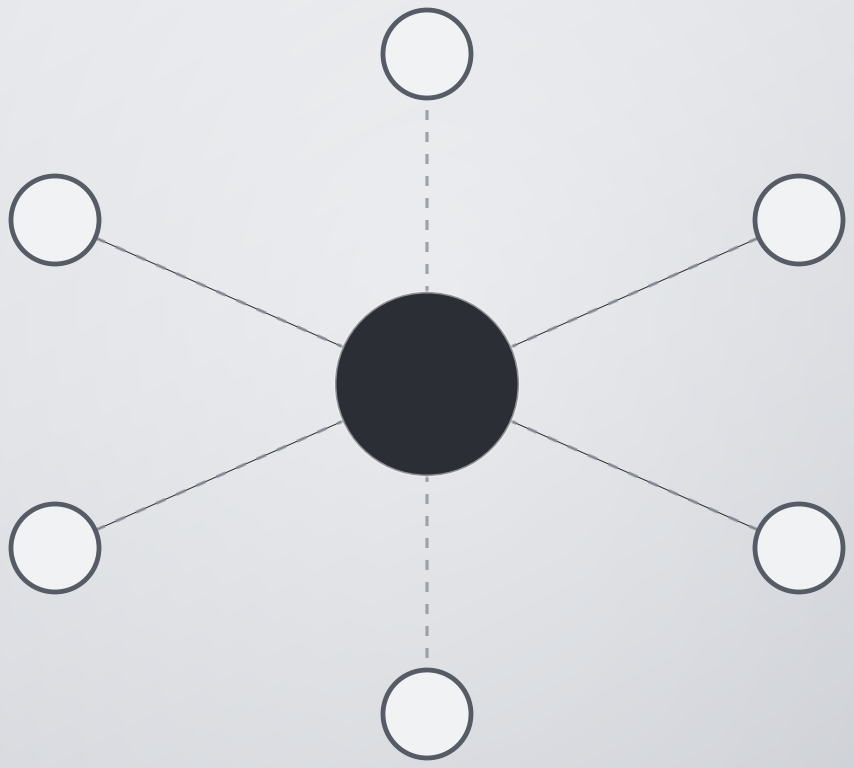

SDA

Fundamentals

JOZEF BAROŠ

A PRACTITIONER'S FIELD GUIDE

SDA Fundamentals

A Practical Guide to Designing, Building, and Operating Cisco SD-Access

Underlay & Overlay · LISP · VXLAN · TrustSec · Catalyst Center + ISE
Greenfield & brownfield, end to end · a fictional project from design to day-two

JOZEF BAROŠ
ProDigitalJ

Book 2 in the Fundamentals series · companion to SD-WAN Fundamentals

SDA Fundamentals

A Practical Guide to Designing, Building, and Operating Cisco SD-Access. First edition.

Trademarks and independence

Cisco, Cisco Systems, Catalyst, Catalyst Center, DNA Center, IOS XE, Identity Services Engine (ISE), TrustSec, and Meraki, together with their respective product names, are trademarks or registered trademarks of Cisco Systems, Inc. and/or its affiliates. All other trademarks are the property of their respective owners and are used for identification and educational purposes only.

This book is an independent publication and is not affiliated with, authorised by, sponsored by, or otherwise endorsed by Cisco Systems, Inc. It is an educational work written to help engineers understand and implement campus fabric concepts on the Cisco SD-Access platform.

No Cisco-copyrighted text, figures, diagrams, screenshots, or other proprietary material is reproduced in this book. All diagrams and illustrations are original works created by the author to explain openly documented concepts; the LISP and VXLAN protocols at the heart of SD-Access are open IETF standards (RFC 6830 and its successors, and RFC 7348, respectively). No third-party logos appear anywhere in this book.

