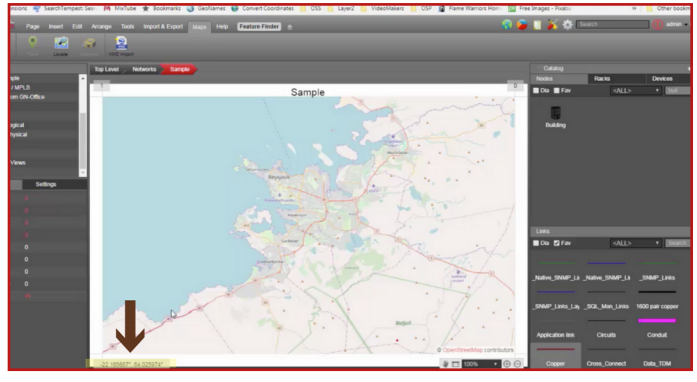
Light Layer

The light option is as it sounds: it's not as crowded and doesn't have as many labels.



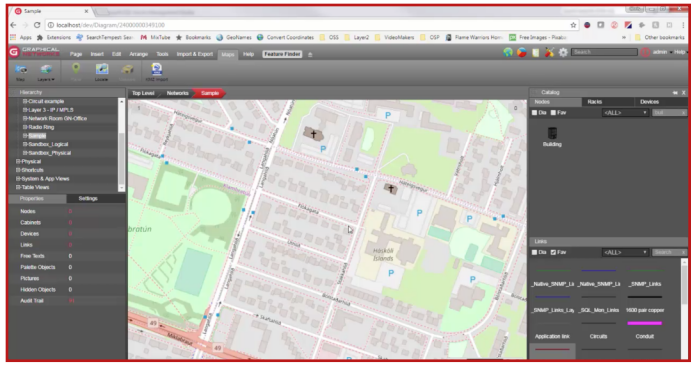
For this demo, we'll use a standard layer (shown above). Though the map you select here defines the boundaries of your dynamic map, changing these boundaries is simple: you can just go back in and select a larger area without losing any of your work. netTerrain will automatically recompute the position of the objects that already existed on a diagram once you make a change.

So, we've now created a dynamic map of the capital of Iceland. The coordinates are on the bottom left corner (see image below). We provide up to 6 decimal places for our lat/long coordinates. As always, you can zoom in: as you zoom in, you'll see the details emerging – street names, houses, and so on.



I created this map while working online, however: you don't have to be connected to the Internet to use netTerrain OSP. Working offline is simple: just download some of the maps, or even the entire world, from an openstreetmap server. You can regenerate them in your own local database if you need to (and we'll provide you with the steps for doing that).

Locating a place is easy with the locate button. Looking for a certain street? Just click on the button and you're there.



Of course, you can do far more than find general places on the map. You can find objects — from manholes to fiber splices (and much more). Stay tuned: we will explore all of the OSP features in chapters to come.

MANHOLES, CONDUITS, DUCTS

Welcome back to our eBook on managing your outside plant with dynamic maps: in this second installment in the eBook, you'll learn how to place manholes, conduits, and ducts within your outside plant diagrams.

In the introduction to this eBook, you got an overview of how you can create maps in netTerrain. To recap, these are dynamic maps: they use OpenStreetMaps to georeference your outside plant assets (with up to 6 decimals of precision).

Once you've created a map, you can start placing different types of outside plant related objects. These objects can be poles, towers, antennas, and the usual suspects: manholes, handholes, conduits, poles, and so on.

The three types of objects most frequently used are:

- Manholes
- Conduits
- Ducts

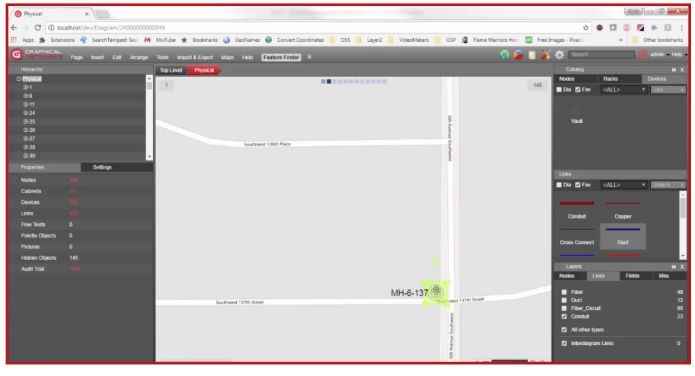
Adding Manholes (& Objects)

Because manholes and conduits are typical examples of objects placed at the map level, we'll start with these. You can place a manhole on one part of the map, place another manhole on another part of the map — and then join them with a conduit. Inside the conduit, you can have a bundle (or an association) of ducts. For these, you can set a number of 4, 8, 12, or however many you need (it's up to you).

How do you bring in your manholes in the first place? If you have them in a spreadsheet, you can use our spreadsheet import. If you have them in a database, you can use the Integration Toolkitto bring them in. In both cases, you can use existing lat/long values to place them, automatically, on the map.

For the sake of demonstration, we'll assume you don't already have your manholes on a spreadsheet or external

system. You'll need to create them manually. Go to the catalog and search for some object that is a manhole (you can use the predefined catalog that we ship with, or you can create your own, or request them from us as part of your maintenance).

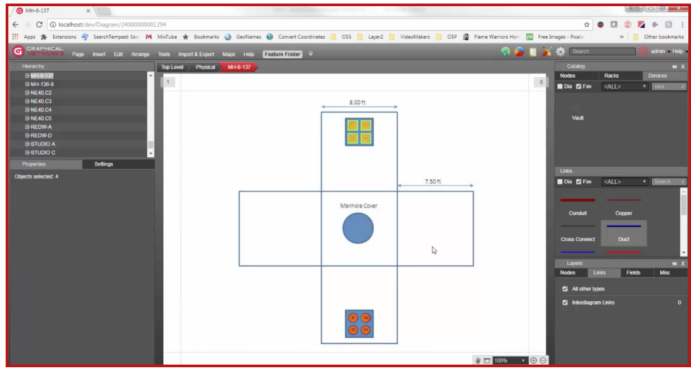


Let's bring in a manhole from the catalog and drop it on the map. You can then rename it: click on it and edit 'Properties' on the left). Type in your preferred name – in this example, we call it 'MH-136-8'.

Let's drop another one on the map — we'll name this one 'MH-6-137'. What are the general components of a manhole?

You can drill down to see. While these manholes are actually modeled as devices in the catalog, they are not actual devices. We do this because they are 'smart': when you create the manhole, or the vault, it already has a specific butterfly diagram – and maybe it has some ducts represented.

