Transforming Campus Networks to Intent-Based Networking

Enabling Your Network for the Future

ciscopress.com

Pieter-Jan Nefkens

Transforming Campus Networks to Intent-Based Networking

Pieter-Jan Nefkens

Cisco Press

221 River Street

Hoboken, NJ 07030 USA

Transforming Campus Networks to Intent-Based Networking

Pieter-Jan Nefkens

Published by:

Cisco Press
221 River Street
Hoboken, NJ 07030 USA

ScoutAutomatedPrintCode

Library of Congress Control Number: 2019914065

ISBN-13: 978-0-13-546633-9

ISBN-10: 0-13-546633-4

Credits

Figure 2-3a	Shutterstock
Figure 2-3b	Shutterstock
Figure 3-1	Courtesy of The Open Group
Figure 6-11	Screenshot of NetBrain Technologies © NetBrain Technologies, Inc
Figure 9-2	Courtesy of Everett M. “Ev” Rogers
Figure 10-2	Screenshot of iPhone app © 2019 Apple Inc.

Contents at a Glance

Contents

About the Author

Pieter-Jan Nefkens is a long-term Dutch-based IT and network consultant. From early on in his career he has been connecting devices and people, even before the Internet era. Pieter-Jan started his career immediately as an entrepreneur in IT with expertise in networking, security, virtualization, and active software development. Throughout his 20+ years of experience as a consultant, he has always been on top of new trends and technologies and applied them through implementation projects and consultancy to solve specific business problems of his customers, varying from small companies to large international operating enterprises. Pieter-Jan firmly believes that you can only consult and apply technologies if you have used them yourself. Pieter-Jan has always had a strong and close relationship with Cisco since the start of his career, resulting in his participation in beta tests and early field trials, often being one of the first to deploy a new network technology.

Sharing and applying new technology is in the DNA of Pieter-Jan, which resulted in his becoming a Cisco Champion since 2017. Besides networking consultancy, Pieter-Jan also participated in standardization processes for inland shipping across Europe.

Over the past years, Pieter-Jan has been working for both the Dutch government as well as his own consultancy company.

About the Technical Reviewers

Denise Donohue has worked with information systems since the mid-1990s, and network architecture since 2004. During that time she has worked with a wide range of networks, private and public, of all sizes, across most industries. Her focus is on aligning business and technology. Denise has authored several Cisco Press books and frequently shares her knowledge in webinars and seminars, and at conferences. She holds CCIE #9566 in Routing and Switching.

Shawn Wargo is a Principal Engineer of Technical Marketing (PTME) for the Cisco Systems Enterprise Product Marketing team. Shawn has been with Cisco since 1999, and worked in both TAC and Engineering, before becoming a TME in 2010. Shawn primarily focuses on Catalyst multi-layer switching products, with special emphasis on next-generation hardware (for example, Catalyst 9000) and software products (for example, Cisco SD-Access).

Dedications

I would like to dedicate this book to my late father, Piet, whose guidance and counsel I still miss, and my lovely and wonderful partner, Renate, who has been a great support for me throughout the complete process from idea to actual writing of this book.

Acknowledgments

I would like to thank and acknowledge several people who have helped me directly or indirectly with the necessary skills that enabled me to write this book.

First of all, I would like to thank my partner, Renate, and our two beautiful daughters for believing in me, accepting me for who I am, and supporting me when things got a bit more challenging.

I would also like to thank my parents for allowing me to follow my dreams and to provide me with the opportunities to learn skills and gain knowledge. I would specifically like to thank my late father, Piet Nefkens; Jos van Splunder; and Lex de Lijster for teaching and coaching me that consultancy in IT is only successful if you first understand the business and then apply technology to solve a specific problem.

A big thank you goes out to Patrick Nefkens for reviewing my rough writing material and providing honest and candid feedback. It helped me stay on track.

I would also like to thank Dick van den Heuvel for providing me the opportunity to start working with DNA Center and help the team along with automation.

A thank you also goes out to Brett Bartow, Chris Cleveland, and everybody else at Cisco Press for believing in this new concept of merging process and organizational aspects with technologies in a single Cisco Press book.

I would also like to thank Brett Shore, Lauren Friedman, and Andrea Fisher Bardeau for running the Cisco Champions program and allowing me into this program, and—via the program and meeting Brett Bartow—making this book possible.

And, well, a very big and warm thank you goes out to all Cisco Champions. I am honored to be part of that warm community of experts on networking for some years now. All feedback, opinions, good recipes, and meetups have always been great and fun.

Also, a special acknowledgment goes out to the Piano Guys. Their music allowed me to get into the necessary flow and focus for writing this book.

Finally, I would like to thank my technical reviewers Shawn Wargo and Denise Donohue for their patience, commitment, and support in the adventure of writing my first book.

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([ ]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

Introduction

Intent-Based Networking (IBN) is the next revolution in networking that Cisco, Juniper, Gartner, and others are conveying. Cisco explains the concept with the Network Intuitive communication and solutions such as Cisco DNA Center, Software Defined Access, and Cisco SD-WAN. But IBN is much more than just a combination of those technologies and solutions. It is also a concept on how a modern network infrastructure should be designed, managed, and operated, leveraging Cisco Digital Network Architecture as the foundation.

And although the concept of IBN is accepted by the industry as the next generation of network infrastructures, many IT specialists and organizations face the challenge of how to design and transform network infrastructures and network operation teams into IBN, specifically for existing environments. This challenge is mostly seen with questions like "Yes, I do understand about IBN but how do I get started?"; "Now that I installed Cisco DNA Center, what can I do with it?"; or "How can IBN help me in providing services faster to my internal users?"

This book is written as a compendium to that challenge and its related questions, specifically focusing on campus enterprise networks. This book provides a detailed explanation of IBN, specifically related to campus networks. With that background information, this book documents a unique four-phase approach to answer the question of how an organization can get started on the transformation (or journey as some call it) to IBN for existing campus networks.

As IBN requires changes in both technologies (and how they are applied) and organizations (how networks are managed), this book also provides tips on how change can be realized throughout an organization. This book should be of help and support for anybody who wants to transform their network into an Intent-Based Network.

A large part of my career as an engineer and consultant has been focused on enabling change and making change happen. The change always involved technology in one way or the other, primarily with use cases where technology would solve today's or tomorrow's problems, or sometimes the technology would open up new innovative ideas and concepts. As an external specialist, my role was always to help and support the organization with the change of work.

Intent-Based Networking will bring very interesting times of change to the networking industry in general. The network, specifically the campus, will play an important role in the future of any organization. The rate of change is increasing too, which I see in my day-to-day work as well. The ability to deploy intents in this manner is for me only the first step. What would be the next step when you can do this? The opportunities are really unlimited.