Disclaimer

The information in this book is provided “as is,” without warranty of any kind, express or implied. While every effort has been made to ensure accuracy, the author assumes no responsibility for errors or omissions, or for damages resulting from the use of the information herein. Product capabilities, names, and interfaces change over time; always confirm details against current vendor documentation for the software release in front of you.

The company “Northforge Industrial,” together with its sites, staff, device names, addresses, and configurations, is entirely fictional and used purely for illustration; any resemblance to a real organisation is coincidental. All configuration snippets, IP addresses (drawn from documentation ranges reserved in RFC 5737 such as 192.0.2.0/24, 198.51.100.0/24, and 203.0.113.0/24, along with RFC 1918 private ranges), device identifiers, and command output are illustrative and simplified for teaching; they are not intended to be copied verbatim into a production network.

About This Book

This is a book for the engineer who has never implemented SD-Access and wants to be able to. It assumes you are comfortable with traditional networking — VLANs, routing, switching, a little security — but treats every SD-Access concept as new, defining each term the first time it appears and building understanding from first principles. If you can configure a traditional campus, you can follow this book; by the end, you will be able to design, build, secure, and operate a campus fabric.

How the book is organised

The book follows the real arc of a project. Chapters 1 and 2 build understanding — what SD-Access is, and how the fabric actually works. Chapter 3 turns requirements into a design. Chapters 4 and 5 stand up the foundation and bring sites online, both greenfield and brownfield. Chapters 6, 7, and 8 deliver the outcomes that justify the project — segmentation, wireless, and security. Chapter 9 extends the fabric across sites and connects it to the WAN, and Chapter 10 operates, automates, and troubleshoots it. Read in order, each chapter assumes the ones before it.

One project, all the way through

Rather than abstract examples, the whole book follows a single fictional company — **Northforge Industrial**, a mid-size manufacturer with offices, a plant floor full of OT and IoT, a warehouse, and small sales offices. Every concept is applied to Northforge's real problems: separating IT from OT without re-addressing, containing IoT, isolating guests and contractors, and migrating a running factory onto the fabric without stopping production. Following one company from design to day-two operations is what turns a pile of features into a method you can repeat.

A note on the value boxes

Throughout the book you will find four kinds of highlighted note: **Did you know?** (a deeper mechanism or a useful aside), **Common Problem** and **Watch out** (mistakes people actually make, and how to avoid them), **From the Field** (how a concept plays out in real production work), and **Design Tip** (a recommendation worth following). Each chapter also ends with a **Chapter Recap**

that distils its essentials. Skim the recaps to review; read the boxes for the hard-won lessons that separate a fabric that works from one that merely builds.

> ◆ **A companion volume**
> If you have read this book's sibling, SD-WAN Fundamentals, you already know the wide-area half of the story. This volume builds the campus half, and Chapter 9 is where the two fabrics shake hands — giving one identity-based policy from the plant-floor cabinet to a server in another city. Neither book requires the other, but together they describe one coherent, end-to-end network.

Contents

Chapter 1 — Understanding SD-Access

Welcome to your first SD-Access project. If the companion volume, *SD-WAN Fundamentals*, rebuilt the wide-area network, this book rebuilds the other half: the campus — the wired and wireless access network where users, phones, cameras, machines, and sensors actually plug in. Over the course of the book we will take a single, realistic manufacturer from an aging three-tier campus all the way to a fully operational Cisco SD-Access fabric — designing it, building the underlay, standing up the controllers, migrating live sites onto it, layering on the segmentation that makes it worth the effort, folding in wireless, securing it, extending it across sites, and finally operating, automating, and troubleshooting it day to day.

Before anyone touches a controller, though, you need a solid grasp of what SD-Access is and why it works the way it does. This first chapter builds that foundation: the problem SD-Access solves, the handful of ideas at its core, a tour of the components you will spend the rest of the book configuring, and the company whose network you have been hired to transform. Skip it and every later design decision will feel arbitrary. Read it and those decisions will feel obvious.