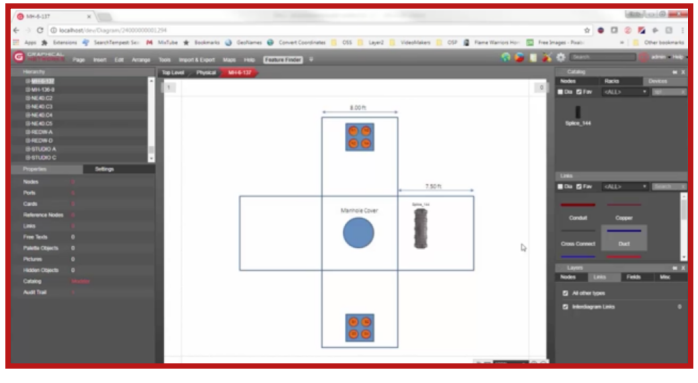
Below you can see the duct starting and ending points — these are typical components of a manhole.



The manhole can have enclosures or splice boxes inside. In this example, we'll add a splice box that has a capacity of 144 fiber strands. Of course, you can add any OSP elements that you'd like documented inside of the manhole.

Inside of the manhole we named 'MH-6-137', we have both the north part of the ducts and the south part of the ducts. Note: north and south are just what we call them here. Your project may be different — it's completely customizable (including the splice box).

Below, you can see the splice box dropped inside of the manhole.



Working with Hierarchies

We have these two manholes here. They are usually joined by conduits (again, in your project, it may be different and you may just work with ducts directly).

You can have as many levels (aka hierarchies) as you want: not just nodes, but also what is generically called a 'link' in netTerrain (a link in netTerrain is anything that connects two nodes).

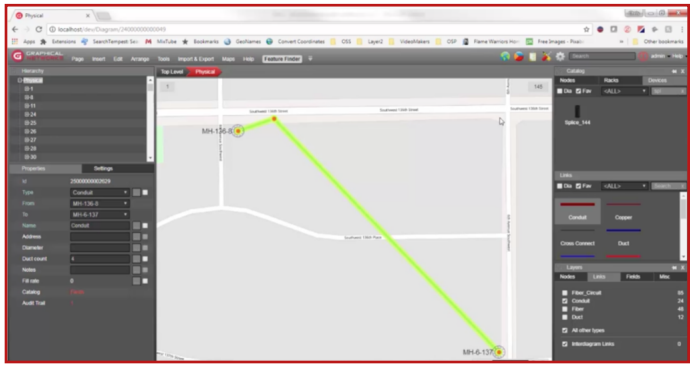
To clarify, when I say unlimited hierarchies, I mean this: you can have a conduit which inside of it has a number of ducts which have a number of microducts — and each microduct has a fiber trunk and the fiber trunk has fiber strands.

In this example, we'll have three levels:

- Conduits
- Ducts inside the conduits
- Fiber strands (we'll associate fiber strands with ducts in a subsequent chapter)

Linking Two Manholes

Creating a conduit between two manholes is simple: just drag and drop. Below, you can see what the conduit looks like when it joins two manholes using the shortest path. Note: this is not what it would look like in real life.



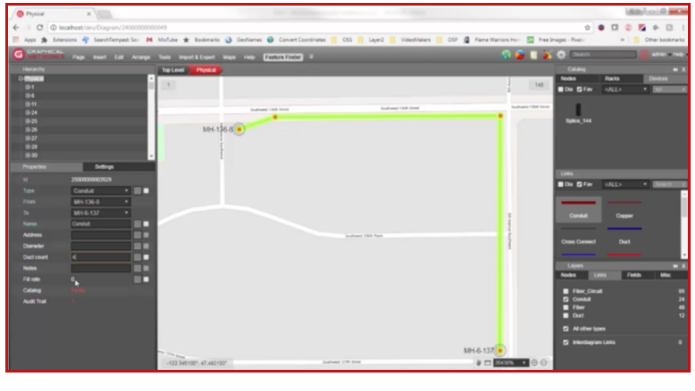
In real life, you would probably somehow reroute it. How do you do that? Using bendpoints. The bendpoints really are just what they sound like: points that bend the conduit line. It's simple to create them: just select the conduit and add them with a very simple shortcut. These bendpoints are not actual objects- they simply will bend the line.

Adding Properties to Manholes

What properties can you have for manholes? To edit property values, click on the manhole itself: you can then edit the values (on the left side). The properties can be anything you would like to track. If you are the catalog manager, you can create as many properties as you want — and you can even import the properties from an existing data structure (the same is true for conduits).

In this example, the conduits have an address, diameter, and duct count. As a catalog manager, you can go ahead and add more properties by going to the catalog(adding them takes a few seconds).

The duct count here is already prefilled with a number 4: this is the default value. What does this number mean? You can add 4 ducts inside the conduit. While netTerrain will not restrict you from adding more ducts, you can set up rules so you get alerted when someone adds more (and through the API, you can set up restrictions, i.e: you can't bundle more than 4 ducts inside this conduit).

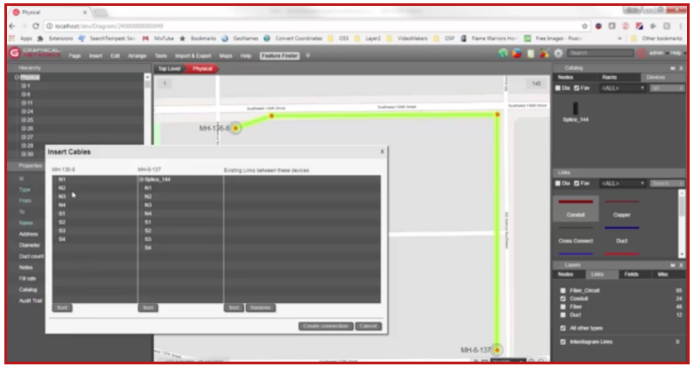


Look at the lower left corner of the above image and you'll see a property that does not seem to be editable. This is the 'fill rate'. It's calculated automatically using a function: the function counts the number of ducts the conduit has bundled divided by the number of count we have for the duct count. This gives you the 'fill rate percentage'.

When you start adding ducts to the conduit, this percentage will automatically change.

Creating Ducts

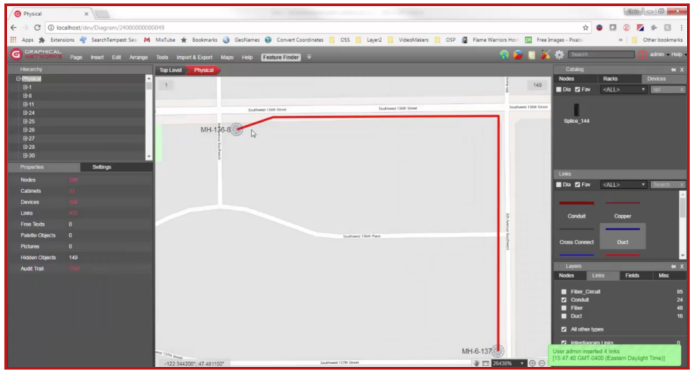
How do you create ducts? With the conduit already selected, it's actually pretty simple. netTerrain has a 'cable mapper': it's one of many different ways to create cables between objects.