I have used my personal experiences and observations throughout this book together with the concepts of Cisco DNA and IBN to support you on your journey to the next generation of networks.

I do hope that reading this book provides you with enough information and background on why and how to transform an existing campus and network operations team to the concept of Intent-Based Networking.

This book also has two appendixes that are used to provide you with conceptual background information on the technologies named throughout this book as well as reference configurations for the underlay of an Intent-Based Network.

Who Should Read This Book?

This book is written for network consultants, network architects (designers), senior network engineers, and IT managers who have any questions related to IBN and how existing campus networks can be transformed to IBN. A background in networking is helpful when reading this book, but a deep technical understanding is not required as each technology is explained at a conceptual level in an appendix.

How This Book Is Organized

This book covers a diverse set of topics related to the successful transformation of an existing network to IBN and is divided into three logical parts that are best read sequentially:

Part I, "Overview of Intent-Based Networking," provides you with background information related to campus networks and the concept of Intent-Based Networking (IBN). This part contains a logical buildup of information, starting from common classic campus network deployments, why change is required through architecture frameworks, to the concept of IBN. If you are already familiar with the specific topic of a chapter, it is possible to skip that chapter. Part I includes the following chapters:

- **Chapter 1, "Classic Campus Network Deployments":** This chapter provides you with an overview of campus network deployments found in many organizations today at a conceptual level. The chapter does not provide many details on how technologies are configured and used within a campus network but focuses on the conceptual designs and choices for a campus network with the advantages and disadvantages of each choice. The chapter covers concepts such as a hierarchical campus network, a collapsed-core model, different wireless deployment models, and alternatives for typical networking technologies found in the campus network such as the Spanning Tree Protocol (STP).
- **Chapter 2, "Why Change Current Networks? A Need for Change":** This chapter provides a summary of external trends, drivers (or forces), that require the current campus network design and operation to be changed to cope with these drivers. You will learn about external trends such as wireless/mobility, (Net)DevOps, complexity, cloud, and digitization.
- **Chapter 3, "Enterprise Architecture":** This chapter describes the concept of enterprise architecture, why that is beneficial to enterprises in general and specific to a

network design within the enterprise. You will learn that a network design (or architecture) is part of a larger technology architecture and an architecture for the enterprise or organization as a whole. This chapter uses the TOGAF® standard for enterprise architectures both as an introduction to enterprise architecture as well as an example for the relationship between network infrastructures and enterprise architecture.

- **Chapter 4, "Cisco Digital Network Architecture":** This chapter provides you with a detailed explanation of Cisco Digital Network Architecture (DNA) that Cisco introduced in May 2016. Cisco DNA is an architecture that is intended to be the foundation for both modern state-of-the-art network infrastructures as well as the concept of Intent-Based Networking. You will learn the different requirements Cisco DNA has, the building blocks of its architecture, and the different design principles.
- **Chapter 5, "Intent-Based Networking":** In this chapter you will learn the concept of Intent-Based Networking (IBN) via the explanation of what Intent means, how that can be applied to a network infrastructure, and how it relates to Cisco Digital Network Architecture. You will learn how IBN can help the network operations team cope with the changes described in Chapter 2 and be able to remain in control of the network. Chapter 5 also introduces two technical concepts, Software Defined Access and Classic VLANs, that can be used to deploy an Intent-Based Network. Intent-Based Network (or Intent-Enabled Network) is used throughout this book as a network that is configured based on the concept of Intent-Based Networking (IBN)—that is, IBN describes the concept, and Intent-Based Network is an implementation of that concept.
- **Chapter 6, "Tools for Intent":** This chapter provides a short overview of the tools that are available to enable Intent-Based Networking within the campus network. In this chapter you will learn about important concepts within IBN, such as automation and assurance, and which tools can fulfill those requirements for the concept of IBN.

Part II, "Transforming to an Intent-Based Network," describes a four-phased approach that supports you in a successful transformation to Intent-Based Networking, including tips, tricks, and problems you might face during the transformation. It is best to read this part completely and not skip a chapter as they are built on one another.

- **Chapter 7, "Phase One: Identifying Challenges":** This chapter describes the first phase of the transformation and is all about identifying the requirements (I used the word *challenges* on purpose, as that is more positive and challenges can be solved) within a campus network and getting the proper support and commitment. You will learn that IBN is a concept that not only involves (new) technologies and sets specific hardware requirements but also sets requirements and expectations on the organization. You will learn which challenges are to be identified on the required hardware and software as well as challenges related to the organization and its processes. The chapter ends with an action plan that contains details on how to approach the transformation to IBN.

- **Chapter 8, "Phase Two: Prepare for Intent":** This chapter describes phase two of the transformation. Phase two is used to prepare the campus network and the organization for the transformation. It starts with solving all the challenges identified in the previous phase. After these challenges are solved, the remaining steps of the phase focus on preparing the network (and network operations team) for a successful transformation to IBN. You will learn about the standardization of the campus network, the introduction of automation and assurance to the network operations team, and why these steps are important for IBN. These steps also include tips on why and how to execute them. The last section of this chapter contains information about risks that you might encounter during this phase, including some suggestions on how to cope with them.

- **Chapter 9, "Phase Three: Design, Deploy, and Extend":** This chapter provides all information required to actually transform the campus network to an Intent-Based Network. You will learn two technology concepts (Software Defined Access and classic VLAN) that can be used to deploy your campus network with their pros and cons. You will be executing sequential steps to gradually implement IBN on the campus network. As with the previous phases, a special section on risks for this phase allows you to identify and prepare for potential problems.

- **Chapter 10, "Phase Four: Enable Intent":** This chapter describes the last phase of the transformation; now that the campus network is transformed to IBN, it is time to fully take advantage of the possibilities created. The chapter involves a strategy on how you can introduce IBN to the rest of the organization, including a special methodology to allow the campus network to deliver services to the business on demand.

Part III, "Organizational Aspects," is quite a different part compared to the first two. Whereas the first two are primarily focused on background information and hands-on for the transformation, this part provides information on the impact that IBN has on the organization. This part's chapters can be read individually and are as follows:

- **Chapter 11, "Architecture Frameworks":** This chapter provides a quick recap of architecture frameworks and provides an insight to how IBN will impact and change the traditional enterprise architecture frameworks.

- **Chapter 12, "Enabling the Digital Business":** This chapter provides a more detailed description of the concept of digitalization and digital business. It describes how Intent-Based Networking fits within (and enables) the digital business and what impact this will have on an organizational level.

- **Chapter 13, "IT Operations":** This chapter describes the relationship between IBN and common IT operations. It provides an introduction to common IT operation models such as ITIL, DevOps, and Lean. It also describes what impact and change IBN will have on these IT operation models.