Meet the project: Northforge Industrial

Throughout the book we will follow one company. **Northforge Industrial** is a mid-size manufacturer of precision components: roughly 1,800 employees across a headquarters campus with an attached production plant, a second manufacturing site in another city, a distribution warehouse, and a handful of small sales offices. The main campus mixes ordinary office life — laptops, IP phones, wireless, printers — with a busy plant floor full of programmable logic controllers (PLCs), operator panels, machine-vision cameras, robotic cells, badge readers, and industrial sensors. Today Northforge runs a traditional three-tier campus LAN, and it hurts: onboarding a new machine line takes days of manual work, the security team loses sleep over how easily a compromised device could reach anything else, and nobody can answer a simple question like “why did the scanners in the warehouse drop off the network at 10:14 this morning?” without logging into switches one at a time.

By the end of the book you will have redesigned Northforge's campus as an SD-Access fabric — building one brand-new plant building as fabric from day one (greenfield) and migrating the existing campus onto the fabric with minimal disruption (brownfield). You will finish with the mental models, the configuration patterns, and the hard-won “watch out for this” knowledge to run a project like it yourself.

1.1 The Problem You Are Being Asked to Solve

How the traditional campus was built

To understand why Northforge hired you, you have to understand the network you are replacing. For most of the last two decades the standard enterprise campus followed one recipe: a **three-tier hierarchy** of access, distribution, and core. Endpoints connect to access switches; access switches uplink to a pair of distribution switches; distribution switches connect to the core. Between access and distribution the network is **Layer 2** — a patchwork of VLANs stretched across switches — and Spanning Tree Protocol (STP) runs everywhere to keep those Layer 2 loops from melting the network down. The distribution layer is where Layer 2 turns into Layer 3: each VLAN gets a switched virtual interface (SVI) as its default gateway, a first-hop redundancy protocol such as HSRP keeps that gateway alive, and access control lists (ACLs) on those interfaces decide who may talk to whom.

In this world, three things are welded together that later you will want to pull apart: **where a device connects** (which switch port), **what IP address it gets** (which VLAN/subnet), and **what it is allowed to do** (which ACL applies to that subnet). A device's identity, in practice, is its IP address, and its policy is a set of IP-based rules maintained by hand on every distribution switch.

The traditional model in miniature — policy welded to VLAN and subnet (illustrative, fictional addresses)

```
! Access switch: the port decides the VLAN, the VLAN decides everything
interface GigabitEthernet1/0/12
 switchport access vlan 40          ! VLAN 40 = "Plant-Floor"
 spanning-tree portfast

! Distribution switch: the SVI is the gateway AND the policy point
interface Vlan40
 ip address 10.40.0.2 255.255.255.0
 standby 40 ip 10.40.0.1            ! HSRP virtual gateway
```

```
 ip access-group PLANT-IN in        ! policy is an IP ACL, maintained by hand

ip access-list extended PLANT-IN
 permit ip 10.40.0.0 0.0.0.255 10.55.0.0 0.0.0.255   ! plant -> historian
 deny   ip 10.40.0.0 0.0.0.255 10.10.0.0 0.0.0.255   ! plant -> corporate (block)
 permit ip any any
! ...now maintain the equivalent on every distribution switch, by hand, forever.
```

This design earned its place. It is well understood, it is deterministic, and for a network of fixed desks and a handful of VLANs it works. The trouble is that the assumptions underneath it — devices that never move, a short list of endpoint types, and security that a subnet boundary can express — quietly stopped being true, and the network did not change with them.

The pressures that broke the model

No single event killed the traditional campus. A handful of pressures piled up until the access network became the bottleneck and the risk of the whole business. You will see every one of them at Northforge.