In the above image, the cable mapper is showing me, on one side manhole '136-8' and '136-7' on the right side.

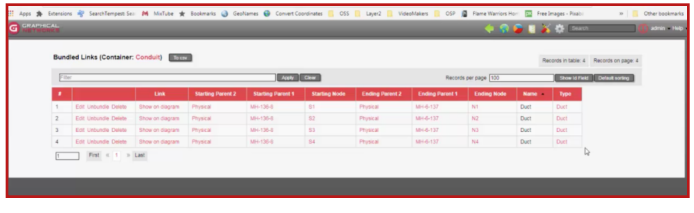
We're going to take advantage of the duct connection points, which you saw on the butterfly diagram, and link 4 on the left with 4 on the right side. We're going to create connections using ducts — and bundle it, automatically, with the conduit.

When full, the conduit changes colors automatically. Below, you can see that it's now red because it's full. Click on the conduit and you'll see that the fill rate is now at 100. You can also see we have 4 bundled links — in this case links are ducts themselves.

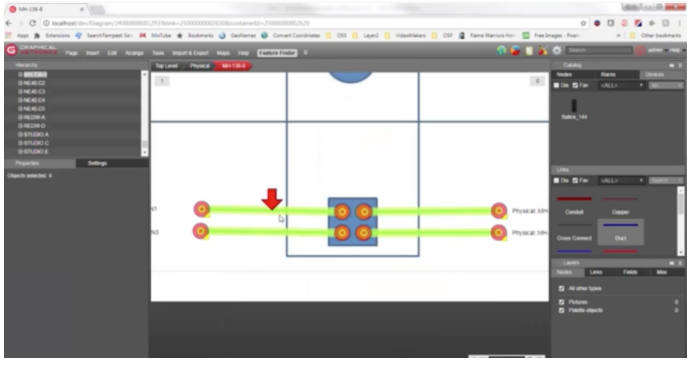


Click on the number 4 next to the 'bundled links' and you'll get a list of all the ducts that are associated or bundled (associated and bundled are just different words for the same concept) with a conduit. Whenever you look at a list view of links associated with another link (here you're seeing ducts associated with a conduit), you can see the starting point and the ending points.

In the image below, all the ducts start at manhole MH-136-8 — and end at manhole MH-6-137 — and all the corresponding positions.



Under 'link' across the top, you can click on 'show on diagram' and it will take you to the starting point of the duct.



It takes you inside of the manhole and you see inside of the butterfly diagram that specific duct that I just clicked on.

Up to this point in our eBook, we've covered how to create dynamic outside plant maps in netTerrain and how to place different types of objects on the map. In the next installment, you will learn how to map fiber inside of splice boxes.

MAPPING FIBER STRANDS IN A SPLICE BOX

In this 3rd segment in our eBook on managing your outside plant with dynamic maps, you will learn how to apply netTerrain OSP to a very common scenario in GPON network mapping and fiber documentation: creating fiber strands and mapping them to a duct.

To recap what you've learned in the eBook up to this point, we've

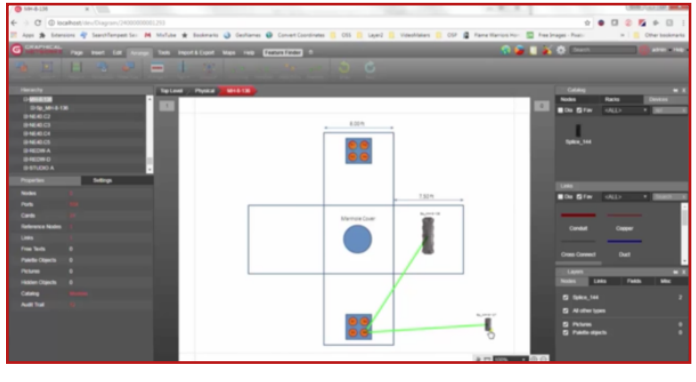
- created a map,
- and placed manholes within it.

With the manholes, we:

- named them,
- added some splice boxes inside of them,
- connected the manholes with a conduit,
- and we also created a bunch of ducts between the splices inside the manholes.

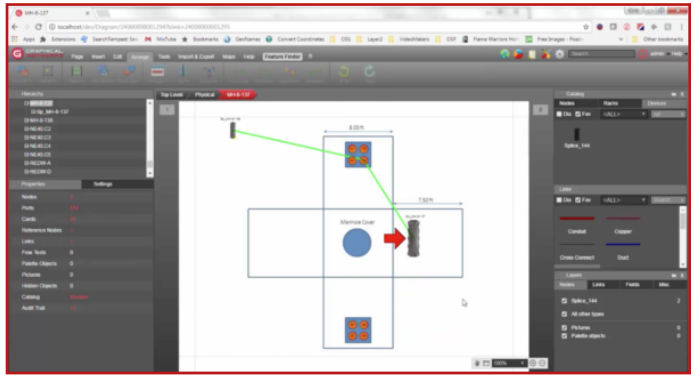
Mapping Fiber in Splice Boxes: Introduction

In the example we'll be working on here, you'll learn to map the fiber strands we create directly to the duct — this is to keep things simple, although, as a general rule, you'll probably have a buffer in between. Similar to the example we used in segment #2, we have a conduit which has one duct inside (note that this is considered a 'bundled' link in netTerrain). Double-click on one of the manholes: you can see that we have a splice box that goes through one of the duct connection points — and then out to the splice which is inside the manhole located southeast of this manhole.



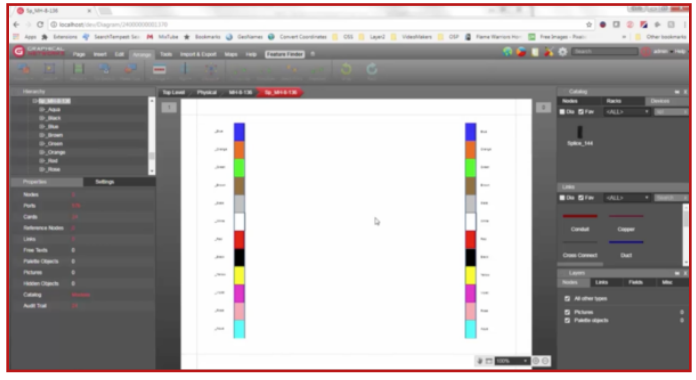
Need to go to the other manhole? Double-click on this splice box (which serves as the reference point here) to

do so: if you go up, it shows up here on the manhole.