- **Chapter 14, "Tips to Make Your IBN Journey a Success":** The transformation to IBN involves quite some change, including technical, organizational, and individual. This chapter provides background information and tips on how change can be achieved at both an individual and an organizational level. It covers information on human change and associated fears, and it provides tips you can use to make the change happen. It also contains some final tips that can be used to make the transformation to IBN last.

Part IV, "Appendixes" This book also has appendixes that provide you with conceptual background information on the technologies named throughout this book as well as reference configurations for the underlay of an Intent-Based Network:

- **Appendix A, "Campus Network Technologies"**: This appendix provides a conceptual overview of the different technologies used in this book that are commonly found on a campus network. This appendix does not provide a detailed technical explanation of the technologies but rather a more conceptual summary of the technology and how it is applied. This appendix is a recommended read for less-technical readers to provide a more common understanding of the technologies;
- **Appendix B, "Sample Configurations"**: This appendix provides a set of sample configurations that can be used on the campus network to implement IBN; this appendix is rather technical as it contains specific configuration elements. It is provided as a head start to Intent-Based Networking for network architects and engineers.

Chapter 5

Intent-Based Networking

By and large, Cisco DNA describes the requirements and operations of a network infrastructure of an enterprise on an abstract level. Cisco DNA achieves this description by dividing the requirements of the enterprise network into several functions and design principles. Cisco DNA itself does not describe how to use or implement that network architecture. You can compare it with the design of a large office building. The drawings provide enough guidelines and a viewpoint of how the building will look. But it does not provide details on which materials the contractor needs to create the building or which functionality the building will be used for. Cisco DNA is nothing more than a description of the network in an abstract manner.

Intent-Based Networking (IBN) provides a powerful description and methodology on how you can use that network if it is built using Cisco DNA's specifications and requirements. IBN is essentially a viewpoint or perspective of an implemented network using Cisco DNA's requirements, design functions, and abstraction levels.

But what is Intent Based Networking? What perspective does it provide? This chapter describes IBN in more detail and covers the following topics:

- What is Intent
- Intent-Based Networking paradigm
- IBN designs
- Network as a platform
- Possible IBN implementations
- IBN examples

What Is Intent?

To understand what Intent-Based Networking is, it is important to know more about what *Intent* encompasses. *Purpose* is a synonym and probably makes the definition of intent easier to understand.

Every person, department, or organization has multiple intents or purposes. An organization can have the purpose to provide the best in class of software to schools, or to provide the best phones in the world. A business process can have the intent to fulfill its described task in the most efficient manner. A person can of course have multiple intents or purposes. In general, intent or purpose is a description of a goal to be achieved.

A good example of intent would be that my wife likes me to clear out the garbage cans in the kitchen and put their contents in the containers outside our home. My actions to fulfill her intent would then be: Take out the general-waste trash bag from the can in the kitchen and carry it to the appropriate container outside. Walk back to the kitchen and then take the bag of recyclable waste and put it in its correct container. Clean the kitchen cans if needed and put new trash bags in them.

This example describes intent quite well. My wife has an intent, and I have described steps to fulfill that intent. And once you take this point of view to many common tasks, intent can be seen everywhere. Table 5-1 shows some examples of intent.

Table 5-1 *Overview of Intents*

Intent	Execution Steps
I need the lawn cut.	Take the mower out of the garage, connect it to power, pull cord to start, push onto lawn and mow in lanes until lawn is finished, power off the mower, remove grass from the lawn, disconnect the mower from power cord, clean grass from the mower, and put it back in the garage.
I'm organizing a dinner party.	Invite friends, prepare dinner as much as possible ahead of time, clean up the house, dress up, welcome friends, finish and serve the dinner, clean up the table, and have a great evening.
I want to drive the car.	Check whether enough fuel is in the car; if not, drive to the nearest gas station and fill up the tank; start driving.
This sales order needs to be shipped.	Check the stock for this order, search each item in warehouse, pick the required items of the sales order, place them in a box, print the packing slip and place it in the box, fill the box with bubble wrap and close it, notify shipping organization of shipment, print the shipping label, stick it on the box, and place the box on the outgoing platform.
Next year I need to replace firewalls.	Prepare a budget proposal for the CFO explaining why replacement is required, present the proposal, wait for approval, request quotes, procure hardware, execute project to replace firewalls in production.

Intent	Execution Steps
This car needs to be assembled.	Procure all required parts, components, and implementation details; weld the chassis; place the chassis on the belt; let robots and workers assemble all parts; execute quality and assurance testing; prepare the car for shipment, and ship the car to the dealer.
I need to upgrade the code on the network switch.	Determine the new version of the software, upgrade the test environment with the new version, execute tests to check if the new version works with existing designs, validate results, request a change window for update, notify end users, execute update, validate if the upgrade was successful, update documentation, and close the change.

As you can see, intent is everywhere. An intent is essentially a brief description of the purpose and a concrete predetermined set of steps that need to be executed to (successfully) achieve the intent. This principle can also be applied to the operation of a network infrastructure. The intent and its steps describe very specifically what needs to be done on the network to accomplish a specific task. Table 5-2 provides a number of examples of how intent can be applied on a network infrastructure.

Table 5-2 *Overview of Network-Based Intents*

Intent	Execution Steps
I have a telepresence session at 10:00 a.m.	Create an HD video session to the remote peer, create the required end-to-end Quality of Service parameters for this specific session, reserve the bandwidth, set up audio, validate performance, keep the connection safe and secure during the session, once finished disconnect the HD video session, remove the end-to-end quality of service session, and remove the bandwidth reservation.
This application is migrating to the cloud.	Take the existing access policy for that application from the datacenter policy, transform the policy into an application policy for Internet access, deploy the policy on all perimeter firewalls, and change routing for that application to the cloud.
This new IoT application needs to be enabled.	Create a new logical compartment on the network, create an IP-space, set up Internet access policies, create access policies to recognize the IoT devices, and assign them to the logical compartment.
This application needs access to HR systems during salary runs.	Once the user starts the run, request access to system via the network, open the required ports and IP addresses for the device that user is connected with via an access policy, wait until salary run is finished, remove the temporary access policies, and clear the open connections.
Potential malware has been found on a device.	Reallocate the device to an investigate policy that includes in-depth monitoring of traffic and host isolation, execute a Change-of-Authorization to place the device in the new policy, notify security and administrator of a possible incident, and await investigation.

Table 5-2 provides only a small number of examples, but the possibilities are endless. The most important condition (and restriction) is that the proposed intent must be written in controllable, repeatable execution steps, so that the automation function within Cisco DNA can execute those steps automatically. In summary, Intent-Based Networking is a perspective or viewpoint on how a network infrastructure that meets Cisco DNA's functions, design principles, and requirements is operated. Using this perspective to operate the network will in turn enable the enterprise to embrace digitalization and the digital enterprise.