1. Mobility and wireless untied identity from location. When a laptop or an AGV (automated guided vehicle) roams across the plant, a design that pins policy to a switch port and a subnet cannot follow it. Either you stretch one giant VLAN everywhere — and inherit an enormous Layer 2 failure domain — or the device changes subnet as it moves and its policy silently changes with it. Neither is acceptable on a modern campus where most endpoints are mobile.

2. The endpoint population exploded. A campus used to mean PCs and printers. Northforge's campus now carries IP phones, wireless clients, security cameras, badge readers, building-management sensors, and — on the plant floor — PLCs, HMIs, vision systems, and robots. Each class needs a different level of access, but the flat, VLAN-per-location model has no clean way to express “a camera may reach the NVR and nothing else” without yet another VLAN and yet another ACL.

3. Segmentation by subnet and ACL does not scale. Every new isolation requirement becomes another VLAN, another subnet, and more lines in more ACLs. Those ACLs are tied to IP addresses, so they break the moment a device is renumbered or moved, they grow until they exhaust switch

hardware tables, and no one dares delete a line for fear of what it silently permits. Segmentation becomes something you talk about far more than you actually enforce.

4. Flat networks give threats the run of the place. The most expensive consequence is security. In a flat or lightly segmented campus, a single compromised endpoint — a phished laptop, an unpatched camera, an infected contractor machine — can reach laterally to almost anything else. When IT and OT share the same flat fabric, as they partly do at Northforge, a commodity ransomware infection on the office side can find its way to the plant floor and stop production. Regulators and cyber-insurers increasingly expect real internal segmentation, not a subnet diagram that claims it.

5. Operating it is manual, slow, and inconsistent. Every switch is configured by hand, one CLI session at a time. There is no single source of truth for "what should the plant-floor policy be everywhere," no automatic check that a change actually took effect, and no unified view of whether policy is really in force. Onboarding a new production line means touching many switches flawlessly; a small mistake becomes an outage. Scripting the CLI with Python or Ansible speeds up the typing, but it does not give you a model of intended network state — which is the real problem.

▲ **Common Problem — "We already have VLANs and ACLs, so we are already segmented"**

This is the single most common objection you will hear from an incumbent team, and it confuses two different things. VLANs and ACLs segment by **location and IP address**: a rule says "10.40.0.0/24 may not reach 10.10.0.0/24." That breaks the instant a device moves or is renumbered, it grows without bound, and it says nothing about *what a device is*. Real segmentation follows **identity** — "a plant-floor PLC may never reach a corporate workstation, wherever either one happens to be plugged in." SD-Access is built to express exactly that, and to enforce it consistently. When you hear this objection, that distinction is your answer.

◆ **From the Field — The day a flat network becomes an outage**

Ask any operations team that has lived through it. An infection lands on one ordinary machine in the office. Because the campus is flat where it matters, the malware scans and spreads sideways — and because a few plant VLANs were bridged into corporate years ago "just to collect production data," it reaches an HMI and a line stops. The root cause on the incident report is never the malware; it is the absence of enforced internal segmentation. This is the

risk Northforge is paying you to remove, and it is why the segmentation chapters later in this book are the heart of the project, not an add-on.

◆ **Did you know? — Spanning Tree was a workaround, not a goal**

Layer 2 needs a loop-free topology, so STP deliberately blocks redundant links to break loops — which means you pay for links you are not allowed to use, and a single misbehaving device or unidirectional fiber can still trigger a campus-wide storm. Decades of STP tuning exist to make a fundamentally fragile Layer 2 core survivable. One quiet benefit of a fabric is that the transport underneath it is fully routed: every uplink forwards, and STP stops being your core's single point of fragility.

Chapter Recap

The traditional three-tier campus welds three things together that you will soon want separate: where a device connects, what address it gets, and what it may do. Five pressures broke that model — mobility untied identity from location, the endpoint population exploded, subnet-and-ACL segmentation stopped scaling, flat networks let threats move laterally (dangerously so where IT meets OT), and box-by-box operation is slow and inconsistent. Northforge exhibits all five. SD-Access exists to fix them — and the next section reduces the whole solution to three ideas.

