

# Solution overview

## BECS BNG Solution

## Contents

1 Overview.....	2
2 Network Plan and Build.....	2
3 Automated Service Provisioning .....	3
4 Real-time Network Control and Documentation .....	4
5 Integration to other OSS/BSS Systems.....	5

## 1 Overview

BECS is a network and service management software designed to complement the functionalities offered by BNG nodes to create a fully automated end-to-end service delivery in multi-vendor environment. The solution meets the high technical requirements of a multiservice network and at the same time reduces load on technical staff, thereby reducing cost. The solution simplifies and streamlines configuration of network elements, provisioning of services, management and documentation of the network. It targets operators that:

- Need to **increase operational efficiency** through network abstraction
- Want to have **full control of the service provisioning** chain
- Value a **completely documented network** that is automatically updated and cleaned-up in real-time at each network or service change.

The network can be harmonized to a point where a customer care/sales organization can provision and support services end to end, without any dependencies from either the technical department or the network.

## 2 Network Plan and Build

The Network Plan and Build functionality of BECS enables operators to define the network topology and element configuration in a structured way in advance.

The network engineer builds the network in BECS using pre-defined element- topology- and resource- templates:

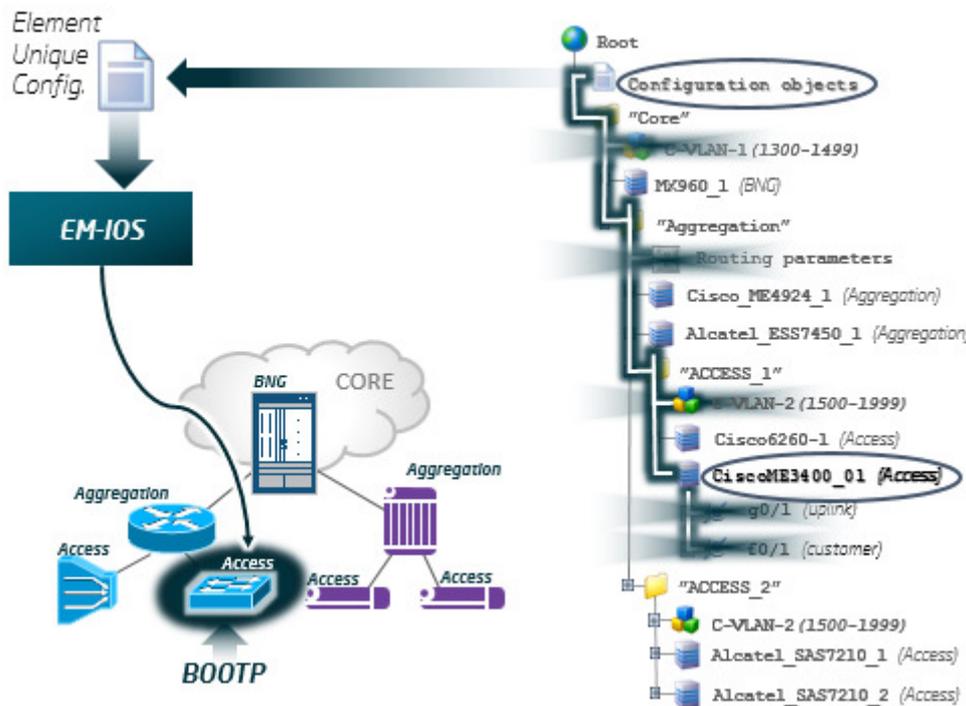
- Element templates define vendor, chassis, models, line-card settings, type and number of interfaces etc.
- Topology templates (L2, L3, aggregation, MPLS, and more) structure how elements are connected.
- Resource templates define how and what IP, VLAN, Routing, etc. are allocated for these topologies.

PacketFront Software provides standard templates as a part of the supported product kits. These standard templates can be adapted to support the exact configuration used in the network. In addition the scripting function allows the network owner to create wizards for adding entire rings or POPs with full configuration.

The result of the network planning can be utilized to automatically configure the network elements as they are connected to network. This zero-touch deployment and configuration is usually most valuable when installing CPEs and access elements. For example, BECS can configure the customer VLAN's between the BNG node and access upon element installation.

The first step is that the engineer adds an element to BECS from the BECS GUI. When the element is connected to the network, BECS will make element configuration:

- BECS will catch a BOOTP request from the element.
- BECS then traverses the configuration tree picking up the correct element configuration, such as VLAN id and IP addresses on the way creating a unique element configuration.
- The configuration is sent to the element by using a vendor specific BECS Element Manager.
- If the element needs new firmware, BECS will also trigger a firmware upgrade.



If an element does not support BOOTP, it needs to be pre-configured with IP configuration before the element is connected to the network.