The following sections describe in more detail how this perspective leverages Cisco DNA's functions and design principles to achieve Intent-Based Networking.

Intent-Based Networking Perspective

IBN can be seen as a perspective on a Cisco DNA-based network infrastructure that describes how the network can be managed, operated, and enable a digital business. It translates an intent within the business into the configuration of the network required for that specific intent. This is achieved by defining the intent as a number of (repetitive) steps that can be deployed. The IBN perspective uses all aspects of Cisco DNA (design principles, concepts, and so on) to accomplish this method of approaching the network.

The IBN perspective is based on a systematic approach where the network infrastructure is seen as a holistic system.

Figure 5-1 shows this systematic approach.

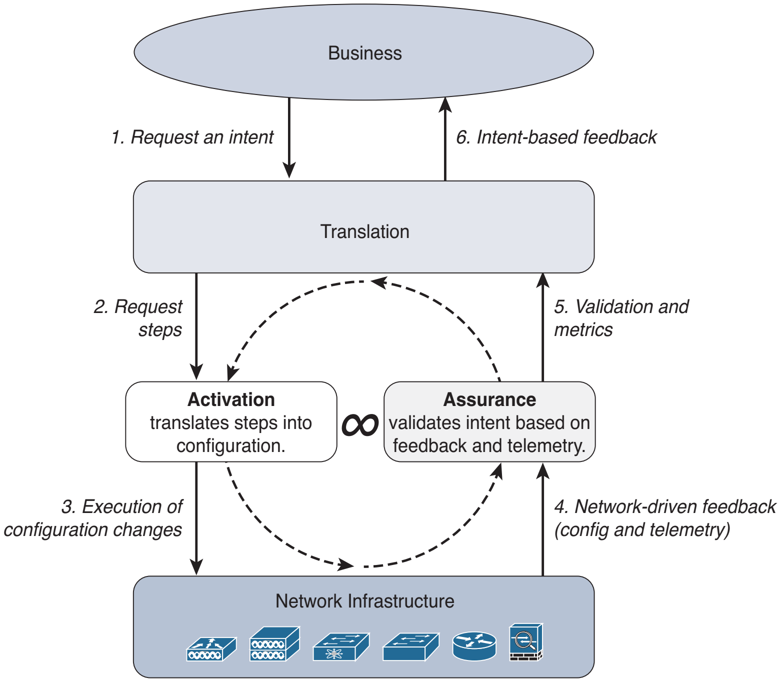

Business
1. Request an intent
6. Intent-based feedback
Translation
2. Request steps
5. Validation and metrics
Activation translates steps into configuration.
∞
Assurance validates intent based on feedback and telemetry.
3. Execution of configuration changes
4. Network-driven feedback (config and telemetry)
Network Infrastructure

Figure 5-1 *IBN Systematic Approach to the Network*

This approach resembles the functional approach of Cisco DNA a lot. It is, of course, a perspective on how a Cisco DNA-based network infrastructure is operated. This approach is based on six steps in a continuous loop.

1. **Request an intent:** A part of the business, whether a process, front-end application, or operator, specifies a specific Intent request to the network infrastructure. This is of course based on a number of available intents, where the variety of the Intent also depends on the organization and the level of availability.

   The translation process, receiving the request for Intent, translates the specific intent into a number of repetitive executable required steps. This is, for deploying a network Intent, perhaps one of the most important aspects of Intent-Based Networking. These steps need to be designed, tested, and defined within the translation process. Depending on the solution, these steps could be predefined templates or specific pieces of network configuration specific to the enterprise. The implementation of these repetitive executable steps needs to be as predictable as possible and is quite often defined by the network designer.

   For example, if the intent is a new IoT network, the Intent is translated into steps like create a new network, assign an IP-pool to that network, and place this network into a single logically separated compartment.

2. **Request steps:** Once the required steps for the specific intent are defined and created, the requested steps are sent to the Activation process. This process receives the requested steps and translates these steps into device-specific configuration changes that are required on the network infrastructure. The Activation process knows which configuration changes are required on which devices and uses automation to activate the changes on the applicable network devices. In turn the Activation process pushes the required changes to the network infrastructure.

   Based on the earlier example of intent, the Activation process translates the new network into a new VRF on a number of core switches in the network and allocates a new VLAN where the IoT devices are placed. The allocated IP pool is translated into a DHCP scope on the device that provides DHCP services to the network. A security policy can automatically be added to detect and authorize the new IoT devices.

3. **Execution of configuration changes:** This is the step where the Activation process actually connects to the network devices and deploys the changes to the network infrastructure. At this stage the requested Intent has been translated into a specific configuration on the network infrastructure. The requested intent is implemented. Although the Activation process performs pre- and postchecks to validate if the configuration of the network infrastructure devices was successful, the Activation process cannot determine whether the deployed configuration has the desired outcome.

4. **Network-driven feedback:** In this step, the network infrastructure devices provide feedback to the assurance process. The feedback is based on a number of data flows, including the generated network configuration, telemetry on the network, which client is connected, and how clients behave on the network. The network-driven feedback is used to validate if the executed configuration changes from step 3 have resulted in the desired outcome.

Using the same example, the IoT devices are now connected to the network and assigned to the specific VLAN and VRF. Telemetry data on whether the IoT device gets an IP address and which communication flows are seen by the network are sent to the Assurance process.

5. **Validation & metrics:** In this step the Assurance process has analyzed and validated the different data flows received from the network infrastructure devices. The received data is combined and analyzed, and potential issues or correct operation is sent back to the Translation process. This step provides the feedback on whether the requested intent is working as expected, and which IP addresses clients have received, including metrics on the usage, are provided to the Translation process.

   Within the same example, the status of the intent, including client-related information is sent to the translation process.

6. **Intent-based feedback:** The translation process receives metrics and the validation of the requested intent operation. The Translation process checks the status of the requested intent continuously and determines whether problems exist with the requested intent. In case of a problem, the operations team is informed of the failed state, and the business can request the status of the requested intent as well. Similarly, statistics on the usage of the requested intent are provided to the business layer using application-based metrics such as number of devices, accumulated data usage, and availability statistics, some of the most often used key performance indicators for the business.

   Within the same example, this step provides a positive status back to the business that the requested intent is operating as requested and provides an aggregated overview on availability and bandwidth usage to the business. The new IoT application is running successfully on the network.

These individual steps are executed for every requested intent. A large network can quickly have hundreds of requested intents to be added and run on the network. In addition to the requirement that these steps can run in parallel for different intents, the network must also be able to validate whether the requested intents are still working as expected. Therefore these steps, including the validation, are run in a continuous loop (shown by the dotted arrows in Figure 5-1) to validate whether the network is operating and working as designed. This allows the network to provide intelligent support to the operation teams on the performance and operation of the network.

