



HOW TO CHOOSE THE RIGHT DCIM SOLUTION

JAN DURNHOFER
GRAPHICAL NETWORKS





TABLE OF CONTENTS

- 04 INTRODUCTION
- 09 ARCHITECTURE & USABILITY
- 14 ARE YOU READY?
- 22 TRY BEFORE YOU BUY

INTRODUCTION

So: you've been tasked with finding the right Data Center Infrastructure Management (DCIM) tool for your organization. In baseball, before you step up to the plate, you first take a look at the whole field. The same is true for DCIM.

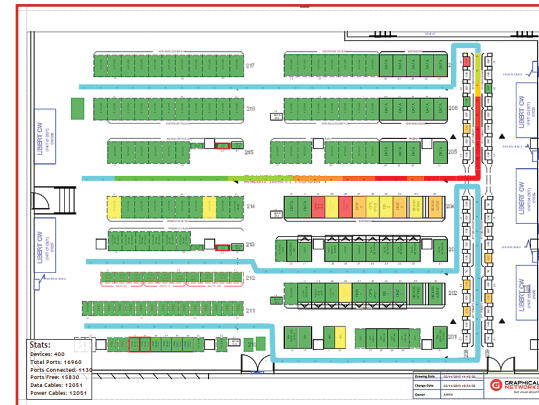
This book will explore the ins and outs of choosing the right software – and the right company – to work with. You may have to go through the usual motions of writing an RFP and evaluating responses. We feel for you if that's the case. RFP's may be necessary in many cases for a variety of reasons, but don't plunge straight into the tedious process of writing an RFP before reading this!

What is DCIM?

If you ask ten different people to define DCIM, you will, most likely, get ten different answers. Why? DCIM spans a broad spectrum of data center functions. We can talk for hours about how the definition of DCIM has been muddled by different vendors in the space.

Yeah, yeah, DCIM is whatever the customer wants it to be. We know. But if we want to have a conversation about how to choose a DCIM we still have to somehow define it, so here's our take: DCIM is a process and set of solutions to track, monitor and manage the data and

power utilization of all IT-related assets (hardware and software) as well as infrastructure components (racks, cooling units, trays, floor plan space, etc.) that are related to the Data Center for the purpose of optimizing consumption and reducing cost.



When choosing a DCIM solution, start by finding out what your specific needs are and the level of preparedness of your organization for managing that solution (and process) . With a DCIM solution in place, managers gain access to data, power and cooling info, space resource consumption and growth patterns for your IT resources, as well as the ability to plan and forecast when a resource will expire or when a new purchase will be on the horizon. Resources are optimized and maximized.

DCIM is a powerful way to automate data center operations while maximizing the ROI from company resources. The DCIM solution you choose can come with

a wide variety of functions. Your organization may need all of them or just a few. Probably no tool in the market will perform all the functions that have been touted in the past as part of DCIM and you probably don't need to worry. Just because you can do something, doesn't mean you should. DCIM solutions can be expensive and take a long time to implement, so focus on the core and use sound criteria when evaluating solutions.

To better gain an understanding of what a DCIM solution would mean in your organization, let's take a look at some of the most common DCIM sub-categories:

Floor & Rack Management

How do you manage space and racks? Where is space underutilized? Where does new equipment go? Space and rack management can greatly assist data centers in streamlining and optimizing physical space, rack unit utilization and geographical location of assets.

Environmental Power & Temperature Monitoring

DCIM solutions assist with real-time power and temperature readings bringing them into one view that can relate them with the underlying IT components that are affected. DCIM solutions offer managers comprehensive views into energy usage and thermal

data on the device, rack, room and overall data center levels.

Asset Management

This is a key part of what DCIM is all about. As you know, data centers can contain thousands of assets and accurately tracking all of these can be nothing short of a herculean task. How long does it take to locate a server once it goes down? Not being able to locate equipment quickly when needed can really put a damper on things. Beyond just finding an asset, asset management includes getting details about how its configuration. What is a server connected to? If a rack power strip is disconnected, will the server shut down?

Proper asset management helps data centers run with greater efficiency while reducing costs. Then...there is that little thing called discovery. We will talk more about asset management and discovery capabilities in a subsequent chapter. This is a key part of what DCIM is all about. As you know, data centers can contain thousands of assets and accurately tracking all of these can be nothing short of a herculean task. How long does it take to locate a server once it goes down? Not being able to locate equipment quickly when needed can really put a damper on things.

