PacketFront Software

BECS
Solution Brief

Your Questions Answered

BECS Solution Brief

Your Questions Answered

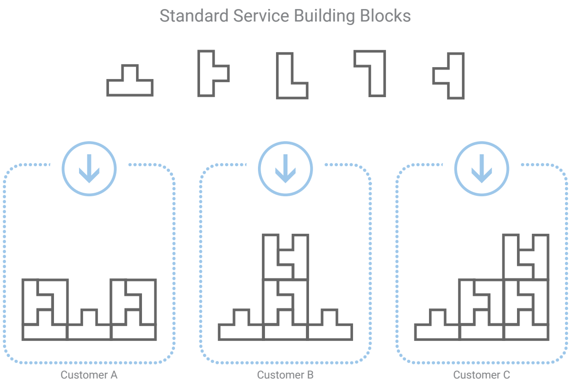
BECS takes care of all aspects of the network within the fulfillment process including design, creation, configuration and activation of services. BECS also harmonizes the network by completely removing dependencies between OSS/BSS layer and the network.

Questions answered in this document:

Question 1:

How will BECS make my life better?

If you work in an organization that uses large computer networks – and let's face it – all big organizations and corporations do, you will benefit from implementing BECS. But the benefits will look different depending on where you are in the organization.



Easily re-use the same building blocks and push configuration across connected devices

Management

If you are in higher-level management, maybe with a title that has the word "Chief" in it, you want the company machinery to work as smoothly as possible in order to spend as little money as possible to grease the wheels. BECS will give you the possibilities to increase automation and pave the way for quick and efficient scaling of your operation. If you find a way to perform a task simpler or quicker (which at the end means less expensive) you want to be able to implement this change quickly and without hassle, by the press of a button.

Engineering

If you are in engineering, BECS will simply make your life more fun. Everyone working with network management has experience performing tedious tasks, repetitive and monotonous, to implement changes in the network. Not all traditional network engineering tasks have been like this, but some have, and it's not why you chose a career in engineering. You want to use your intellectual prowess to promote thinking, the creative process that engineering really is or at least should be.

Product Management

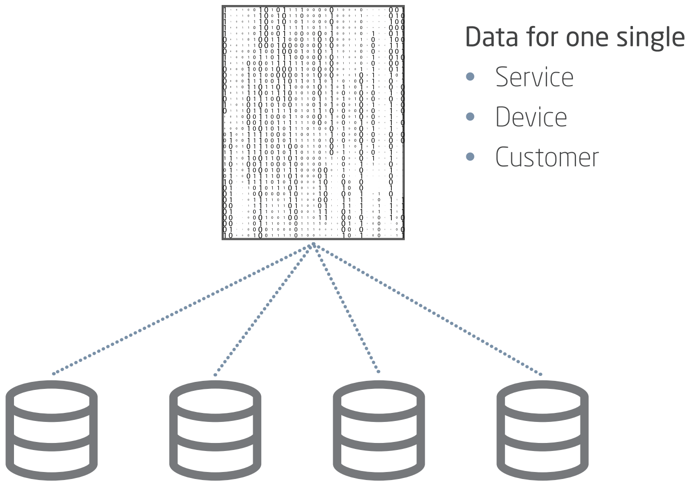
If you are in product management, you will get tools at your disposal that you never had before. If the networking components and the functions therein are structured better, modularized, and describable on a higher-level, it's possible to combine these well-defined functions to create new services and packages, without having to involve low-level configuration by the engineering team.

> *"Whenever you update a configuration for a specific element, type of element, geographical area or even which role it might have, BECS will push the changes to all affected elements, including customer premises equipment."*

Question 2:

What problems does BECS solve?

It’s difficult to run a network efficiently basically because it’s difficult to keep track of everything in it. It just is. It’s difficult to keep a constantly up-to-date documentation of a big network, and it’s even more difficult to know whether your documentation is up-to-date. BECS was designed to solve this.



Why provision the same data to multiple locations?

Inventory

You may be managing a lot of data that is simply incorrect or obsolete, using up time and resources where you shouldn’t need to. The “inventory” of a network is the list of things connecting to each other (“network nodes”), and many global operators claim that 40% of their inventory data is incorrect or corrupt in some way. That’s a lot!

Most network operators want their network to grow, and even if growing is not the main goal, change and transformation is important. You need to get with the program and follow the trends in your business, whichever business that may be. But what if you don’t know exactly where in your network services X, Y and Z are currently running?

What if service X exists in 30 slightly different versions simply because it has been assigned to customers 30 times without re-using any configuration? Steering this ship towards a new direction may prove extremely difficult, or even impossible.

This should come as no big surprise to you who are reading this, as you are a human being, and the essence of the problem is that we humans are not very good at meticulously keeping perfect order in super complicated systems, and continuously documenting the current state of every single part of it. This is clearly a job for machines. A job for BECS.