The first three steps of IBN are common in today's campus networks. They are commonly deployed where automation is gaining momentum using a wide range of automation tools. The key difference between classic networks and an Intent-Based Network is that with IBN there is automated validation of the changes made to the configuration. The testing and validation of a configuration, also in operation, is unique to Intent-Based Networking and raises the quality and capability of the network.

Another key difference is that for IBN all steps and communication are based on APIs and models, conforming to Cisco DNA's design principles. This provides a new unique approach to the network, where applications can now automatically request intent to the network without the network operations team requiring to perform these changes. The feedback, based on the assurance, is also provided via APIs, so the same applications can validate that the requested intent is working as operated.

These two aspects of IBN—automatically validating changes of the network, as well as providing the network as a platform to software engineers—allow the enterprise to use the network in new, intuitive ways and enable the digital business.

Intent-Based Networking Designs

Intent-Based Networking is a perspective to a Cisco Digital Network Architecture. This means that specific designs and technologies are still required to allow a campus network to become Intent-enabled. The following sections describe two common Intent-Based designs (Software-Defined Access or using classic VLANs, known as non-Fabric) that can be used for a campus network to enable IBN.

Before the two Intent-Based designs are described, it is important to be aware that both designs have a certain set of requirements in common:

- **Policy-centric network:** First and foremost is the requirement that an Intent-Based design is based on a policy-centric environment and not a port-centric design. In other words, the network is not configured on a port-by-port basis, but uses a central policy server that pushes the required network port configuration as a policy to a network port.

  All policies for endpoints are pushed from this policy server into the network. This is key to enable intent onto a network, as the intent for an endpoint can change over time based on circumstances.

  For a specific policy to be set to an access port (or wireless network), it is necessary to know which endpoint is connecting to the network. Network Access Control (using IEEE 802.1X standard or MAC Authentication Bypass) is required to identify the endpoint requesting access to the network and to provide it with the proper authorization onto the network (by sending specific policies to the switch using RADIUS). A RADIUS deployment, such as Cisco Identity Services Engine, is thus required for IBN.

- **Microsegmentation:** To greatly enhance the security and tightly integrate it within Cisco DNA (and thus IBN), you should be able to segment a network into smaller bits than an IP subnet based on specific policies. This mechanism, already used in the datacenter, is called *microsegmentation*. Microsegmentation creates the possibility of having a single IP network for all IoT devices and having a policy that only IoT sensors are allowed to communicate with a local storage device where the data for those sensors are stored, while other IoT devices do not have access to that storage device. This microsegmentation must be based on a policy and be able to be programmatically applied to the network. Scalable Group Tags (SGT, formerly known as Security Group Tags) are used within a Software Defined Access (SDA) network (more on SDA in the next section) to provide this microsegmentation. Appendix A, “Campus Network Technologies,” describes in more detail how SGTs facilitate the required microsegmentation.

- **Feedback from network:** One of the true distinctions between a classic campus network as described in Chapter 1, "Classic Campus Network Deployments," and IBN is the feedback of the status of the network back to the controller. In other words, within an IBN the network devices provide feedback to the controller about the network's state. This feedback is used to validate whether the network is accomplishing the intent required. This feedback is of course received programmatically or via telemetry. Several methods and technologies are available to provide this feedback. The technical details for feedback are described in Appendix A, "Campus Network Technologies."

SDA

Software-Defined Access (SDA) is one of the latest technologies introduced in campus networks. It is the most complete technology (or actually a combination of technologies) that can enable Intent-Based Networking on your network.

The key concept of SDA is that there is a single, fixed underlay network and one or more overlay networks (running on top of the underlay). This concept in itself is not new; it is the founding principle for any network where encapsulating and decapsulating data allows the data to be abstracted from the different OSI layers. This principle is also used for VPNs over the Internet, CAPWAP tunnels for wireless communication, and within datacenters.

The principle of an underlay and overlay network can best be described using a common technology in enterprise networks—the Cisco Remote Access VPN solution based on Cisco AnyConnect. This technology allows end users to connect securely to the enterprise network via a less secure network (Internet).

This is realized by creating specific group policies (and pools of IP addresses) on the VPN headend device (a Cisco ASA firewall or Cisco Firepower Threat Defense firewall).

Users use the AnyConnect client to connect to the VPN headend over the Internet. Based on the authentication and authorization, users are allocated a specific internal IP address and the policies determining their access to the enterprise network. The user's endpoint uses the internal IP address to communicate with the enterprise network. This is accomplished by encapsulating the internal IP addresses into an outer packet destined to the VPN headend.

At the VPN headend, the packet is decapsulated and routed into the enterprise network. A similar path is realized for return traffic. The enterprise network only knows that the IP address of that user needs to be sent to the VPN headend. The VPN headend takes the internal traffic and encapsulates it in an outer packet with the destination of the public IP address of the end user.

In this example, the underlay network is the Internet, and the overlay network is the specific VPN group policy that the user is assigned to with the appropriate IP pool. SDA takes the same principle but then applies it internally to the campus network. SDA calls the underlay network a campus network, and it uses virtual networks on top of the underlay to logically separate endpoints. In other words, there are **no VLANs** within an SDA fabric. Figure 5-2 provides an overview of an SDA network.

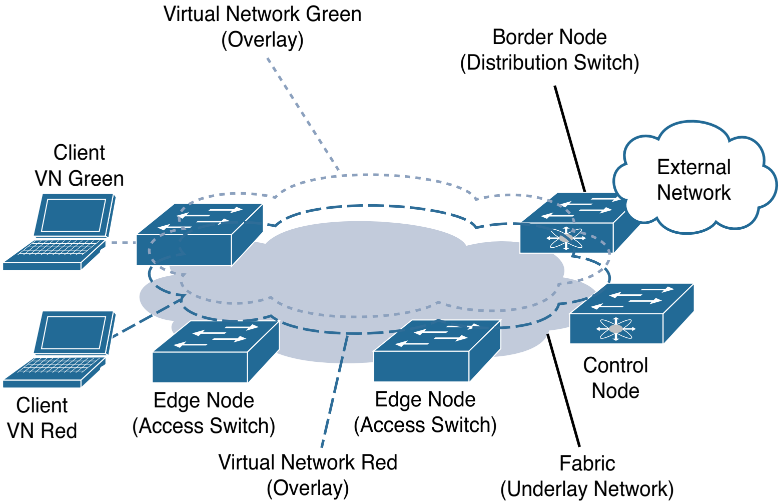


Figure 5-2 *Overview of an SDA Network*

SDA uses its own terminology to describe the roles and functions the switches (or in some cases routers) perform within the SDA fabric:

- **Virtual Network:** A virtual network is used to logically separate devices from each other. It can be compared with the way VLANs are used to logically separate devices on a switched network. A virtual network can be IPv4 or IPv6 based with one or more pools of IP addresses, but it can also be used to create a logical Layer 2 network. Each virtual network has its own routing and forwarding table within the fabric, comparable with VRF-Lite on switches. This principle provides the logical separation of the virtual networks.