Beyond just finding an asset, asset management includes getting details about how its configuration. What is a server connected to? If a rack power strip is disconnected, will the server shut down? Proper asset management helps data centers run with greater efficiency while reducing costs. Then there is that little thing called discovery. We will talk more about asset management and discovery capabilities in a subsequent chapter.

Cable and Circuit Management

Cable and circuit management is a whole topic in and of itself (see our blog on the website for that). If you have cable management requirements, this is a perfect area to get stuck with an unusable tool that looks good on paper (or Power point). A DCIM with proper cable management features is not just limited to being able to track a connection from A to B. It needs be able to discover them (at least, what is discoverable), visualize them, trace them, create hierarchies of connectivity and above all, do all of this easily.

Logical, Virtual and Application Views

With all the virtualization happening today you probably are interested in mapping which VM images run on which host. You may also want to obtain other

application data so you can map them to the appropriate virtual and physical host in your data center. Moreover, it would be nice if the same device that is logically depicted as part of a network serving specific VM images, applications, systems and so on, can also be physically displayed on the appropriate cabinet and rack position.

Computational Fluid Dynamics

DCIM works in tandem with computational fluid dynamics: complex airflow and thermal patterns in the data center can be modeled and monitored for optimum Data Center design.

Do You Know What You Need?

With so many sub-categories, it's key to have a solid understanding of your organization's requirements and needs before you choose a DCIM solution. Let's look at an analogy.

When you go car shopping, you probably walk in with a certain set of specifications based on how you'll use the vehicle. If, for example, you need to haul heavy equipment, you've probably got a certain kind of truck in mind – and you'll probably do some research on the

different load capacity engines. You'll probably also test-drive that truck. It's hard to believe, given price tags that dwarf the cost of large trucks, but not only do many companies fail to test drive a DCIM solution; they don't even have a solid understanding of the requirements they have to meet their needs.

Think about it like this: you wouldn't buy just any vehicle a salesperson suggests, would you? Knowing what you need and how it will work for you is an essential step in the DCIM purchasing process. The proof of concept then lets you test drive a DCIM solution to fully explore how it will work with your data.

What's Next?

In this book, we'll be examining the many functionalities of DCIM and how they relate to your organization's needs.

Once you've gained a solid understanding of how DCIM solutions can work for you, we'll take a look the important proof of concept.



ARCHITECTURE & USABILITY

In the previous chapter, we started with an intro to How to Choose the Right DCIM Solution. We gave you a definition of DCIM and then enumerated different functions that usually fall under the large category of “DCIM”.

Alright: so you’ve identified some of the features you need from that list, and you start looking for vendors that tick those boxes, right? Not so fast.

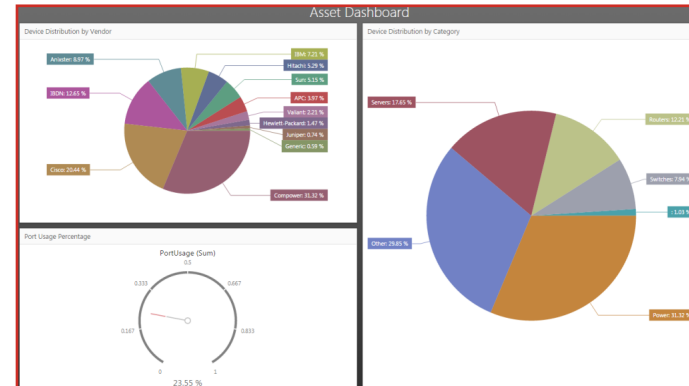
Features-schmeatures

In this second article of the book, we will not talk about features. We will discuss something even more important: the underlying architecture and usability of the platform. These are aspects of both the software and the vendor that you must consider; these are things that go beyond a laundry list of must-have features. Here’s why: all the features in the world will be useless if the software is not securable, easy to use, open, or easy to deploy. It will turn into shelfware. So, let’s start with the deployment itself.

Easy Deployment

News flash: we don’t party like it’s 1999 anymore. This is 2015, where HTML5 is the norm and users don’t want to be or need to be installing yet another piece of software

on their desktops to access DCIM data. So, for us, easy deployment essentially means no deployment at all! Of course, this can mean a SaaS solution for many (which we do offer), but SaaS is non-starter in many situations and for a number of valid reasons.