Finally, to launch a service today, or even just a new node, you may need to provision data to not one but several support systems. You have your physical network, where the service will run, but then you also have inventory systems, databases, element managers and more. Your service will need actions taken in all these systems, often with the same data being provisioned. As these support systems don’t interact with each other very well, they are often referred to as “silos”. A big operator declared, for instance, that for every dollar they spend on hardware, they spend 7 dollars managing it.

We don’t want silos, of course. We want to provision a new service or node in only one location, and this is also something that BECS helps us with.

Question 3:

What makes the BECS approach so good?

Why are we humans not good at keeping perfect order in a complex system? Well, evolution has made us very creative, flexible and adaptive, but really bad at doing the same job 100 times in a row without making a single mistake. We have to accept this fact about ourselves.

Which of these meals
is easiest to mass produce?

Mass Production

Look at any fast food franchise that scales really well (i.e." seems to be taking over the world"). How much manual thought has to go into every making of a hamburger/sandwich/pancake/taco? Not very much, is a likely answer. The employees get machines that are then filled with ingredients, that then create the consumer product at the press of a few buttons.

Compare this with an á la carte restaurant, where a chef (or even a team) painstakingly produce a product specifically designed for you and you alone. It's not a surprise which method costs more and which one scales the best.

Adjusting to Change

The operator industry has to move towards fast food automation in order to survive in an evermore quickly-changing networking landscape. An operator can't just deliver a one-size-fits-all hamburger-like service to all its customers, though. Customers want to feel in charge of what they pay for and be able to customize their service in as detailed a way as possible. In other words: **you want to deliver á la carte services but with fast food automation behind the scenes**, and this is where BECS enters the stage.

BECS helps you automate all changes and transformations, without being prone to human errors, but it is also at the same time the control dashboard and the documentation. When you implement a change it immediately affects the description of your network for anyone else to see.

Work like an Architect

The IT industry has already gone through this process. It is now more common to buy an easily-copied virtual machine somewhere in the cloud to run my service on, than buying a "real" dedicated server in a data center for me and me alone. And it's even further from buying the server yourself and having a server room of your own! It's very scalable and it's based on easily modularized services and functions that can be copied and re-used over and over again.

Here's what BECS does: it helps you think like an architect instead of a carpenter. A carpenter gets called in when there is a problem, and solves it as best he or she can to make the symptoms go away. Maybe the solution creates more problems down the road, but that's a problem for later.

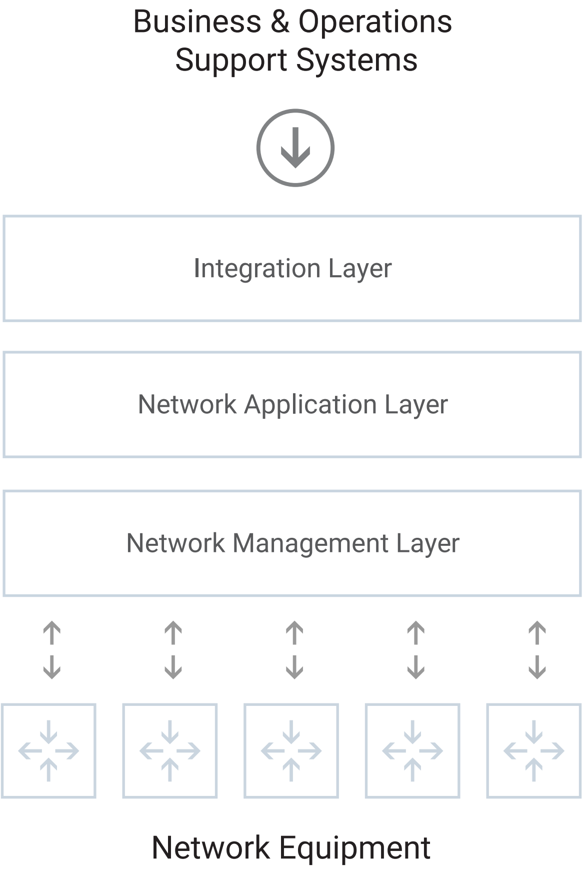
An architect thinks beforehand and designs a platform stable enough to avoid problems in the future, but which is also flexible enough to cater for solutions to problems that may arise, but in a standardized and easily documented way.

Work like an architect

Question 4:

So what is BECS, really?

BECS is computer software that helps you manage your network. It runs on servers, just like any software, with all the optional high availability and redundancy that can be applied to any network function.



Business & Operations Support Systems
Integration Layer
Network Application Layer
Network Management Layer
Network Equipment

The Sandwich Approach

It is essentially a three-layered sandwich with a top, a bottom and a middle part. And just like with a sandwich the meat is really in the middle part. The top and bottom are to talk to the outside, while the middle is the internal logic and brains of BECS.