Working with Fiber Strands & Splice Boxes

Before we dive into working with fiber strands, we'll take a look at the actual splice boxes. Inside of this manhole, you can find our splice. The splice you see here is just one way you can depict a splice box in netTerrain: you can have different types of splice boxes and we can model them in our catalog for you (as part of your maintenance, you can always request custom models from us — just request a model and we'll get it back to you within 24-48 hours).



In this example, our buffers will follow the standard color coding:

- Blue
- Orange
- Green
- Brown
- Gray
- White
- Red
- Black
- Yellow
- Violet
- Rose
- Aqua

To see each of the fiber connection points for a buffer, just double-click it: the fiber strands will be connected to these connection points (which are treated as ports in netTerrain).

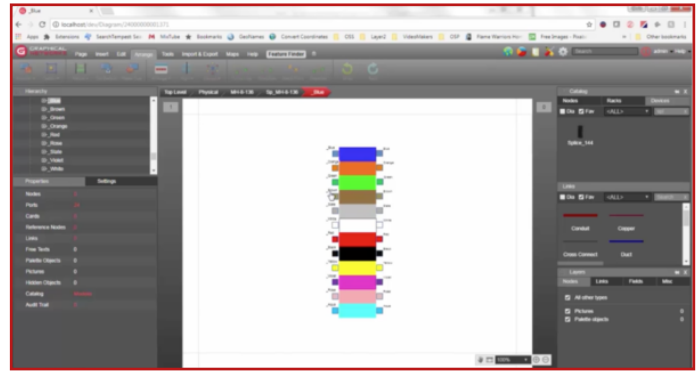
We have 12 buffers here: if you go up, you have 12 that go in and 12 that out for the splice box. This setup is based on how it is in real-life: you'll have the fiber coming from the left and it's cross-connected to the right. A note about colors: there is no hard-and-fast rule that dictates the same color code be used in cross connections. You can, for example, have the blue and orange cross-connected to the green and yellow.

This is a good place to pause and illustrate just why hierarchies are so important to the success of your work: depending on what you do, the colors won't always perfectly match up and having your fiber in a hierarchy — inside the splice box and using these buffers — makes life easier. Simply drill down on one of the buffers, select your connection points, and cross-connect them to the other end using the color codes.

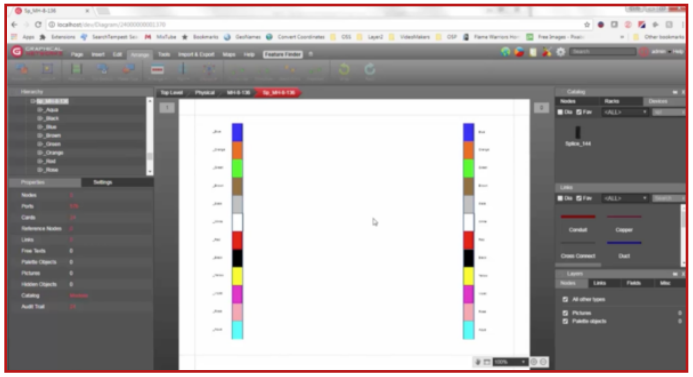
Creating Fiber Strands

Next, we'll create 12 fiber strands.

One method you can use to create your fiber strands is to click on a duct: inside it, you can see we have here connects two splice boxes. To create your 12 fiber strand, start by bringing up the netTerrain Cable Mapper as you can insert cables here.



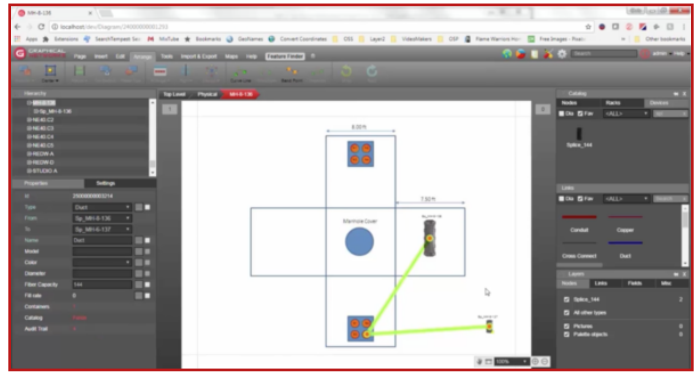
Above you can see the cable mapper's intercable dialog box: in the left column, you can see one splice box 'MH-8-136', and, on the right, one splice 'MH-6-137'.

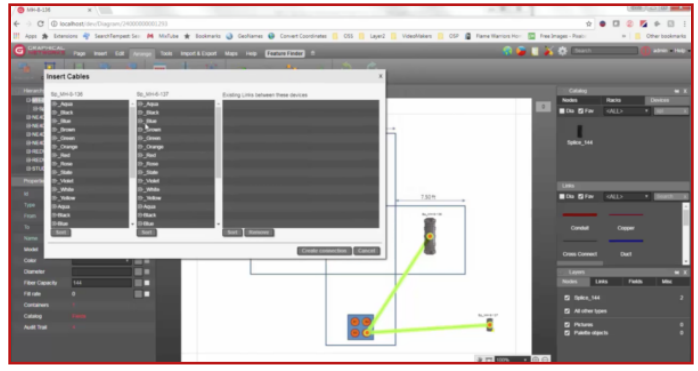


Within each column, every buffer is listed – both on the left side, and on the right side.

In the column representing the 'MH-8-136' splice box you can see that every buffer on the left side is marked with an underscore while the buffers on the right don't have an underscore: look at 'MH-6-137' and you can see it follows the same structure. The use of the underscore here is just a personal notation I've chosen to use in order to differentiate the two sides — and to keep it sorted nicely (and simply).

n general, what you'll want to do is to create 12 fiber strands that connect an outgoing part of a buffer on this splice box and connect it to 12 connection points, on this device, on the incoming side. Again, as with most things



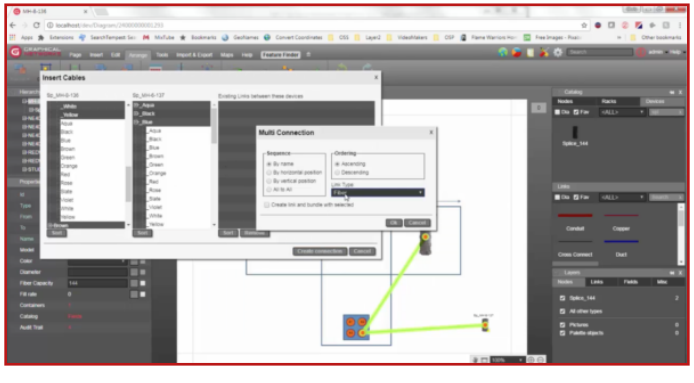


in netTerrain, there isn't a hard-and-fast rule as to how things should be done: there are a myriad of ways to create these connections and you can change the notation — this example is just what works for me and how I've seen it done in many other instances. It's simple because you've got the standard notation with the colors there so you can then simply expand.