1.2 The Ideas That Make SD-Access Work

SD-Access can look like a pile of new acronyms — LISP, VXLAN, SGT, VN, RLOC — until you see that it rests on just three ideas. Get these three and everything else in the book is detail hung on the frame: **separate identity from location**, **build an overlay so the wiring stops mattering**, and **drive it all from centralized, group-based policy**. Each idea has a chapter of machinery behind it later; here we install the frame.

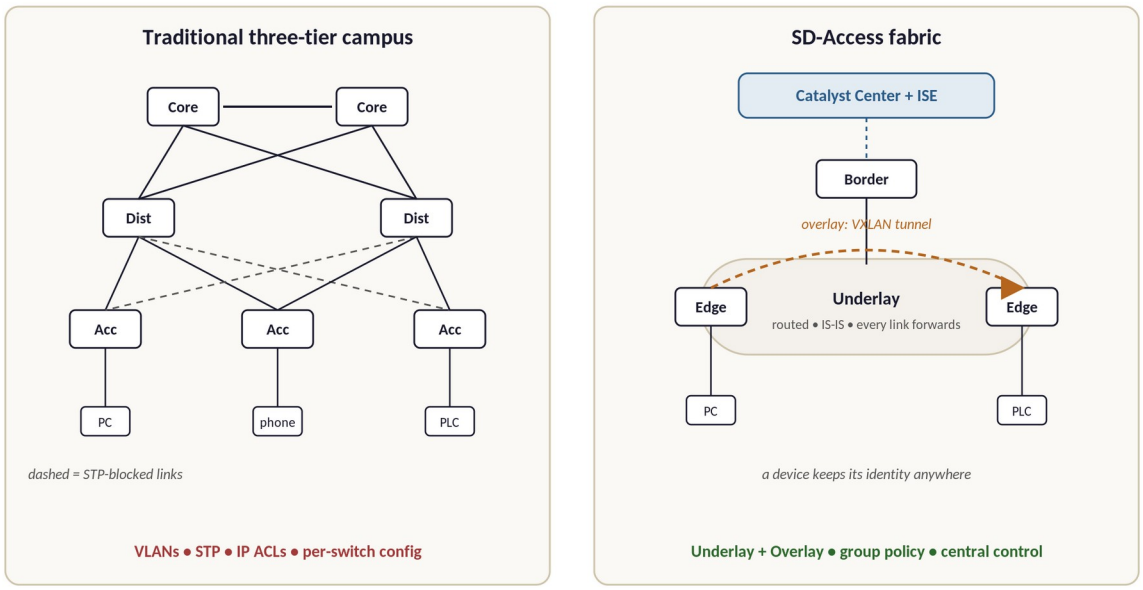

Figure 1.1 — The shift this book delivers: from a transport-bound, VLAN-and-STP three-tier campus (left) to a transport-independent SD-Access fabric with central, identity-based policy (right).

Idea 1: Separate identity from location

The founding move of SD-Access is to stop using a device's location and IP address as its identity. In the fabric, what a device is allowed to do is decided by **who or what it is** — a corporate laptop, a plant-floor PLC, a guest phone, a security camera — not by which port it uses or which subnet it landed in. When an endpoint connects, it is authenticated and placed into a **group**; policy is written between groups, and it follows the device wherever it goes. This is the campus expression of the same principle SDN brought to the WAN: pull the policy logic away from the individual box and express it as intent about the whole system.

The mechanism, which you will meet properly in Chapter 2, is a tag called a **Scalable Group Tag (SGT)**. Think of it as a small number that means "this traffic belongs to the Plant-Floor group" or "…the Corporate group," carried with the packet across the fabric. Policy becomes a compact matrix of group-to-group rules — "Plant-Floor may talk to Historian; Plant-Floor may not talk to Corporate" — that no longer cares about IP addresses at all. Move the device, renumber the subnet, put it on wireless: the group, and therefore the policy, comes along.