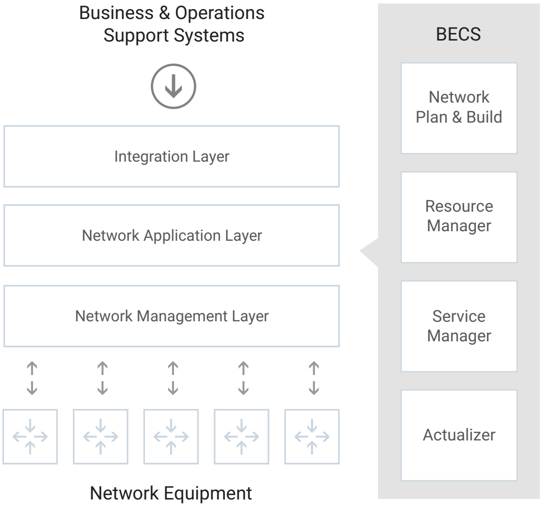
The bottom is to talk to the network nodes (switches, routers, firewalls etc.) and tell them what to do. This is called the "Network Management Layer".

The top part is to talk to other systems in the customer organization, for instance BSS/OSS systems. This is called the "Integration Layer".

Between them sits the "Network Application Layer" that is the heart of BECS.

Plan and Build Networks

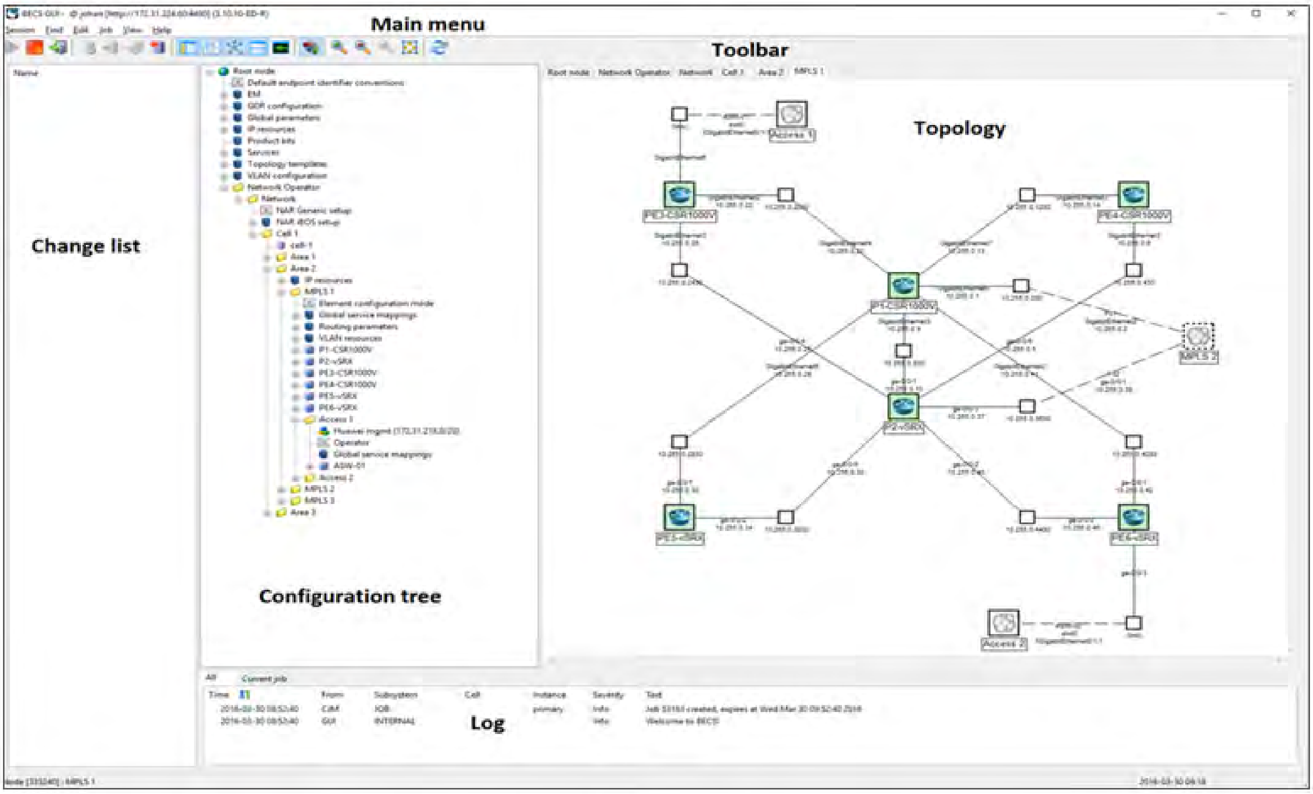
It is the Network Application Layer that lets you plan and build networks that didn't exist before. It enables management of resources in the network as well as services offered using these resources. A resource can be a physical node, but it can also be a resource such as an IP address. The Actualizer is the part where the different behaviors of different kinds of network nodes is handled using prepared models or programmed logic for particular customer nodes.



Question 5:

And what does BECS look like?