That’s why the vast majority of our customers choose to purchase a permanent license of our DCIM tool that will be hosted in their own network. So what about deployment in those cases? The server should be easy to install and end-users should have to do zilch. Because DCIM solutions are (or should be) diagram-rich and feature-rich, this probably means a solution that uses HTML5/SVG rendering (or similar) and requires no fat clients, no add-ons and no plug-ins. In other words: open a browser, log-in and enjoy a feature-rich user experience.

Usability

Fact: 90% of drivers think their driving is better than average. That is, of course, mathematically impossible. Yet, I am certain 90% of DCIM vendors would say their solution is easier than average. In another upcoming chapter, we'll discuss usability and the importance of the proof of concept, since we are passionate and adamant about both, but let's just say this: put the vendor to the test.

Come up with some sequence of operations, such as modeling a new rack type and new router type (and beware of those vendors saying "oh, we'll model that for you in a week" – more on that later), upload a floor plan, add a rack, move it somewhere on the floor plan, add a big router with some cards and a patch panel, and create 24 cable connections between the router cards and the patch panel.

Then, count the minutes (or weeks) this took to create. We could go on and on about usability (such as: ease of creating adapters, importing devices from your home-grown Oracle database, new report types, cloning an entire data center, etc.)...Point is: don't take the vendor's word for granted. Just because they say it is easy, doesn't make it so. By the way, I am a fantastic driver, in case you had any doubt...

Flexibility

I can't stress this enough, and it's also related to the previous point. Let us summarize flexibility by saying that if any of the following questions triggers a no, that's a problem:

- a) Can you create new entities in the system (like, say, a door, or a chair)?
- b) Can you add/remove fields, make them mandatory, unique, displayed, etc?
- c) Can you model new device types and subcomponents in the catalog/library?
- d) Can you create new report types or modify the dashboards?

And let's cut to the chase: you means you. If the vendor tells you that they (as in not you) can do that, and this requires a consultant to do it, then it is far too inflexible already. If on top of that, to add insult to injury, they charge you for wanting to customize the tool to your liking, here's our advice: run for the hills.

Open Architectures and APIs

You can't rely on the vendor every time you need to extract data in a different way. You should be able to do it yourself – with ease. Which brings us to the topic of APIs...There's a lot of talk lately about APIs, particularly in the DCIM field, and its availability from vendors. A lot of vendors are concerned that if they give away too much access to data, they risk giving away the recipe to their "secret sauce."

At Graphical Networks, we believe open data should be an important feature of the tool.

The importance of open architectures and APIs can't be overstated. A DCIM tool that offers you the data you need when you need it and allows you to develop your own tools is far more valuable, long term, than one that does not.

Import / Export Capabilities

This is related to the previous point, but beyond just architecture, your solution should be able to expedite the process of getting data in and out of the system. Whether it's for a bulk import process for the initial deployment, or to export a table or diagram to other formats so that the big bosses can see what's

happening, import and export features are a must. Ask the vendor if they can import csv files and excel spreadsheets, import maps from Autocad, import and export to Visio, export to Powerpoint and PDF and so on. Note: we are not talking about discovery or real-time adapters ... yet. That comes next!

Easy Discovery and Adapters to Other Systems

Every DCIM vendor – and their mothers – will claim that they have discovery and ways to integrate it to other systems.

Unfortunately, this means diddly-squat without understanding the following:

- a) What exactly does discovery mean?
- b) How easy is it to create integrations?

In our opinion, if the end user cannot create integrations themselves, then it is too complicated. In this, we are quite adamant: there is no excuse for not having an integration builder or framework. At a bare minimum, the vendor should create connectors for you at no cost.

Yes, at no cost. Charging for integrations is, in our opinion, ridiculous. Why? Because vendors are, in a way,

already charging you for them in the licensing itself. The more integrations you use, the more stuff you bring in (or racks you use, or square footage you need to document, or whatever it is that counts towards the license).

In a subsequent chapter in this book, we will cover discovery and integrations in more detail.

Powerful Visualization

A picture is worth a thousand words. Your DCIM solution should have powerful visualization capabilities. If it is just a glorified database, the adoption will be more limited and eventually you will have to find a tool to supplement those shortcomings.

By visualization we don't mean a few diagrams here and there to check it off from your bucket list. You, as the end user, should be able to create any hierarchies you want, model devices and racks your way, use any icons you want, upload maps and pictures of any type, change colors and rules according to your specs, etc.