To keep things simple, let's connect blue with blue. You

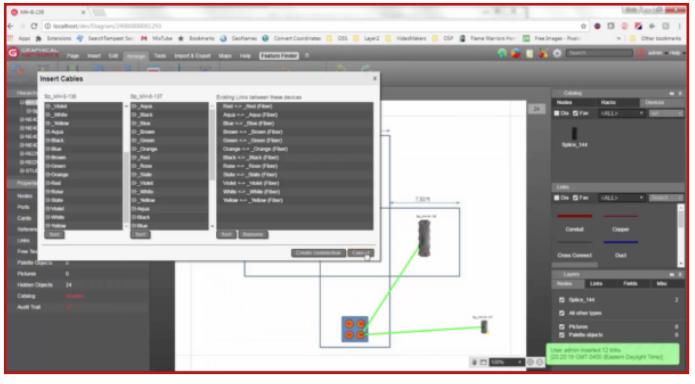
can connect all of the outgoing fiber strands on the outgoing buffer on my local splice box to the incoming strands on the incoming buffer on my remote splice box: it simply multi-selects.

It's easiest to pick the same sequence of colors: here, we'll have 'blue aqua' with 'blue black', for example. Again, you don't have to do it that way — you can pick any color you'd like. For example, you can pick 'slate aqua' with the 'black brown' and create the connection. Bottomline? Cross-connect any which way you need to.



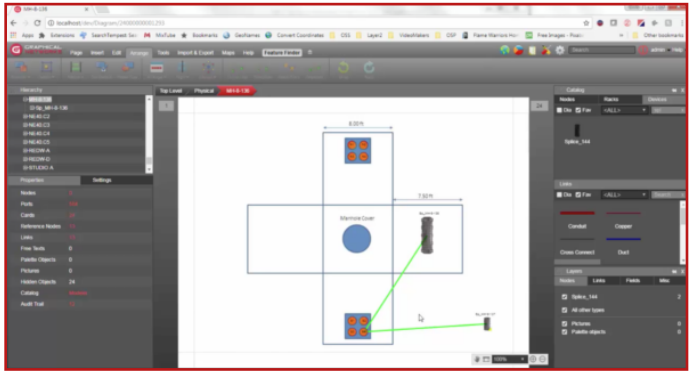
'

Create a Cable Hierarchy



Ready to create a cable hierarchy? We'll use the blue, click 'create connection', select 'fiber' (this is the link type we will use here — you can create your own cable type in netTerrain), I'm going to check this box to bundle "create link and bundle with selected": the 12 strands i'm going to create with the duct itself. You've now created your cable hierarchy.

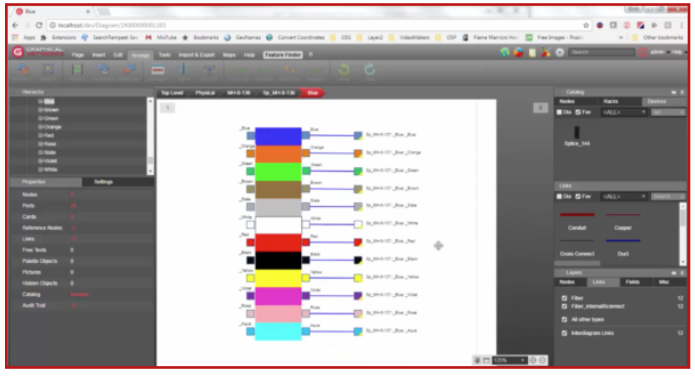
So: we now have the red with red and aqua with aqua and so on. If I click on the duct, we can see that we now have 12 bundled links (this shows in the properties bar on the lower left of the screen).



In the properties bar, you can also check out the fiber capacity for the duct: in this example, fiber capacity is set at 144. In this example, I've set a visual override that changes the color of the duct based on the fill rate — this works the same way as we did with the conduits in the last post: it's based on how many fiber strands you have. As you can see in the properties bar, the fill rate here is still at 8%, which is very low.

Putting It All Together

Curious to see how all of our work looks in the splice box now? Here's how you can see these changes: navigate to the local splice box, filter the fiber out (layers > links on the lower right side) , and go to the blue buffer.



You can see inside your buffer that the blue is connected, remotely, to the blue on the other splice box. You can see that the blue buffer is connected to the blue fiber, the orange fiber in our blue buffer is connected to the blue buffer on the other end and the orange fiber — you get the idea.

In sum, we just created 12 fiber strands in netTerrain's outside plant management software. As you can see, it's a very easy process: these are automatically bundled with the duct simply by selecting the duct and bringing up the insert cable dialogue box, aka the Cable Mapper in netTerrain.

Coming up in the next segment of this eBook, we'll show you how, once you've got all this fiber set up, you can trace specific fiber strands end-to-end. We'll be looking at examples of circuit layout records.

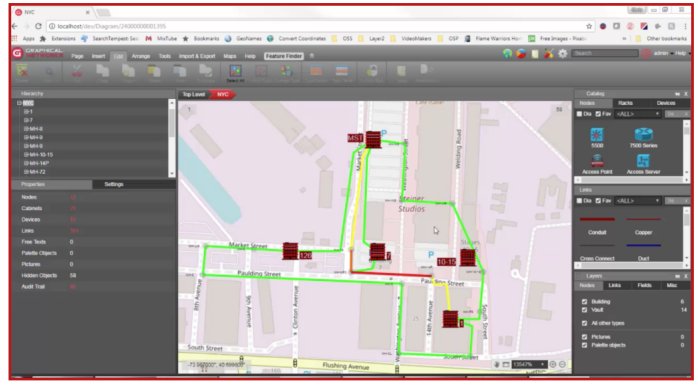
TRACING CIRCUITS

In this 4th segment of our eBook on managing your outside plant with dynamic maps, you will learn to trace a cable in our outside plant software netTerrain.