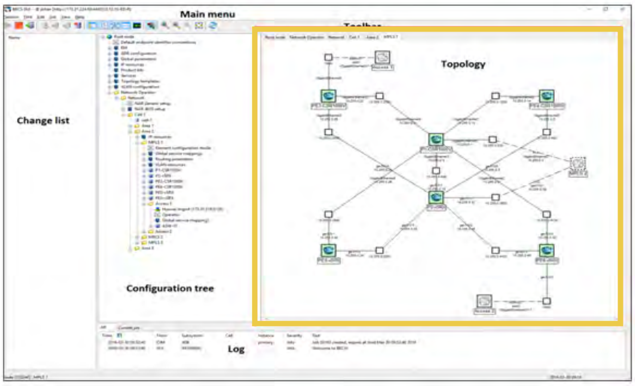
BECS has a graphical interface showing network details or aggregated overview images in an easily digestible way.



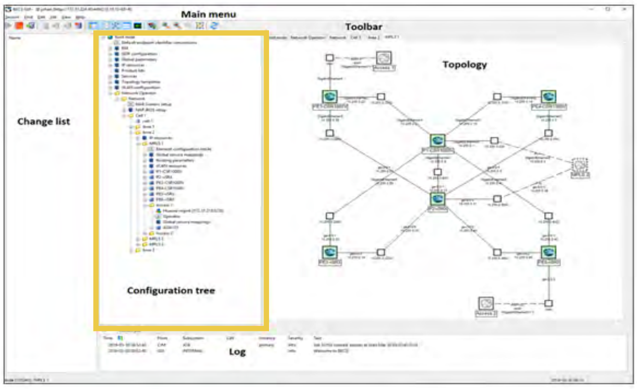
```
Time                 From  Subsystem  Cell  Instance  Severity  Text
2016-03-30 08:52:40  CAM   JOB              primary   Info      Job 33161 created, expires at Wed Mar 30 09:52:40 2016
2016-03-30 08:52:40  GUI   INTERNAL                   Info      Welcome to BECS!
```

The actual user interface of BECS

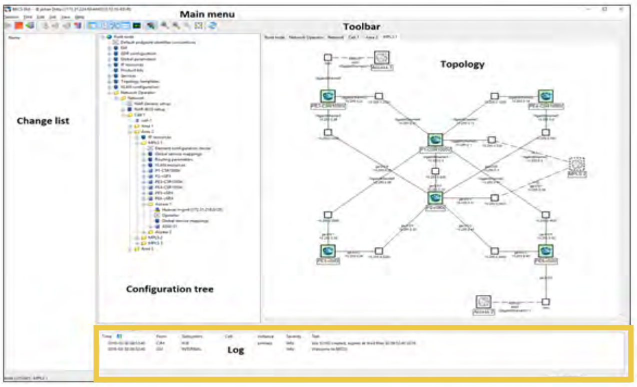
The BECS graphical user interface window has four panes, or views, that are used to display different types of network information, on the following page are some examples.



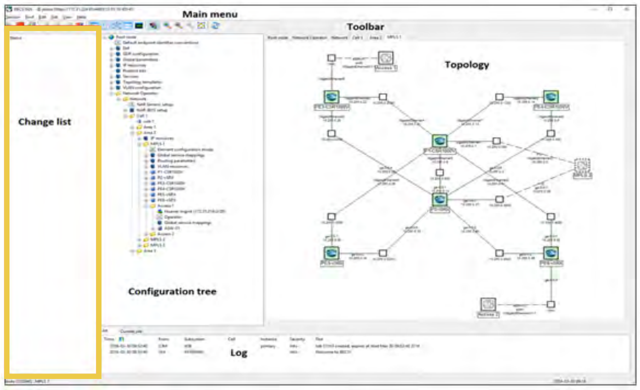
Topology is a graphical overview of the network, it is less detailed than the configuration tree, but the graphical representation of the network provides an overview that is easier to monitor.



The configuration tree is an abstracted view of the network topology and its configuration. It represents the hierarchy in which all networks at scale are built.



There is a Log at the bottom showing system jobs, debugging log and other logs based on events in the network.



The Change List on the left side of the screen lists all the currently uncommitted changes. The magic of this may need to be highlighted a little bit. Since BECS objects are interconnected, a change in one object often results in changes in others. If you change one single thing, the change list shows you the exact and complete effects of your action. You don't need to worry about unforeseen side effects of your new configuration, and if you don't like some of these side effects in the change list, you can simply rollback before the changes are committed and sent out to the actual network.

Question 6:

What makes BECS unique?