We'll talk more about this in a later chapter as well.

Securable

Considering all the stuff we hear about security breaches and the ever increasing audience for Data Center Infrastructure Management functions, you should count on a solution that offers a granular, multi-role security scheme supporting Active Directory, SSL and the whole nine yards.

KISS and Other Licensing Nightmares

You can rock and roll all night and party every day, but we would also advise you to always keep it simple. This goes for starting small, maybe with one data center or one specific problem you need to tackle but also with the licensing itself.

The worst that can happen is to acquire the tool only to find out it was missing this or that module. Licensing should be easy to understand: as in, based on rack or device count. Make sure the vendor discloses right away if there are ANY features that could be excluded.

What's Next?

We will begin discussing the specific details about DCIM frameworks and features.

BASIC VENDOR RESEARCH

In the two previous chapters of this book, we discussed the importance of discovering your organization's unique needs and the features you may need from a DCIM tool. We went on to cover identifying the core tenets of a system such as: openness, security, flexibility, visualization and so on.

Once you've got a solid grasp of what you need in a DCIM solution and its proper architecture, you can move on to finding the right tool. The technical evaluation process usually involves one or more of the following steps: basic research, demos, proofs of concept, request for proposal. In this chapter, we'll discuss the basic vendor research process and give you some tips and tricks related to that step.

Who's Out There? The Basic Research Stage

A basic research search of available tools out there will give you a bird's eye view of the vendor landscape. Read blogs, run searches on forums, LinkedIn and Google. Make your search thorough: don't make the mistake of basing your research solely upon a laundry list of vendors that some analyst firm provides. Some folks claim the whole analyst spiel is a bit of a "pay-to-play" scheme: while we will withhold our opinion on that one, needless to say, just a fraction of the vendors that exist are actually reviewed by analyst firms, let alone tested. The ones showcased are not necessarily the better ones.

A thorough search may yield over 100 different self-proclaimed DCIM vendors that do fulfill one or more DCIM functions. You should be able to narrow down your vendor list to a couple dozen – at most. Why? It will be pretty clear that if you need, for example, cable management, server temperature and power monitoring, VM discovery and customizable dashboards (to name four random features), most vendors will miss one or the other.

I have done research like this myself in previous roles: I remember researching software technologies, such as: performance management tools, root-cause analysis software and reporting engines.

What worked for me? A simple spreadsheet.

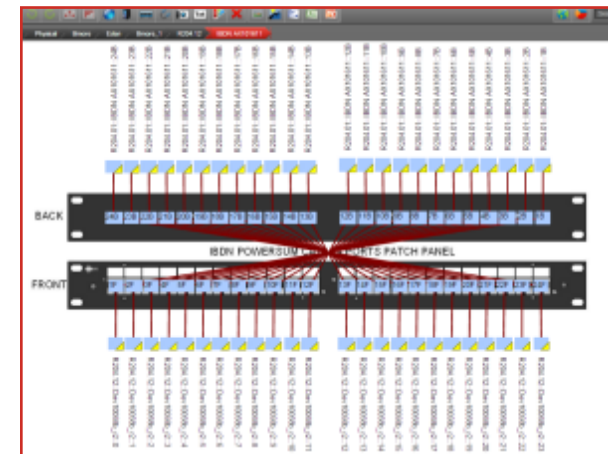
For example: create a column for each important feature and characteristic you need (see pt. 1 and pt. 2 of this book) and create a row for each vendor. Paint a cell for each vendor that, at least on the surface, fulfills a feature. As you haven't seen any demos yet or test driven any software, go ahead and make some assumptions – err on the generous side. You will cross check the features with your narrowed down list later on.

Features	10SCAPE	10-Strike	Docusnap	Graphical Networks	Manage Engine	Masshandra	Ne
Map Accuracy							
View network Devices	Yes	Yes	Yes	Yes	Yes	Yes	
View ports/interfaces	Yes	Yes	Yes	Yes	Yes	No	
View physical connections by port/interface	Yes	Yes	Yes	Yes	No	?	
View VLANS	Yes	No	Yes	Yes	No	Yes	
View virtual server and clusters	Yes	No	Yes	Yes	No	No	
Import Data							
Add/update data from manual entry	Yes	Yes	Yes	Yes	Yes	Yes	

Unfortunately, there isn't such a table publicly available (I wish some of the research firms would provide something thorough like that instead of watered down partial analysis), but in the previous page you can see how our other product, netTerrain Logical (our Network documentation tool) stacks up against some of the competitors according to a wikipedia entry for Network diagramming tools.