- **Fabric:** A fabric is the foundation of the overlay network, which is used to implement the different virtual networks that run within a network. A fabric is a logically defined grouping of a set of switches within the campus network, for example, a single location. The fabric encompasses the protocols and technologies to transport data from the different virtual networks over an underlay network. Because the underlay network is IP based, it is relatively easy to stretch the underlay network across fiber connections on the campus (connecting multiple buildings into a single fabric) or even across a WAN (such as MPLS or an SD-WAN), factoring in specific requirements for SDA. These requirements are explained in Appendix A, “Campus Network Technologies.”

- **The underlay network:** The underlay network is an IPv4 network that connects all nodes within the fabric. An internal routing protocol (within an SDA-campus IS-IS is commonly used, although OSPF is also possible) exchanges route information within the fabric between the nodes. The underlay network is used to transport the data from the different virtual networks to the different nodes.

- **Edge node:** The edge node is used to allow endpoints to connect to the fabric. It essentially provides the same function as the access switch layer in a classic campus network topology. From an SDA perspective, the edge node is responsible for encapsulating and decapsulating the traffic for that endpoint in the appropriate virtual network. It also provides the primary role of forwarding traffic from the endpoint to the rest of the network.
- **Border node:** A fabric is always connected with external networks. The border node is used to connect the different virtual networks to external networks. It is essentially the default gateway of the virtual network to external networks. As each virtual network is logically separated, the border node maintains a connection to the external network for each individual virtual network. All traffic from the external network is encapsulated and decapsulated to a specific virtual network, so that the underlay network can be used to transport that data to the correct edge node.
- **Control node:** The control node is a function that cannot be related to a function within an existing classic campus network topology. The control node is responsible for maintaining a database of all endpoints connected to the fabric. The database contains information on which endpoint is connected to which edge node and within which virtual network. It is the glue that connects the different roles. Edge nodes and border nodes use the control node to look up the destination of a packet on the underlay network to forward the inner packet to the right edge node.

How SDA Works

Now that the roles, functions, and concept of an underlay/overlay network are known, how does SDA operate? What does an SDA network look like? The following paragraphs describe the way endpoints within a virtual network communicate with each other. Figure 5-3 provides an example topology of an SDA network.

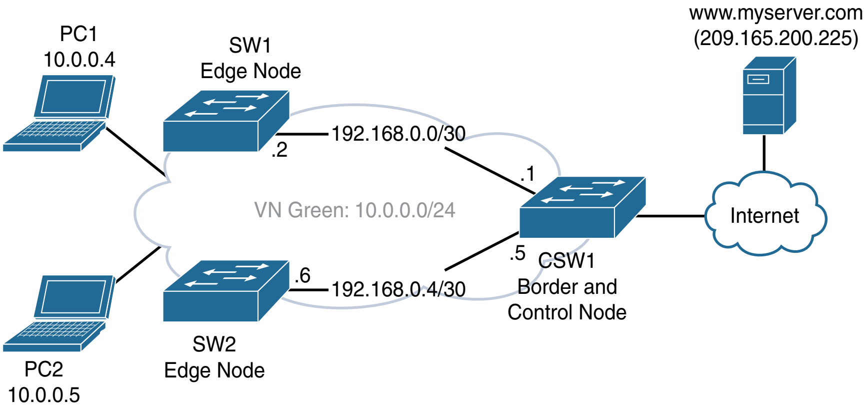


Figure 5-3 *Sample SDA Network*

In this SDA fabric there are three switches. The CSW1 switch provides the Border and Control functionality, while SW1 and SW2 are edge node devices in this fabric. Both SW1 and SW2 have an IP link to the CSW switch, using 192.168.0.0/30 and 192.168.0.4/30 subnets. There is a virtual network (VN) named green on top of the underlay network, which uses the IP network 10.0.0.0/24 for clients. PC1 has IP address 10.0.0.4, and PC2 has IP address 10.0.0.5. The default gateway for VN Green is 10.0.0.1.

CSW1 maintains a table of endpoints connected to the fabric and how to reach them. To explain the concept and operations, Table 5-3 describes the required contents for this example.

Table 5-3 *Overview of Fabric-Connected Devices in CSW1*

Endpoint Name	IP	Network	SGT	VN ID	Reachable Via
PC1	10.0.0.4		Employee	Green	192.168.0.2
PC2	10.0.0.5		Guest	Green	192.168.0.6
Internet		0.0.0.0	Any	Green	192.168.0.1, 192.168.0.5

In this network, if PC1 wants to communicate with www.myserver.com (IP 209.165.200.225), the following would happen:

1. After DNS resolution, the PC sends a TCP SYN packet to the default gateway (10.0.0.1) for destination 209.165.200.225.
2. SW1 as edge switch receives this packet and, as it is an anycast gateway (see Appendix A, "Campus Network Technologies" for more details), the packet is analyzed.
3. SW1 performs a lookup on the CSW1 (as control node) for the destination 209.165.200.225.
4. CSW1 returns a response for the lookup ip-address 192.168.0.1 (IP address of border node).
5. SW1 then encapsulates the *complete* TCP SYN packet in an SDA underlay network packet with source IP address 192.168.0.2 and destination address 192.168.0.1 and uses the global routing table to forward this new packet.
6. CSW1 receives the encapsulated underlay packet from SW1 and decapsulates it. It then, as border router, uses the routing table of VN Green to forward the traffic to the Internet.
7. The server www.myserver.com receives the TCP-SYN packet and generates a response with a SYN-ACK packet back to 10.0.0.4.
8. The incoming SYN-ACK packet is received by CSW1 in the VN Green network. The destination of the packet is 10.0.0.4.

9. CSW1 performs a lookup to the control node for VN Green and IP address 10.0.0.4 and gets 192.168.0.2 as the underlay destination.
10. CSW encapsulates the SYN-ACK packet for 10.0.0.4 into an underlay packet with destination 192.168.0.2.
11. The underlay packet is routed to SW1.
12. SW1 decapsulates the packet, recognizes it is for PC1 (IP 10.0.0.4) on VN Green, and forwards, based on a local table, the packet to the proper access port.
13. PC1 receives the SYN-ACK packet and responds with an ACK to further establish the TCP flow. The principle of lookup is repeated by SW1 for each packet received from or sent to PC1.