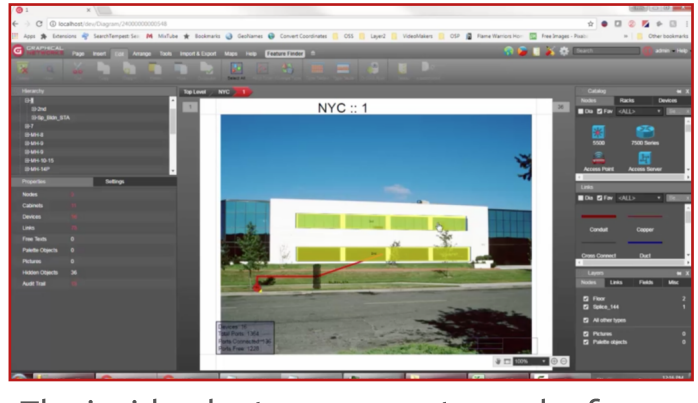
In the previous segments in this OSP eBook, you learned how to: create a map in netTerrain OSP software, how to add objects to it (buildings, manholes, handholes, conduits, ducts, etc), and lay everything out. Now, you'll learn how to inspect an individual cable.

Tracing Circuits with Dynamic Outside Plant Maps: Introduction

What kind of cable can you trace? Anything at all: it could be a fiber cable, a copper cable, or even a circuit. In this segment, you'll also get a glimpse of how netTerrain Outside Plant software also helps you manage your inside plant: we will give you a look into a couple of the buildings you've seen on our maps.



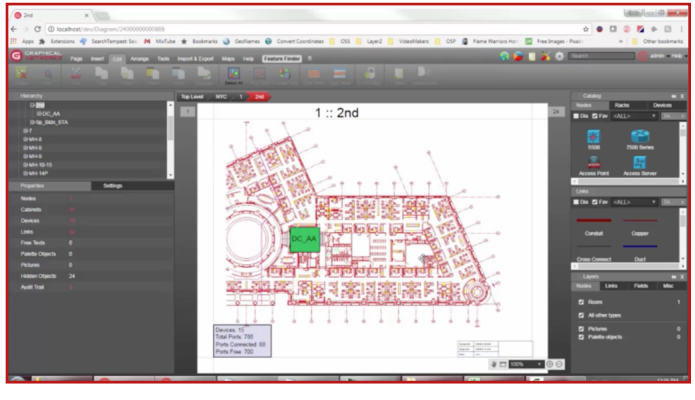
netTerrain OSP includes inside plant management for one simple reason: networks need to be managed not only from the perspective of the outside plant but will, almost always, include some inside plant elements that need depiction.



The inside plant components can be from a small set of points of pre-sets — or maybe a couple racks here and there or some loose equipment — and can scale all the

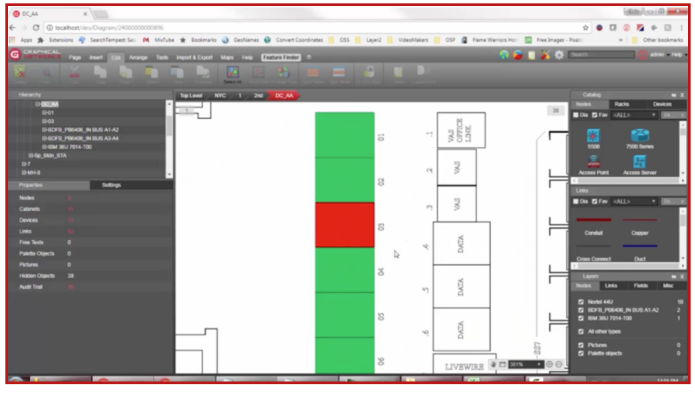
way up to large data centers.

Let's drill down into one of these buildings and see what we have. While some of our outside plant customers choose to show a picture of the building itself — sometimes with some of the floors, etc. — we won't need to go into every single detail of all the things you can do in terms of inside plant. If you want to explore more of what we can do with our inside plant documentation, we have numerous resources available to you around our DCIM software.



In this example, we'll just glance at the inside plant depictions in this current project so that you can understand how you can trace an entire circuit — inside to outside — when needed.

Let's drill down from the floorplan view to a specific room view: so, for example, below you can see that we've got some racks here going from the floorplan view to a specific room view: we have some racks here (and, just to note, the hierarchy we are navigating is not mandatory: you can build your own hierarchy).



You can color code the racks based on different criteria (again, we won't go into the details of all of the things that you can do with netTerrain DCIM) but in this

example, we do want to know about some of the equipment here because the circuits will start and end on actual devices that are in the inside plant.

The circuits will traverse the outside plant and that's what we will trace here. Many software solutions will give you inside plant documentation (aka data center inventory) in one tool — and outside plant in another tool: keeping the two silos separate.

The problem with this method, however, is that when it's time to trace a cable, it can become messy. Here's why:

- It's either missing some of the ports (the inside plant stuff)
- they have to create it manually
- or it becomes very complicated maybe they have some sort of in-house tool or something that creates the trace.

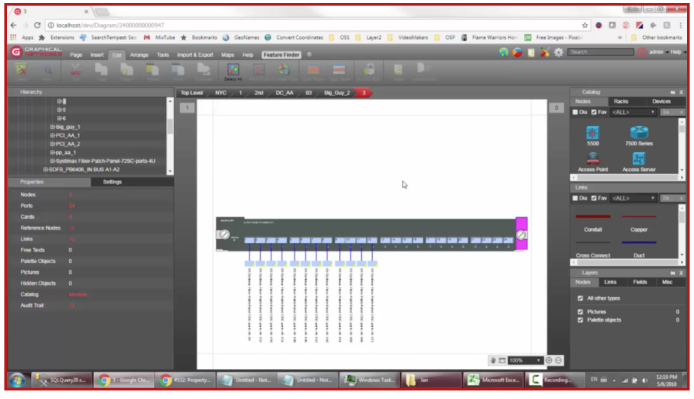
The result is usually something like a blob diagram or maybe just some text.

Tracing Circuits with Dynamic Outside Plant Maps: What are Circuit Layout Records?

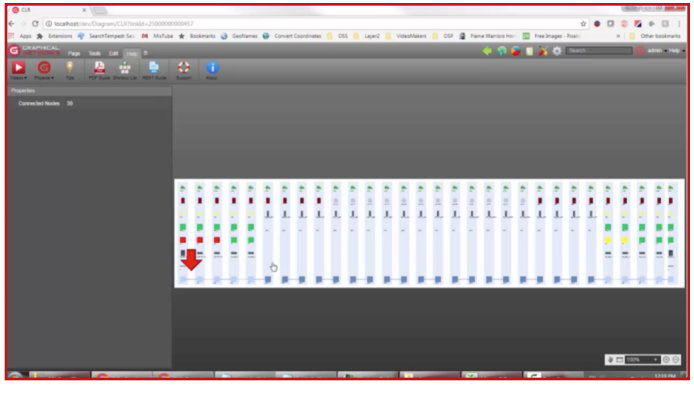
In netTerrain, we have something called the Circuit Layout Record (CLR) that solves this problem. We use the CLR to actually trace a circuit, This is how we actually trace a circuit, in general, in netTerrain — so our customers, once they have their circuits of their cables all connected — then they launch one of these CLR's and that is how they can find out how that cable, or that circuit, traverses from A to Z — including all of the hubs in between.