Analyzing the 'Vendor Credentials Show'

So, what about the vendor credentials? Every vendor will naturally play to their own advantage here. Let's be honest: the big ones will beat their chest with their sheer size while the small ones are going to claim they are more customer focused or agile or something along those lines. The old vendors will tell you that experience is the main thing; the new ones will tell you that their technology was created from the ground up and is not some hodgepodge of acquisitions and odd tools. They may all have a point and it may be valid in certain contexts: I have been on both sides of that conversation and I have heard it all.



We are a rather small company, but definitely not the smallest player. We are not old, but we were founded before DCIM really became a household acronym and before that our founding team worked for decades in the inventory and IT visualization space, so we are definitely not new either. We have no beef or affiliation in this exercise in futility.

Because, let's be frank, that's just what it is. It's funny to read some of the presentations from fellow competitors. "We are a fortune 500 software company with 20,000 employees and 700 gazillion dollars in revenues per minute". Who cares? If the DCIM tool is vaporware and the support is bad those numbers mean exactly zilch to the customer.

The following questions are the only four that matter as everything else is, in my opinion, noise:

- Does the software sing?
- Does the company consistently deliver successful implementations with their software?
- Once deployed, does the company provide good, timely training, maintenance and support?
- Will the company be around for the foreseeable future?

These questions bring up this key question: how, exactly, do you measure all of this?

First off, does the software sing? This one you can measure objectively as it's part of the demo phase. The second and third questions are hard to measure objectively, but there is enough information on the web to sort through some of this – probably your best metric is the vendor's current list of customers and references.

A vendor may have unhappy customers or customers not using the tool, especially once they pass a certain customer mark (say 100 customers), but if a DCIM vendor has their act together, most of their customers should be happy using the DCIM tool. In a recent survey we did for our DCIM product, 95% of respondents said they would recommend netTerrain to others. I guess that's a good number. Ask the vendors on your list if they have conducted similar surveys. Also, be sure to check their website to get a feel for the customers they currently have.

As for the 4th item, that one is also a bit tricky. But one may go with a very large software vendor because, just like certain banks "they are too big to fail", right?

Wrong! I have witnessed several dramas involving large

software makers killing a product (sometimes for borderline arbitrary reasons).

Not only that, large companies tend to jump on the bandwagon of a certain hyped up acronym rather late in the game by simply acquiring another smaller player. So they bought their way to the market by patching up a number of point solutions. We know how many of those solutions turn out to be. Unfortunately you may only find out when it's too late.

So you should go with the small software vendor, right?

Not so fast. Growing markets like DCIM tend to overshoot in terms of vendor count, and the failure (via bankruptcy, less-than-honorable acquisition, etc.) of many small vendors is inevitable. It is already happening in DCIM and will continue to happen as the market matures. Simply put, DCIM seems overcrowded.

If you ask me, I would avoid startups. At least in the Silicon Valley sense of “we-get-plenty-clicks-we’ll-deal-with-those-pesky-profits-later-once-we-burn-the-series-B-round-of-funding”, if you know what I mean...

Instead, look at the very basic financials:

Is the company profitable (you know, the actual objective of a company!)

- Is the company debt free?
- Does the company depend on rounds of funding to continue operating?
- Is the company growing?

How long has the company been profitable? A note on VC backed companies: it is not a plus in my book. I understand many a reader would be irked, but I'd rather go with a bootstrapped company every single time. When the founders put their own money where their mouths are, to me that speaks volumes. A VC company MAY be the real deal, but a bootstrapped, profitable company that has been around for a while, most certainly is. Building a sustainable business organically takes time, patience and doesn't happen without going through the motions. In general (at least that's our experience) it also means that the very founding team had to, at some point, perform every single duty there is without any shortcuts or gimmicks.

It's like good food: it takes time and patience to marinate it properly.

Small companies may not provide you with actual hard numbers, like revenue or profit figures, but they should be able to tell you (and even prove after signing an NDA

or something along those lines) that they run a tight ship. So, in sum, the correlation between a software maker being large and the stability of the software is virtually zero but going with a startup or a small company that doesn't have sound financials is probably not good either.