The preceding steps provide a conceptual overview of how communication is established and packets are encapsulated/decapsulated onto the underlay network. The same mechanism is used for communication within the VN Green itself. The control node is used as lookup to ask where a specific IP address is located, and then the original packet is encapsulated in an underlay packet destined for the specific node in the fabric. If microsegmentation policy would not allow communication from SGT Employee to SGT Guest, an access list on the edge node would prevent that communication.

An SDA-based topology is very powerful and capable in enabling IBN. The underlay network is set up only once, when the SDA network is created. In addition, there is increased flexibility in adding or removing edge nodes when required (as it is essentially a router in the underlay); all the endpoints are connected to one or more virtual networks. These virtual networks can easily be added or removed from the SDA network, without ever impacting the underlying network. This process of addition or removal can easily be programmed in small building blocks that automation can use. The Cisco DNA center solution is used to deploy and manage an SDA-based network.

Classic VLAN

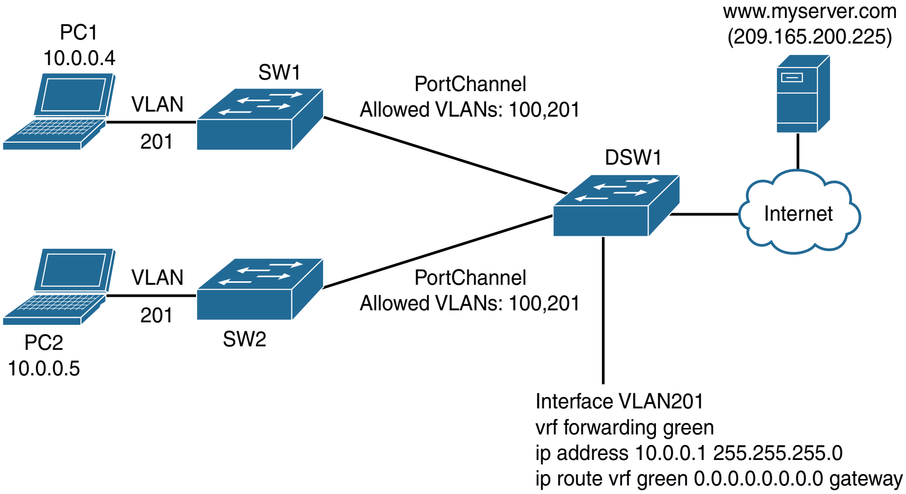
Although SDA was designed and built for Cisco DNA and is meant to solve some problems on classic campus networks, not all enterprises can readily implement SDA in their network. One of the main reasons is that there are some requirements on the network hardware and topology to enable SDA. This is not only Cisco DNA Center but also a full Identity Services Engine deployment as well as specific hardware such as the Cisco Catalyst 9000 series access switches. Although the Catalyst 3650/3850 are capable of running an SDA network, there are some limitations to that platform, such as an IP services license and a limited number of virtual networks.

However, if you look at an SDA through a conceptual looking glass, it is possible to replicate, with some limitations, the same concepts of SDA using classic VLANs and VRF-Lite. It allows an organization to transform to IBN while also preparing the infrastructure for SDA to take advantage of the concepts within SDA. Table 5-4 provides an overview of the concepts used in SDA compared to the technologies that can be used for IBN within a classic VLAN-based campus network.

Table 5-4 *Overview of Design Choices for SDA and Campus Alternative*

SDA Network	Classic Campus Network
Endpoints are, based on identity, assigned to a virtual network and an SGT.	Endpoints are assigned to a VLAN and SGT.
Each virtual network has its own routing table and IP space.	VRF-Lite can be used to logically separate IP networks and have each VRF instance its own routing table.
The provisioning of virtual networks is easy because the underlay is created only once and virtual networks can be added and removed without interrupting the underlay.	With automation tools, it is easy to programmatically add and remove VLANs on uplinks as well as SVIs on the distribution switch.
Routed links in the underlay are used to remove Spanning Tree and Layer 2 complexities.	In a collapsed-core campus network, there is no need for Spanning Tree, or a single Spanning Tree instance can be run to prevent loops.
An underlay network is used to stretch a fabric over multiple physical locations.	This is not possible without an encapsulation protocol in classic networks.
A control node is used for lookup of endpoints.	This is not required as existing protocols like ARP can be used.

With some limitations (specific conditions) it is possible to enable an IBN design using classic VLAN technologies. Limitations for such a design are a collapsed-core design, ability to assign SGT, and VLANs using a policy server as well as preferably no Spanning Tree or a single Spanning Tree instance. If you take these limitations, Figure 5-4 provides an intent-based design based on a classic collapsed-core campus network topology and VRF-lite.



```
Interface VLAN201
vrf forwarding green
ip address 10.0.0.1 255.255.255.0
ip route vrf green 0.0.0.0.0.0.0.0 gateway
```

Figure 5-4 *Intent Design Based on Classic Campus Collapsed-Core Topology*

In this design PC1 and PC2 still have the same IP address but are now assigned into VLAN 201 instead of virtual network green. VLAN 201 is configured on the DSW1 with IP network 10.0.0.0/24 and a default gateway of 10.0.0.1 for the endpoints. The SGTs have remained the same: Employee for PC1 and Guest for PC2.

Just as in the previous example, if PC1 would communicate with www.myserver.com on 209.165.200.225, it would send its TCP SYN packet to the default gateway on DSW1, which in turn would forward it to the Internet, while return traffic would be sent via Ethernet to PC1. ARP is used to map IP addresses into MAC addresses.

The principle of SGT ACLs to restrict traffic within a VRF is the same. In both SDA as well as classic, the SGT ACL is pushed from the policy server to the access switch where the endpoint is connected.

Although the end goal is logically separating traffic between endpoints, using SGT for microsegmentation, there are some limitations and restrictions on a classic VLAN over an SDA topology.

- **Spanning Tree:** It is not preferred to run Spanning Tree on the network, as each change in a VLAN can trigger a Spanning Tree recalculation, resulting in blocked traffic for a period of time. If it is required to run Spanning Tree, then run a single instance of Spanning Tree in MST mode, so that adding a VLAN does not trigger a new STP topology as with per-VLAN Spanning Tree.

- **Management VLAN and VRF:** It is required to have a dedicated VLAN and management VRF to be able to create or remove new VLANs. This VLAN may never be removed from trunks and networks, as this is essentially the underlay network. The automation tool that generates and provides the configuration communicates with all devices in this management VLAN.

- **Configuration via automation tool only:** The configuration of the campus network can *only* be executed via the automation tool. This is generally true for any environment that there should only be a single truth for the provisioning of a network. In an IBN based on classic VLANs, this is more important as the automation tool will generate the VLAN identifiers automatically based on the virtual networks to be deployed. Although it is common in enterprises to statically define and assign VLANs, in this design that concept needs to be removed for automation to work.