Tracing Circuits with Dynamic Outside Plant Maps: How to Launch a CLR

So how do we launch a CLR in netTerrain? You can launch one from anywhere in your entire project — and from any part of your circuit. Below, you can see we have a card for a router and 12 cables which have traversed all the way to the other end.



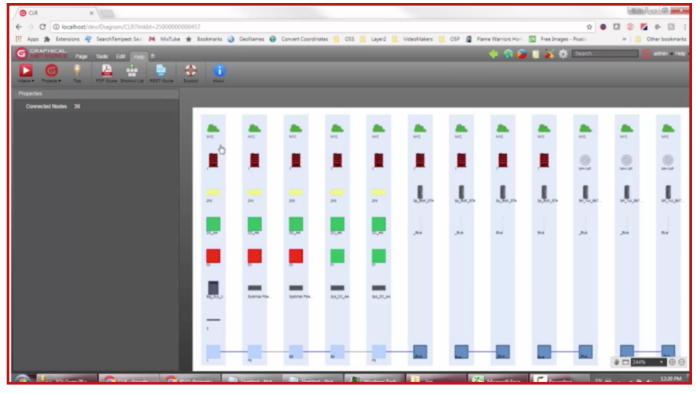
Just right-click on the port and you can launch the CLR from here; you can also just double-click on the particular cable itself and it will launch the CLR.



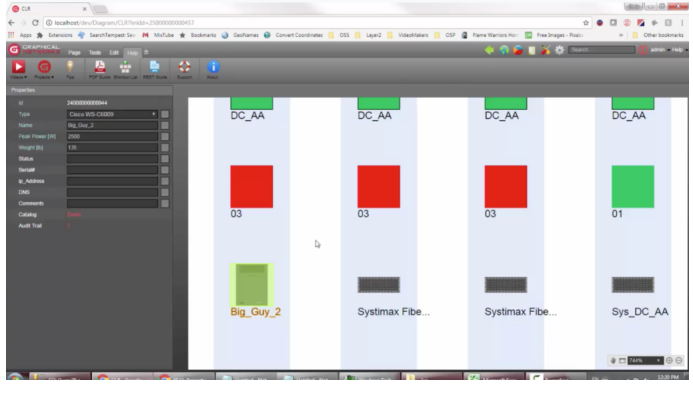
Tracing Circuits with Dynamic Outside Plant Maps: Traversing the Hierarchy

This CLR is quite large: it has a large number of hops.

There is no limit on the number of hops, however, these diagrams can become long. To make it easier to view a long diagram, you can export it using a tableview.



We started the CLR from the left side: below, you can see that we're at one end here for port '1' on card '3'.

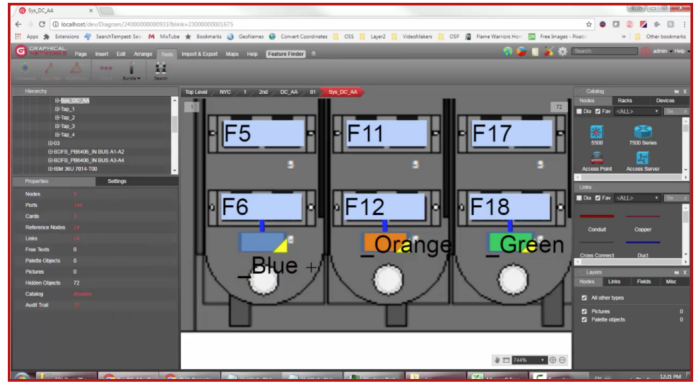


'Big Guy 2' is our router on rack '3': DC_AA is in our building #1 NYC – we'll make a jump, aka a hop, to port

'F-6' on this fiber patch panel.

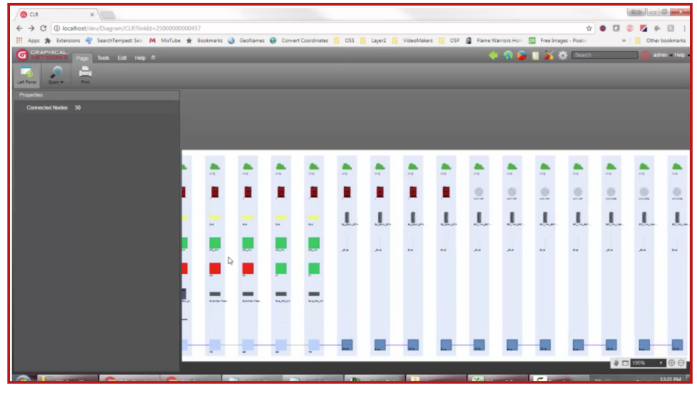
If you go up, you have the hierarchy: go to the right, and you have each hop. As you can see, this is pretty simple to read and it also gives you an added advantage: if, at any point, you want to inspect a particular component in your CLR, just double-click on it and that takes you to the project where the specific component lives. For example: double-click on port 'F6' and it takes you to the project to port 'F6' for that particular fiber distribution panel.

Here you can see 'F6' that is connected to a port in some other part of the project. You can actually navigate this same circuit simply by traversing the hierarchy in netTerrain itself: you start hopping from one end to the next over to here (and so on). You quickly see, however, that if you do that, you could very easily get lost. This is where the CLR plays a vital role: it gives you, in one simple and single view, the entire trace for the circuit – from A to Z.

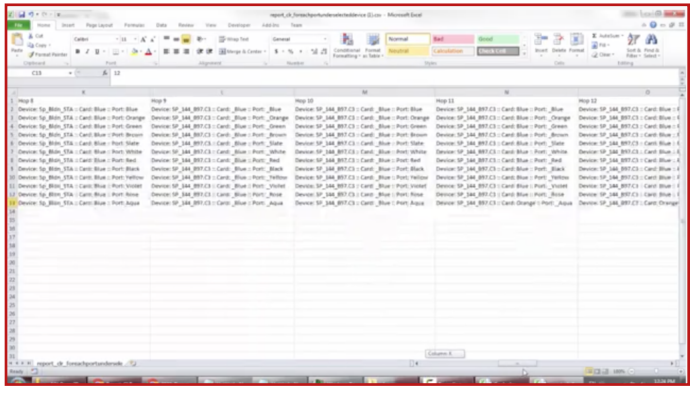


Tracing Circuits with Dynamic Outside Plant Maps: Putting it All Together

Let's go back to the CLR. **This is important:** when you're dealing with both inside and outside plant in netTerrain, you can see that in the CLR itself there is a gap here. This gap (see below) will be the outside plant.



information and process it in any way you'd like. Below for example, you can see each hop here for 12 circuits with the entire hierarchy.