Logos, Schmogos

If you visit our website, you'll see a ton of logos in our client list. Be careful with those logos! Even our own list is not 100% representative of our DCIM install base. 4 of our 5 top customers are not there, but then: about 30% of those logos are exclusively using our other product, netTerrain Logical (about 50% use both). With a decent list like that, you should infer "ok, these guys are for real" and move on. Now if the list is too small or the vendor is too big, proceed with caution.

Why?

For large software companies, their DCIM offering may be one of 1000 products they sell, so a broad list of logos without context may not mean anything in terms of DCIM. The logo-dropping exercise may also be unfair towards new companies or very small ones. Not having too many customers doesn't mean the tool is bad. But, I do believe it could mean the software and/or process is immature. I remember our first customer. They still are

a customer (and a very happy one, managing about 3000 racks with netTerrain). They recently renewed maintenance for about the eighth time. I will always be thankful to them: they took a huge leap of faith.

Our product back then was nowhere near to what it is now. We may have been lucky as back then there were only a handful of DCIM products and DCIM as a market was basically not defined. Take into account that software is only one part of the equation. There's the installation, the deployment, customization process, maintenance process, patching, support, training, certification, user guides, API and so on. All aspects that are important and may not be visible during the software test phase. So, if you are ok trying out a new product ask yourself: do you really want to be the guinea pig?

Getting References

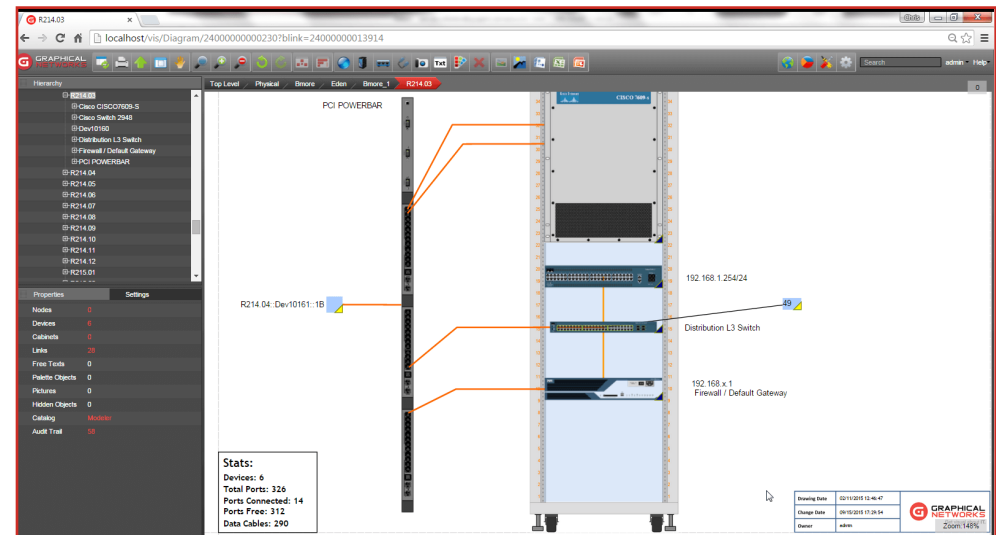
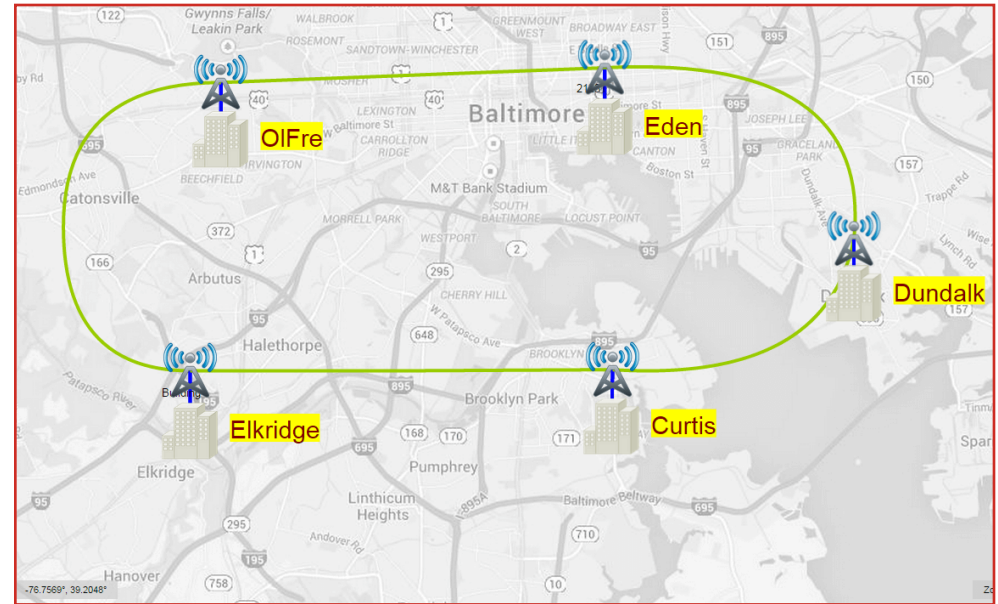
Simply glancing at the list of customers can be misleading. Dig deeper and you may find a lot of horror stories. "The tool was unusable, it took 30 minutes to create one cable". "Device models took months to model, now they are not even supporting that anymore". "Synchronizing data was impossible". "We couldn't use it because we didn't know we had to buy module XYZ.". We know: these are real quotes from real customers that came to us because their existing DCIM