### 3 Automated Service Provisioning

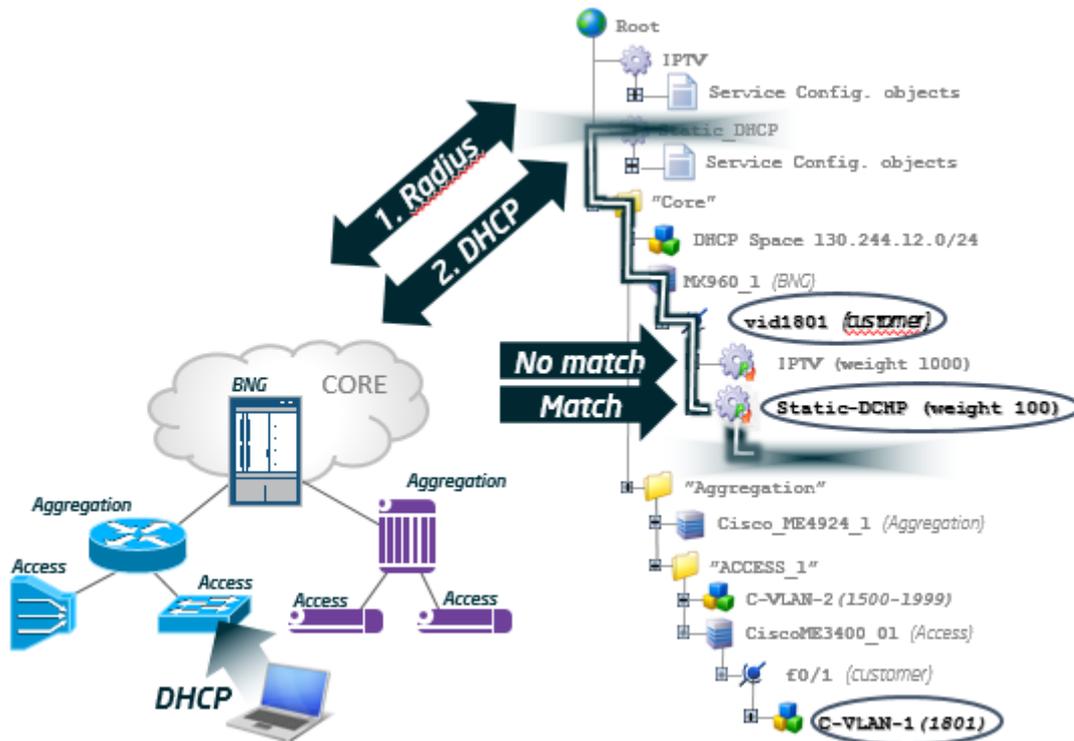
One of the main functions of the BECS is the service provisioning. This chapter describes the full provisioning capabilities of BECS, but it can be implemented in different steps and extent depending on the customer requirements and network conditions.

BECS has advanced functions for defining and provisioning services. A CRM system only needs to send an activation request with two parameters: "Requested service" and "Delivered to specific access port." BECS then resolves what should be done and where in the network, without any technical input from the OSS/BSS layer. This could range from just changing a parameter in a service profile to major reconfiguration of the entire network.

As BECS is network topology aware, it has complete view of the service delivery path including the logical interface in the BNG node, access port and CPE. This means that even if the service is provisioned on the access port level, BECS can do the service provisioning on the logical port of the BNG node.

When a new client connects to the network, BECS executes service provisioning by pointing out the correct service template in the BNG node by using the in-built Radius server or by reconfiguring the interface according to the provisioned service.

When a client connects to the network, the DHCP request is relayed from the BNG node to the DHCP server integrated in BECS. After controlling the service profile, BECS assigns an IP address from a correct address pool enabling the service activation.



In addition to configuring the BNG node, the solution can be further automated to any desired level, where a service provisioning task can be extended to cover multiple elements, topologies, and technologies. The generic configuration is automatically translated into element- and vendor-specific configuration formats, all the way down to the access layer, and can provision sub-services on CPEs, if necessary, in order to ensure actual end-to-end delivery.

#### 4 Real-time Network Control and Documentation

Full documentation of the network is an integral part of BECS. Every element and its configuration as well as every service activation are documented in the system itself, and linked to the actual network in real-time.

When a BECS operator (or an external system) changes anything in the network or activates/deactivates a service, BECS will update the configuration for all elements managed by BECS and documents the changes for elements not in BECS control. Version management enables roll-back to older configuration versions, from a single port to the entire network level. BECS instantly cleans-up unused configuration for all affected elements and CPEs in the service delivery path and IP addresses, VLANs and other resources are released as soon as they are not used.

## 5 Integration to other OSS/BSS Systems

BECS is designed for integration with business support systems and other OSS functions. Every function in BECS is represented by an API-call published on the external API. The external API is a low level API based on SOAP/XML.

On top of the external API, a number of API Adapters publishes specialized high-level APIs. One example of such API adapter is SPAPI which provides simple (service, port) yet powerful calls for service provisioning and management. Each API adapters provide API-calls for a specific purpose that abstract long chain of BECS functions into single non-technical API-calls. This assures for efficient integration with a minimum of API calls between the OSS/BSS application and BECS. It also removes the need for the OSS/BSS application to keep any knowledge about BECS or the network technology or topology.

BECS integration layer contains a script engine that can be used to add logic between BECS and external systems. Scripts can for example be used by an API adapter to execute a series of commands within one single call from an external system, or to interact with external systems.

It is possible to extend the API and the API adapters with new API-calls. This is particularly useful for integration with legacy systems that might have limited and very specific integration capabilities.

BECS also comes with an optional workflow engine that can define, schedule, control and track processes in and between systems and organizations. The built-in scripting capabilities can be used to automate parts of these processes which will minimize errors and speed up lead time between order and service delivery.