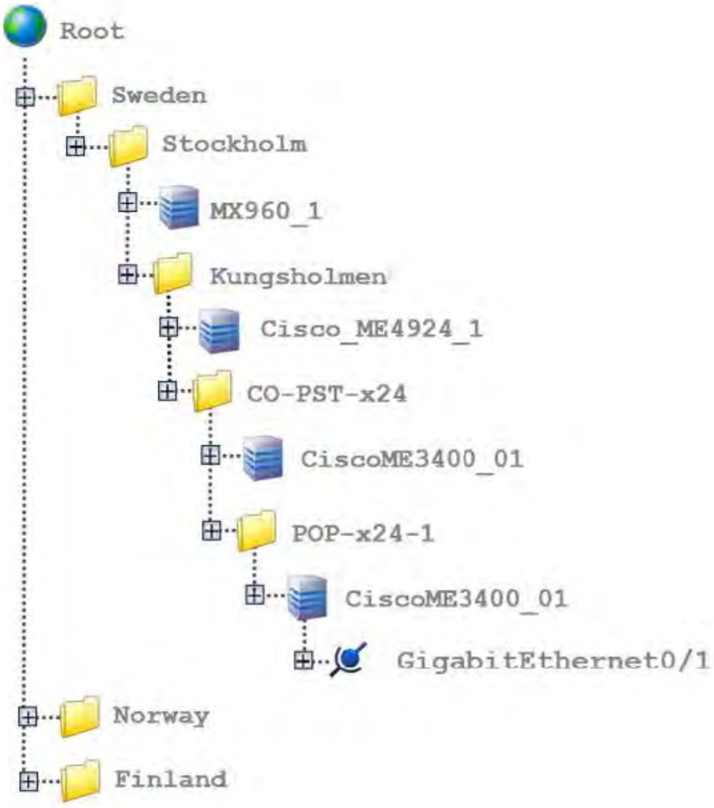
Do you remember what BECS is? It was the three-layered sandwich with all the meat in the middle, right? Also, do you remember that in the BECS user interface you get to work with a tree-like representation of your whole network?

Where in the sandwich do you think that tree is? It's in the middle, of course!

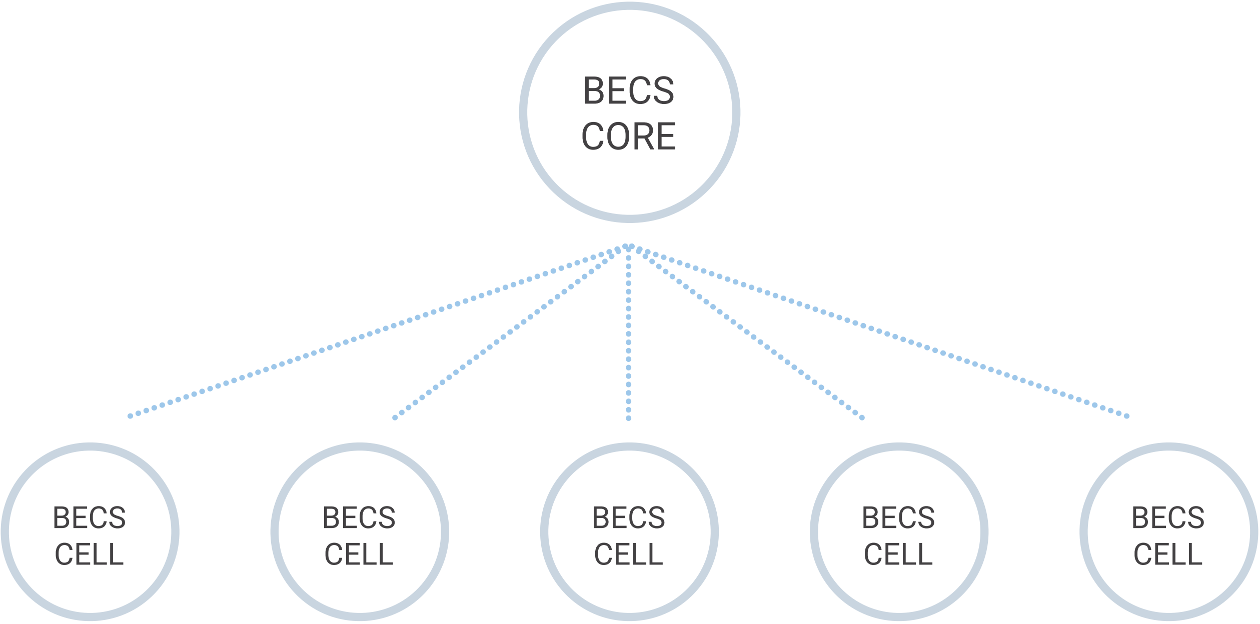
In the Middle of Your Network

Being in the middle of the configuration is what makes the manipulation and restructuring of your network so easy. So, while there are other products on the market that assume the role of central "network controller" or "network manager", the actual logic for how to control or manage the network is just not there when you unpack the box. Any business logic you would like to add would have to be added, programmed, manually. As a contrast, BECS comes right off the shelf pre-packed with useful functions operating on the tree, using any combination of the available variables (e.g. roles, parameters, element groups).

The BECS tree is easy to understand and is intuitive to engineers, product managers and managers alike, depending on what level of detail you look at.



An example of a BECS configuration tree



BECS can be deployed in a hierarchy to handle massive networks

The BECS Tree of Control

In the example picture you start at a level where you only see which Nordic countries you are managing, but can then drill down to a single CPE at a particular customer site.

When you add a new node it immediately shows itself in the tree, bringing the network documentation to a constantly updated state.