implementation (in many cases from some analyst-blessed vendor) just didn't cut it. And guess what? Those customers may still be on the old vendor's logo list!

Eventually, you'll need to ask for references. We recommend doing that once you have really narrowed down your list to the final set of 2 or 3 vendor (just to save you time- and we'll cover that in the next subsequent chapter).

Next Step

Alright, now you've got your matrix, you've found out which companies provide the stuff you need (at least from some cursory research) and you've discarded the vendors that were too big (or small, or new)....now the fun part begins!In the next chapter, we will talk about the important demo stage.



TRY BEFORE YOU BUY

In parts 1 and 2 of this series, we discussed the importance of discovering your organization's unique needs and features you would need from a DCIM tool and then we talked about identifying the core tenets of a system such as openness, security, flexibility, network visualization and so on.

The Devil is in the Details

In part 3, we covered the basic vendor research process. As you discovered, this process will usually yield a great deal of vendors (which you should quickly narrow down to about a dozen or less, based upon your specific needs). In the fourth part of the series, we will talk about the important demo stage: the point in the selection process in which you get a feel for the plausible DCIM candidates by actually seeing their products in action.

I Am From Missouri, Show Me!

Once you've narrowed down your list to, say, 10 vendors, you should start requesting live demos of the product. The demo is your chance to get a bird's eye view of a DCIM tool's key features; it is the company's chance to show off what makes their solution really stand out from the crowd.

During the demo, you can get an idea of whether a

product could be a good fit and have your questions answered. It's through the demo process that you can really weed out weak candidates as you really don't want to end up putting in the time and effort of testing something that simply won't work for you. Make sure the vendor gives you a demo of the actual product – where you can also ask to show a specific feature in more detail – not just a Powerpoint presentation.

The DCIM Tool Demo is the “Show Me” Stage

Be resolute. This should be the “show me” stage. Naturally, the vendor will highlight the DCIM tool's strengths and everything usually sounds just peachy. So, be selfish: put the vendor on the hot seat and make them sweat! If you need that customizable dashboard, ask them to show it to you. It's like this: “Mr. DCIM vendor: show me in your product how you can customize the dashboard”. If you need to know how to display aggregated server inlet temperature per rack, well go ahead and ask them to show it. If you need to be able to bundle fiber strands into a fiber trunk, ask them to show you. If you need to ping, telnet, RDC or tracer devices from the DCIM tool, ask them to show it. If you need to discover and visualize VM images per host, then, well, you get the point...

Usually three things will happen when you ask about certain DCIM features:

- a) The tool can do xyz and the vendor is happy to show it to you
- b) The tool cannot do xyz and the vendor is forthcoming about it
- c) The tool can't do xyz (or a PhD is required to do it) and you hear some mumbling that resembles postmodernist babble

If the situation is b, that's fine. One thing I can guarantee you: there is no vendor that does everything that could be considered DCIM by some customer. How do I know that? We do things no other vendor does (like certain things in our discovery or our visualization) and we certainly don't do it all...But, let's also cut us vendors some slack here: certain questions may be open ended or not well defined and the vendor may provide an answer that seems vague. If you ask us how we do Impact analysis, for example, you will probably get an answer that is open ended. What's impact analysis, anyways? Impact analysis of what? How do you want it visualized? Etc.