> ◆ **From the Field — A badge, not a doorway**

In an old office you controlled access by controlling doorways — who could reach which corridor. That is subnet-and-ACL thinking: security depends on where you are standing. SD-Access hands every device a **badge** instead. The badge (its group) determines what it may open, and it works the same whether the person is at their desk, in a meeting room, or visiting another floor. You stop reasoning about doorways and start reasoning about badges — which is a far smaller, far more stable set of rules.

Idea 2: Build an overlay so the wiring stops mattering

The second idea is the **overlay**, and it is the one that makes the first idea practical. An SD-Access fabric is really two networks running on the same wires. The **underlay** is a plain, fully routed IP network whose only job is to move packets between the fabric switches reliably and quickly. It carries no user subnets and almost never changes. On top of it, the fabric builds an **overlay**: tunnels between the access switches that carry all the actual user traffic. Endpoints live in the overlay; the underlay just transports it.

Two protocols make the overlay work, and both are open standards rather than anything proprietary. **VXLAN** is the data-plane encapsulation — it wraps each user packet so it can be tunneled across the routed underlay while carrying the endpoint's virtual network and its group tag along with it. **LISP** is the control plane — a lightweight directory that keeps track of *where each endpoint currently is*, mapping the endpoint's address to the fabric switch it sits behind. Because the overlay treats the underlay as just “something that forwards packets,” you gain the same freedom SD-WAN gave the WAN: the transport underneath becomes interchangeable, every uplink is usable, and an endpoint can keep its IP address as it roams because LISP simply updates where that address lives.

◆ Did you know? — You have already used an overlay

If you have ever run a VPN over hotel Wi-Fi — or read the SD-WAN companion to this book — you have used an overlay: a tunnel that behaves the same regardless of the network carrying it. SD-Access industrializes that idea inside the campus. VXLAN and LISP are IETF standards (RFC 7348 and RFC 6830 and their successors), not a closed invention; Cisco's contribution is the controller and the integration that make them easy to run at campus scale. That is worth remembering the first time the acronyms feel intimidating.

> ▲ **Watch out — The overlay does not create bandwidth or erase design**
> A fabric makes the transport flexible; it does not conjure capacity or forgive a weak underlay. VXLAN adds header overhead, so the underlay must carry a larger MTU (you will plan for this in Chapter 3), and a congested or unstable underlay produces a congested, unstable overlay. "Overlay" means "transport-independent," not "transport-free." Honest capacity and resilience planning for the physical network is still your job — the fabric just stops that physical design from dictating your addressing and policy.

Idea 3: Centralized, group-based policy

The third idea is where SD-Access earns its keep operationally. In the old world, a business rule such as "the plant floor and the corporate network must never talk directly, and guests never touch either" had to be hand-translated into VLANs and ACLs on every distribution switch. In SD-Access you express that intent **once**, centrally, and the system compiles it into the fabric and pushes it to exactly the devices it applies to. Two products do this together: **Catalyst Center** is the controller where you design the network and author intent, and **Identity Services Engine (ISE)** is the identity brain that authenticates each endpoint, decides which group it belongs to, and holds the group-to-group policy. Change the intent in one place and the whole fabric converges on the new rule — no per-box editing, and a single source of truth for what the policy actually is.

These three ideas are not independent — they reinforce each other. Group-based policy (Idea 3) only works because identity is separated from location (Idea 1), and identity can follow a device around only because the overlay (Idea 2) decouples the endpoint from the physical topology. Everything else in this book — LISP, VXLAN, SGTs, virtual networks, LAN Automation, fabric wireless — is machinery in service of these three ideas. When a later mechanism feels fiddly, trace it back to which idea it serves and it will make sense.