As you can include resources such as IP addresses in the tree as well, this will simultaneously be a documentation of your entire IP subnetting plan!

Only PacketFront gives you this tree view of your network. Drawing a network can be difficult with only a handful of nodes connected. Try drawing one with 350,000 nodes! In the tree view you can easily find resources in a way that is as easy as finding a file on a computer for anyone who has every used a treebased file browser in Windows or Mac.

The tree view does not only give a comprehensive view of how the network is structured, but also of how it is managed by BECS. Objects' locations on different levels and in different branches, is an essential aspect of how they will receive services and configuration.

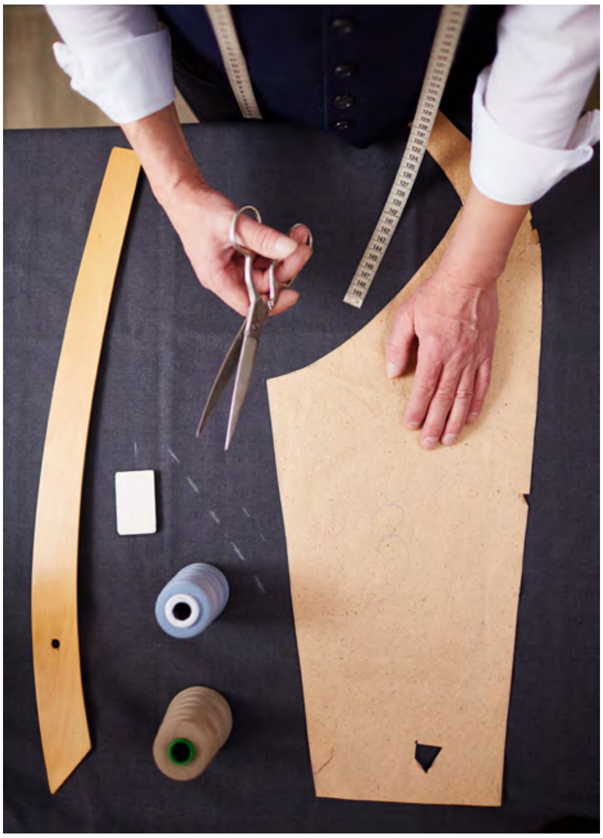
Networks can be built in this tree structure based on many criteria such as region, types of elements, roles of elements and so on. It determines how the network gets configured. This gives an overall control of the network but also gives arbitrarily fine granularity when needed.

For large deployments, BECS itself can be deployed in a distributed manner with central definitions and databases in one location (the BECS Core) and local installations to handle certain parts of the network (BECS Cells).

Question 7:

How will I actually use BECS?

As a PacketFront customer you don't need to do any programming if you don't want to.



Just like with a good tailor, all you need to do is look good

Flexible Network Management

The heart of BECS is essentially a flexible network management core which is what all network operators want. BECS comes with support for a multitude of makes and models of network equipment on the market. If that is not enough PacketFront will create the network management packages for you to be able to manage 100% of your network should you want to.

And then you may want to integrate BECS with your existing support systems such as BSS and OSS. Again, PacketFront will step up to the plate and help you with this.

Tailor-made for You

BECS simplifies and shortens the integration needs in the OSS and BSS layer by harmonizing services. This is done by creating a single service per service type for the whole network, regardless of vendor or technology. This means that you don't have to change anything in the OSS/BSS layer for your service activation when introducing a new hardware vendor in your network. Simply put, "You will be a consumer of BECS, and PacketFront will make all personal adjustments". Just like with a good tailor, you just wear the suit PacketFront makes for you.

Question 8:

How do I get from where I am to being a BECS user?

Excellent question, thanks for asking! Nothing in the natural world changes overnight. Humans have come a long way since being fish, but it took quite some time. Similarly, we are currently witnessing the transformation from IPv4 to IPv6, from dedicated servers to virtual machines, from stand-alone network devices to software-defined networking.

Start Small

Things take time to change and we must let network management transformation take its time too.

This is why you can start using BECS on as small a scale as you want to. Begin by letting BECS just manage a small aspect of one kind of node, for instance. Then you can increase the responsibility by letting it control the whole node, or maybe the small aspect, but for other kinds of nodes. The point is: grow organically and not forced. Take baby steps when needed.