But in general, if you run into situation c where you get a vague answer, then the lack of the feature is the least of

your worries. You've got a vendor that is trying to dupe you and it's time to run for the hills.

Try Before You Buy

Once the demo round is complete, you should have narrowed down your list to, say, 3 or 4 candidates (max). The software now needs to be tested. In fact, the things that will be a software's undoing will almost never be found on an RFP or a demo. The devil really is in the details, so we recommend you don't be shy and ask to try the software. Ideally, you should be able to test the software with little to no training and get an idea of how the software will solve your pain points as well as what the process and interface are like.

Here's our take on trying before you buy: you should only pay for a trial if the vendor will literally walk down the rack aisles and do data collection for you or if you are planning to test the software for several months in production (and usually you should only do that if you already know which vendor you will pick) – otherwise if you do manual data entry and run this in a test environment the trial should be 100% free. *Yes, free!*

If the vendor is not going to do manual data collection and you will not use the software to actually start working in production for a long period of time and they still want to charge you, I would cross them off. Maybe that's just me, as I admit I have little patience for excuses that sound very "90's". I didn't even have that

patience IN the '90's! Back then, analyzing Inventory Management systems for Inside Plant

(which, by the way, was the DCIM of yesteryear, but that's another story) I had a field day crossing off the pretenders. Ah the '90's, the decade of raves, grunge music, the internet craze and really bad enterprise software.

Let's relive some of the excuses that were thrown at you for not being able to test an enterprise software and apply them to DCIM today:

“Mr. Customer, it'll be complicated to have you test the software for free because...”

“Just setting up the system takes a week or a month”.

Lame. What on earth are you installing anyways? An entire OSS suite? Not even that should take as long nowadays. And why don't you have a SaaS version where the customer can at least get a feel for the UI, process and overall components?.

“Bringing in the customer data is complicated”.

This is just an excuse. Why don't you have a Cloud or SaaS version for testing and trial purposes ([we do](#))? If a Cloud version for testing isn't available and a DCIM

vendor gives you this excuse for not being able to give you a proof of concept, run the other way. I am sorry, but if your importing toolkit, adapters or whatever you call the process for synchronizing the DCIM with other systems and databases isn't in shape, well, that's not the customer's fault. A DCIM system should have different protocols for discovery to aid that process: it should not take that can not take as long.

“We need to model the power equipment, devices and so on that you have in place”.

Again...this should give you pause. For example, the netTerrain catalog of devices is already robust...[and you can model your own devices in seconds](#). If a DCIM vendor tries to claim modeling your equipment is too difficult and that's why you can't test out the product, I'd ghost them.

- Consider what your production environment will be so find out if the system is 3- or n-tiered (database on one server, app server on another)
- Can be both sitting on a VM?
- Can I run this in disaster recovery mode?

Get a Panoramic Picture

It's also important that life is easy for your end users. Ideally, they shouldn't require anything other than a browser. Thick clients present an extra deployment step (which grows linearly with your user base): it's another possible thing that later needs to be patched and an extra point of failure (besides other inconveniences).

Make sure you get the full picture of what other software and hardware you need to commission. When quoting the tools, if each client, for example, needs Visio, this is extra money required for third-party licenses.

So, ideally the tool should not use third-party components or as few as possible. Take a look at the manuals. Yeah, we know, you don't want to read them. Still, take a peek! It is very common for software companies not running a tight ship to be sloppy with the stuff that is hard to see during demos, such as user guides. If the vendor says they have an API, ask for the guide!

Your Time Investment Will Pay Off

Although it may be tempting to rush over one of the above steps in the race to get a DCIM solution in place, remember how important it is that the solution you choose responds to the needs your organization has. So, once you've got the attention of the vendor, make sure you have whatever hardware you need to install the product, come up with a test scenario that is simple (maybe a few racks, some basic KPIs you would like to obtain and some basic discovery and one or two integrations).

The demo and testing stage are key to finding the right tool. Ask as many questions as you need to in order to get your questions answered and allocate the necessary time for it. It will be a wise investment, because the alternative may yield an expensive piece of shelfware.



GRAPHICAL NETWORKS
814 W. DIAMOND AVE
SUITE 102

GAITHERSBURG, MD 20878 USA
+1-240-912-6223
WWW.GRAPHICALNETWORKS.COM