- **Standardized building blocks only:** It is important to only allow standardized building blocks, defined via the automation tool, on the campus network, where the policy is assigned policy-centric using IEEE 802.1x and RADIUS. The building block can then be standardized in such a way that small pieces of configuration code can be generated on-the-fly to create or remove the required compartments on the network. This is realized by creating small repetitive code blocks of command line configuration to be executed, for example, for the creation of a new compartment on the access switch:

```
vlan $vlanid
name $vrfname
interface $PortChannelUplink
switchport trunk allowed vlan add $vlanid
```

If the campus network configuration cannot be standardized, it will not be possible to enable an Intent-Based Network using VLANs.

- **Build your own automation:** With SDA, a lot of automation and configuration is executed by Cisco DNA Center in the background. With this design, an automation tool needs to be installed and configured by the network team to provide similar functionality. This can require some custom coding and testing before running the solution in production. This could be Cisco DNA Center with templates or another tool that provides automation functionality.

In summary, both mechanisms (SDA and classic VLAN) work quite similarly, and when you take certain precautions and keep the limitations in mind, it is feasible to start with IBN based on a classic collapsed-core topology. Part 2, "Transforming to an Intent-Based Network," provides more details on limitations, drawbacks, and when which technology fits best for transforming a campus network to IBN.

Summary

Cisco Digital Network Architecture describes the requirements and operations of a network infrastructure of an enterprise at a functional or abstract level. Cisco DNA achieves this abstract description by dividing the requirements of the enterprise network into several functions and design principles. It does not describe how to use or implement that network architecture.

Intent-Based Networking (IBN) describes, using a powerful methodology, how a campus network can be built and operated using Cisco DNA as network architecture. IBN is based on the premise that every endpoint that connects to the network consumes a predefined set of services (that include access, connectivity, security policies, and other network functions). In essence, every endpoint has a specific intent (or purpose) when connecting to the network, and each intent is defined as a set of services to be delivered to that endpoint.

This set of intents (that are deployed on the network) are defined dynamically based on which endpoints are connected to the network. As soon as an intent is not required anymore, its configuration is removed automatically from the network infrastructure.

Although IBN itself is not based on Cisco Digital Network Architecture, its description and methodology are so similar to Cisco DNA that you can state it is a perspective of Cisco DNA. IBN describes how a network based on Cisco DNA can be configured and operated by the network operations team. Figure 5-5 describes the systematic approach IBN describes in providing intents to the network (by defining Intents as repetitive pieces of configuration).

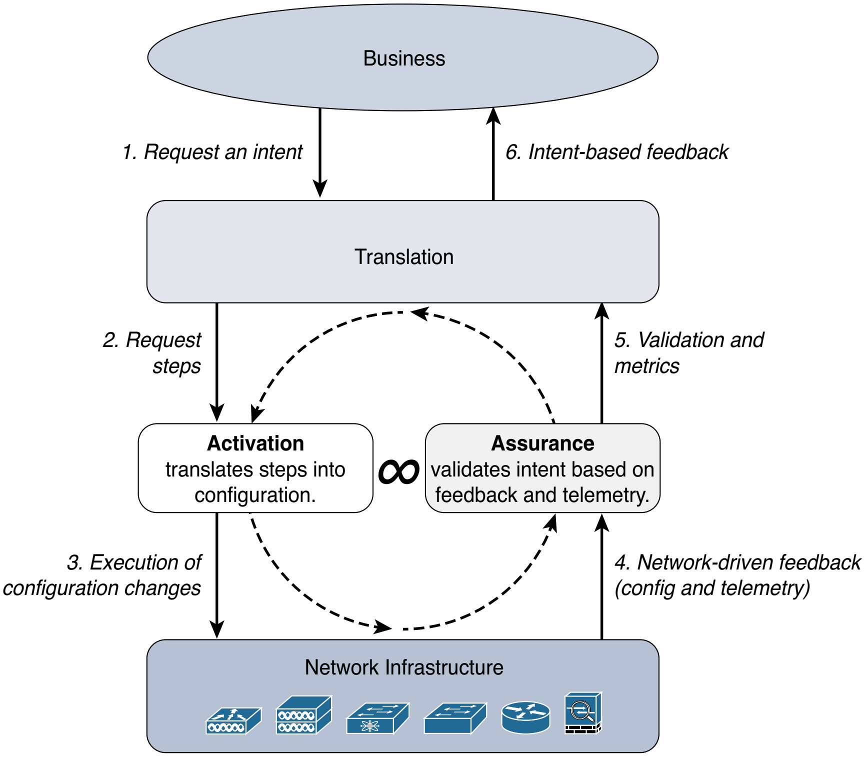

Business
1. Request an intent
6. Intent-based feedback
Translation
2. Request steps
5. Validation and metrics
Activation
translates steps into configuration.
∞
Assurance
validates intent based on feedback and telemetry.
3. Execution of configuration changes
4. Network-driven feedback (config and telemetry)
Network Infrastructure

Figure 5-5 *IBN Systematic Approach to the Network*

Figure 5-5 is similar to Cisco DNA, and IBN is based on six steps in a continuous loop:

1. Request intent; business or network operations request a specific intent.
2. Request steps; the intent is translated into a set of configuration changes to be executed.
3. Execution of configuration changes; network configuration changes are executed via automation.
4. Network-driven feedback; the network infrastructure provides feedback on its operation.
5. Validation & metrics; the analytics component validates the received network-driven feedback with the requested intents to validate that the requested intents are operating as requested and designed.
6. Intent-Based feedback; business-outcome based values are used to report on the status of the requested intent and its operation.

Two Designs

Two network designs are available to implement IBN:

- Cisco Software Defined Access (SDA) is based on Cisco DNA and is the most complete technology that can enable IBN on the campus network, but Cisco SDA does have specific requirements on the network infrastructure devices (and Cisco DNA Center).
- Classic VLANs with VRF-Lite can be used, with limitations, as an alternative to SDA for those organizations that are not (yet) able to meet the requirements of SDA.

IBN itself, and therefore both designs, relies on three key requirements on the campus network to be successful:

- **Policy-centric network:** The campus network is not configured port-by-port but uses a policy-centric identity server so that based on the identity of the endpoint the specific network policies (and thus the intents) can be pushed to the appropriate network infrastructure device.
- **Microsegmentation:** Microsegmentation is used within IBN to allow for more granular security policies than those based solely on IP addresses.
- **Feedback from network:** IBN relies heavily on the feedback that network infrastructure devices provide back to the analytics component; it is used to validate whether the requested intents are operating as designed and requested.

In conclusion, IBN is a perspective on Cisco Digital Network Architecture, and it describes a powerful methodology of how a Cisco DNA-based network infrastructure can be operated and managed. IBN can be used to provide the network operations team with the tools and methods to cope with the exponential growth of devices connecting to the campus network.

Index