Stop Patching

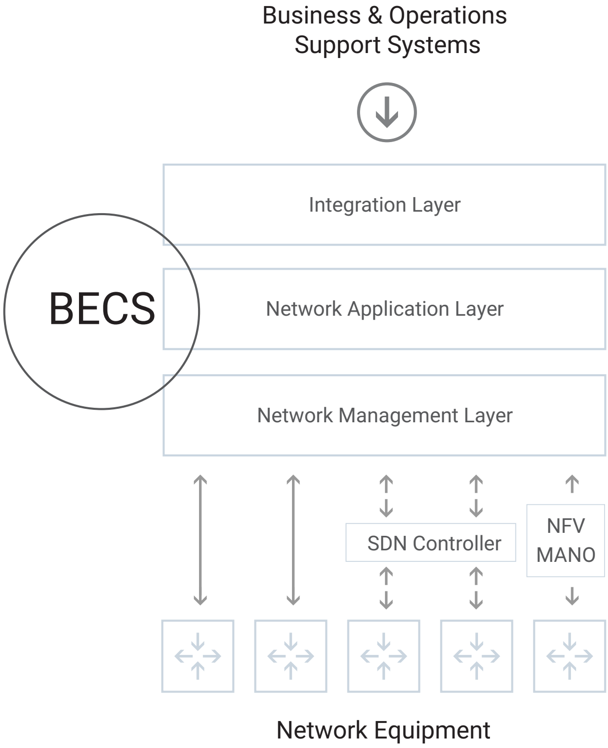
Because let's be realistic: if you want all or nothing, and you're waiting for a miracle that will clear up your whole network management situation immediately, you're in for a long wait.

It's also a common situation to be so focused on patching and patching again the current way of network management (whatever that may be), that you don't find any time to actually sit down and come up with a solution that will hold water over time. While being full of respect for your plate already being kind of full (we are network engineers too), it's just not historically a good method of reaching long-term success.

Question 9:

What about all the SDN/NFV buzzwords?

We are fully buzzword-compliant, but not because we have read the latest standards and are quick to follow, but because we are ahead of the curve.



We Were Born Ready

First, some background. SDN (Software-Defined Networking) and NFV (Network Function Virtualization), are currently ongoing revolutions in the Datacom and Telecom industries. SDN separates the control plane from the traffic-forwarding plane in network equipment in a way that is very far from how networks have historically been built. NFV enables network applications that historically ran on dedicated and special-made hardware to instead run as "virtual applications" on standard, off-the-shelf, hardware. Both of these changes alter the very foundation of network design and management.

It's a revolution much like the revolution from circuit-switched to IP-based communications and in comparison, the IPv4-to-IPv6 change is a minor upgrade next to SDN/NFV. But no revolutionary changes happen overnight. We didn't have an IPv4-to-IPv6 switch to flip, but instead had to come up with clever migratory processes. These processes work but are often slow and costly, and since we don't have a go-to-SDN switch to flip either, figuring out the most effective way to do it beforehand is a very worthwhile investment. BECS is the key to your transition into SDN/NFV.

As BECS is specifically designed to be able to handle any kind of networking equipment while hiding the differences to the network management (OSS/BSS), it can handle SDN controllers with the same ease as individual switches, routers or firewalls. This means that you can manage a network with any mix of legacy and SDN-based equipment. You may start with 0% SDN, and then gradually push that number up to 25% over a year, and then to 50% over the next year, and then so on, finally ending up near 100% SDN.

You may have equipment that you for some reason don't want migrated to being SDN-controlled, thus leaving you at maybe 98% after five years, and that's perfectly fine too. This is the intrinsic beauty of BECS: being able to securely and robustly handle different kinds of networking equipment from a consistent and simple management interface. Legacy or SDN-based, vendor A or vendor B, it doesn't matter.

Easy Transition

In summary: BECS will help you transition into an SDN/NFV world, not because we were quick to follow when SDN/NFV started being the talk of the town, but because when the talk started we had already built a platform that can handle revolutionary changes, including this one.