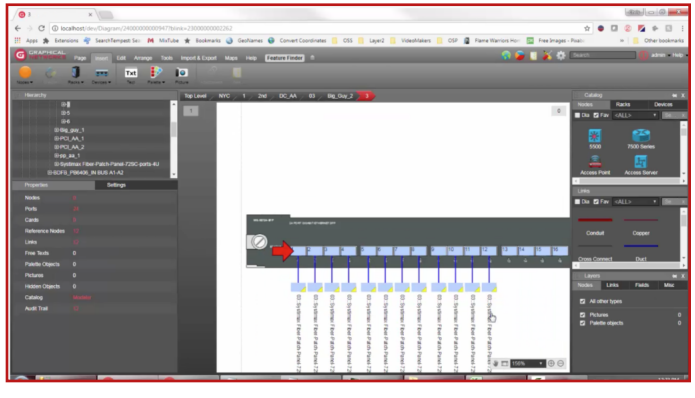
Tracing Circuits with Dynamic Outside Plant Maps: Conclusion

In sum, you should now have an understanding of how you can create, manage, and visualize circuits (and their entire components).

Now that you understand how circuits in netTerrain traverse from A to Z, you can now quickly find what you need for an outage, a decommissioning (so you know which parts you need to uninstall) or in any troubleshooting initiative — information that would simply be too hard and too time-consuming if you have to start looking for all of the disparate information in your different silos.

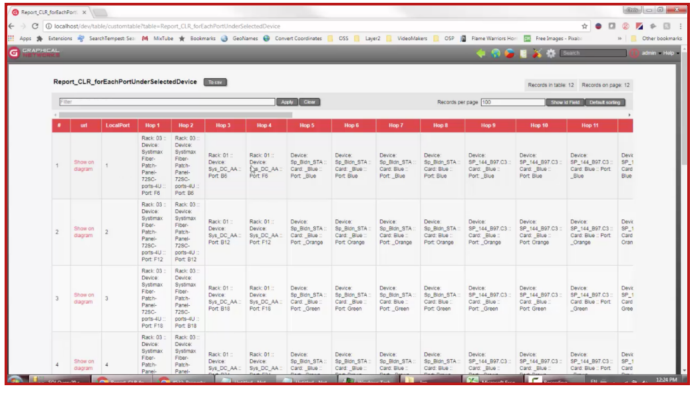
Why is that? The inside plant typically includes the rack, the room, the floor, etc. You can visually see that the hierarchy is different; with the outside plant, you're at the map level which includes components such as splice box, splitters, and the like. It's very easy to identify which is which because the icons in the inside plant are different from the outside plant icons.

Next, let's return to our starting point.



Within the router, you can see that we have 12 cables here (not just one). If you were to give this to a technician, it's possible that they may want to see all 12 CLRs. If you were to try to depict all of them in just one diagram, this creates a big real estate problem…

There is an alternative: create a tableview of that CLR for the multiple circuits. You can do this very easily: just right-click on the diagram (in this example, the diagram of the card that has the 12 circuits) and this will spit out the tableview for each of the 12 circuits. In this example, we have, from top to bottom — each circuit and, from left to right — each hop.



The hierarchy itself is produced and presented as text for each hop: while this is not that easy to understand as a diagram,you get the advantage of having all 12 circuits in one place.

You can export this to CSV and open it in Excel. Initially the format will be very raw but you can take the

OUTSIDE PLANT SOFTWARE OPTIONS

netTerrain OSP helps you visually manage your fiber, copper, transmission and wireless networks; you can take control of your outside plant connectivity with dynamic maps that let you zoom down from world-view to the street level. So, if you need a tool to assist in graphically managing your fiber network on GIS maps, your outside plant objects, or any network of assets requiring geo-referenced maps, you can now do that with netTerrain OSP.

The Good, the Bad & the Ugly

One could probably make the case that the fiber optic network documentation market is overcrowded….that is, until you start looking under the hood. Believe it or not, the options on the market are fairly limited. How else, then, could we explain why so many of our netTerrain customers asked us to come up with a good solution?

Before netTerrain OSP hit the market, you could find three types of tools for fiber optic network documentation:

The (Unaffordably) Good

Ah, the good tools – and the most expensive ones. Of course, there are tools out there that offer the goods – but they are simply way too expensive. You get the features, and (hopefully) the usability you need – but expect a high 6 figure or 7 figure price tag. Some of these tools are part of a larger inventory or provisioning OSS – but if you are responsible for just the fiber base piece, you'll still end up shelling out the big bucks – even without some of the other modules.

In sum: good but unaffordable unless you are Verizon.

The Bad

Of course, out there in the wilderness you will find fiber management / outside plant tools that don't cost an arm

and a leg. They are packages exclusively suited for campus fiber management and the like, but they are limited in many ways:

* Usually not web based, and in many cases outdated code

* Limited or no inside plant tracking

* Poor visualization capabilities

* Not enterprise grade (limited reporting, auditing, security, etc)

In brief: affordable but deficient.

The Ugly

We have also seen many organizations that cannot afford expensive fiber network documentation tools opt for some sort of custom solution such as by using open source software like Google Earth – or maybe around an affordable tool like AutoCAD. Spreadsheets cannot be used to visualize the data or place objects on maps so migrating to a custom implementation of AutoCAD or similar may feel like a step up. This approach is very common in campus fiber documentation projects or for small to mid-sized fiber base deployments.

However, these tools offer no inherent features for dealing with fiber base or outside plant specific features. They have no built-in business rules to deal with some of the complexities of documenting fiber optic networks, such as: fiber strand bundles, circuit layout records, circuit rerouting, shortest path calculations and so on….the mapping capabilities of these tools are fine, but that's where they typically end.

Enter netTerrain OSP

netTerrain OSP bridges the gap between the capabilities users need and the pricing they can afford.

Below is a table summary of netTerrain vs the good (and pricey), the bad and the ugly. Enjoy!

OSP Software Comparison

	Price	Features	Flexibility
netTerrain	✓	✓	✓
The Good		✓	✓
The Bad	✓	✓	
The Ugly	✓		

GRAPHICAL NETWORKS
814 W. DIAMOND AVE
SUITE 370

GAITHERSBURG, MD 20878 USA

+1-240-912-6223

WWW.GRAPHICALNETWORKS.COM