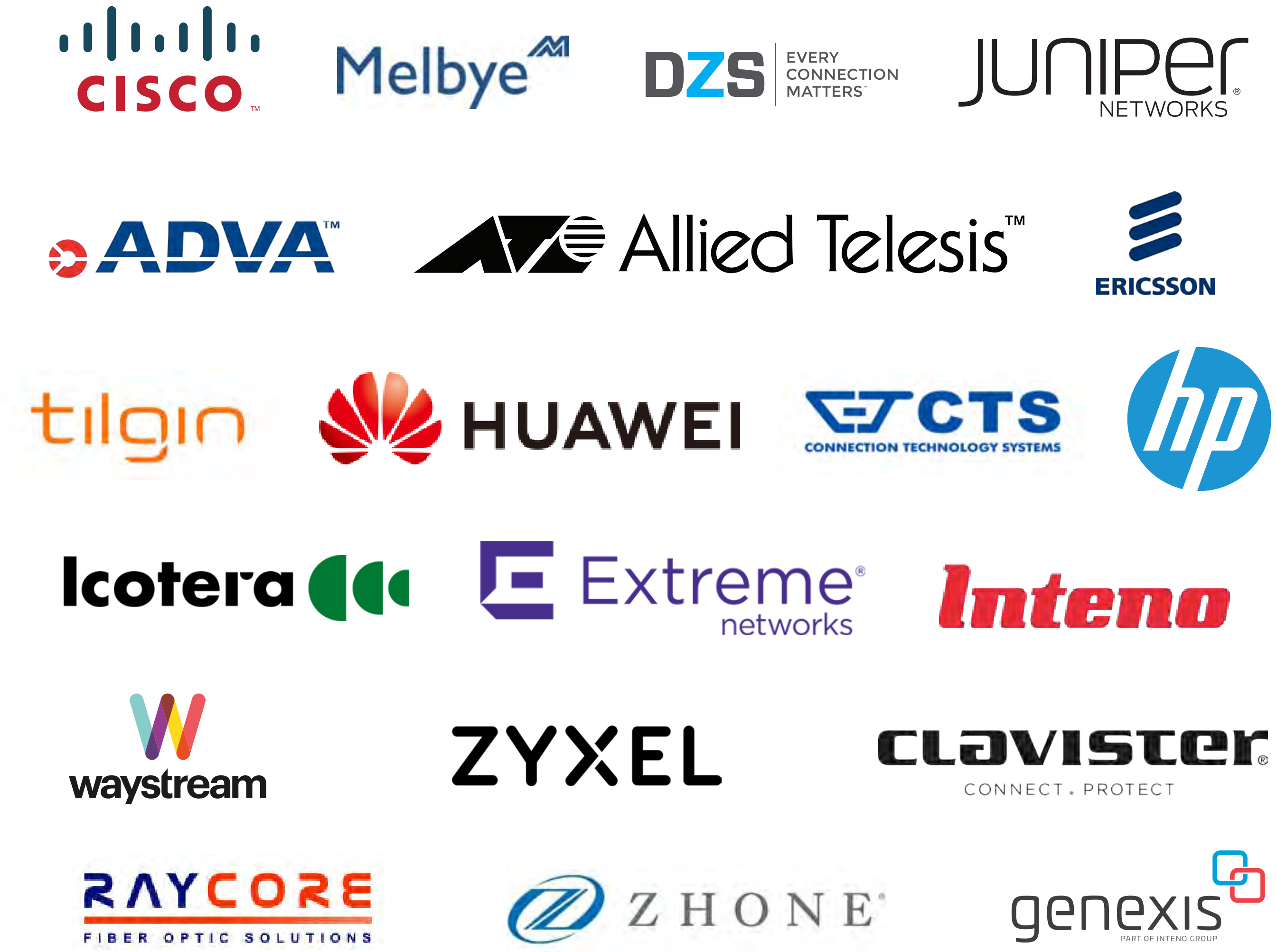
Question 10:

Which vendors is BECS compatible with?

All of them. Easy answer, right?

A more detailed answer could be: we can manage any network equipment that has a network-accessible API or CLI towards the outside.
As we create network management drivers on-demand when you need them, we will never be unable to manage any part of your network that is manageable in any way.

And as we constantly add to our library of these drivers, we already have quite a selection of support for most of the equipment on the market right out of the box.


CISCO
Melbye
DZS EVERY CONNECTION MATTERS
JUNIPER NETWORKS
ADVA
Allied Telesis
ERICSSON
tilgin
HUAWEI
CTS CONNECTION TECHNOLOGY SYSTEMS
hp
Icotera
Extreme networks
Inteno
waystream
ZYXEL
clavister CONNECT . PROTECT
RAYCORE FIBER OPTIC SOLUTIONS
ZHONE
genexis PART OF INTENO GROUP

Question 11:

Is there a way to summarize all these answers?

There is!

Summary

Let's face it: large networks are complex and running one is fraught with obstacles and risks of all sorts. You could have interoperability problems in a multi-vendor deployment, you could have difficulties upgrading and changing an all-too-static network, and if you do change things you risk ending up with obsolete documentation. Just to name a few possible ways things can go wrong.

And let's face it a second time: when customers choose operators to buy network services from today, those operators are offering increasingly similar services. If you buy a cell phone subscription, how many parameters do you compare? "Only price" is not an uncommon answer. You could add data plans, bundled services and other things, but a lot of those things really just translate into price as well.

This means that the operators' competitiveness comes primarily from how effectively they can run their network. With how much efficiency is the available computing power, network bandwidth and storage used? To how great a degree can the operator keep using the hardware already invested in, without having to invest in more?

BECS is the magic wand that lets you as an operator utilize your investments fully, lowering both CAPEX and OPEX.

You won't have unused, and unseen, resources lurking in undocumented or illdocumented parts of your network. You will be able to buy the best and most price-effective network equipment without having to worry about keeping all your shopping with one vendor. The people that used to do the manual configuration of your network don't risk committing human errors, and they are now free to work with business-improving projects and innovation instead.

Come to think of it, there is actually one thing apart from price that the customers are interested in, and BECS improves this too: Customer Service. When the customer calls and wants to change something in their service, which do you think they prefer: that change happening in a couple of days or in a couple of minutes? With the automation offered by BECS, no service changes need to take more than minutes or even seconds any more.

Key Benefits in Short

In short using BECS you will be able to:

- Manage huge networks with minimal risk of human errors
- Easily scale and transform your networks
- Have a constantly up-to-date documentation of your whole network
- Make more money
- Have more fun

And you can get there one small and easily manageable step at a time.

You just need to take the first one.