> **Chapter Recap**
> SD-Access stands on three ideas. **Separate identity from location:** decide policy by what a device is (its group / SGT), not where it plugs in or what IP it holds. **Build an overlay:** run a simple routed underlay whose only job is transport, and carry all user traffic in a VXLAN overlay whose location directory is LISP. **Centralize group-based policy:** express intent once in Catalyst Center and ISE and let the system compile it to the fabric. The three interlock —

> and the rest of the book is machinery serving them.

1.3 Meet the Platform: SD-Access, Catalyst Center, and ISE

One solution, two names you will hear

SD-Access (Software-Defined Access) is the name for the campus fabric solution as a whole: the fabric protocols, the switches and access points that form the fabric, and the two software systems that design and police it. As with the WAN products, you will live with a naming quirk: the controller was launched as **DNA Center** (spoken "DNAC") and has since been renamed **Catalyst Center**. Both names refer to the same product, and you will see "DNA Center" in older menus, documentation, forum posts, and API paths for years to come. This book gives the current name — Catalyst Center — first, notes the classic name where it helps, and then uses the current name throughout, exactly as you will on a real project.

The two brains and the fabric they run

SD-Access has two control points and one fabric. It helps to meet them before the mechanics, because every later chapter is really "do something in Catalyst Center, backed by ISE, that changes how the fabric behaves."

Catalyst Center (formerly DNA Center) is the controller and single pane of glass. You use it to model the network hierarchy (sites, buildings, floors), automate the underlay, provision devices into fabric roles, express segmentation and policy intent, and — through its **Assurance** feature — monitor health and troubleshoot with rich telemetry. It is a GUI and a full REST API; most of your day is spent here, not on the switch CLI.

Identity Services Engine (ISE) is the identity and policy server. It authenticates every endpoint that joins the fabric (via 802.1X for devices that can log in, and MAC Authentication Bypass, MAB, for devices that cannot), **profiles** unknown devices to work out what they are, assigns each one to a Scalable Group, and stores the group-to-group (TrustSec) policy that the fabric enforces. Catalyst Center and ISE are joined at the hip: they share information over a secure channel called **pxGrid** so that policy authored in one place is understood in the other.

Underneath the two brains are the **fabric devices** — the switches and access points that actually form the fabric. Each plays a **role**, and you will meet all of them in detail in Chapter 2. For now, a first acquaintance:

Component	Role in the fabric	What it does for you
Catalyst Center	Controller / management	Design, automate the underlay, provision fabric roles, author policy intent, and run Assurance — from one GUI and API.
ISE	Identity & policy	Authenticate and profile every endpoint, assign it to a group (SGT), and hold the group-based policy the fabric enforces.
Control plane node	LISP directory	Tracks where every endpoint currently is — the map that lets the overlay find any device on any switch.
Border node	Fabric edge to the outside	The doorway between the fabric and everything beyond it — the WAN, the data center, the Internet, shared services.
Edge node	Access switch	Where endpoints connect. Authenticates them, registers them with the control plane, and encapsulates their traffic into the overlay.
Fabric WLC + APs	Wireless into the fabric	Bring wireless clients into the same fabric, groups, and policy as wired — not a parallel network bolted on the side.
Extended node	Reach beyond the wiring closet	Extend fabric connectivity to small or ruggedized switches — ideal for the plant floor and IoT closets.

How the pieces come together: the join story

Here is the whole system in one story — each step becomes a full chapter later. When a device at Northforge powers on and connects to an **edge node** (an access switch), the edge asks **ISE** to authenticate it. ISE checks the device (by certificate or credentials for 802.1X, or by MAC address and behavior for MAB), decides which **group** it belongs to, and hands the edge back three things: permission to connect, the **virtual network** the device lives in, and its **group tag (SGT)**. The edge then registers the endpoint's address with the **control plane node** so the fabric knows where the device now lives. From that moment, when the device sends traffic to another endpoint, the edge wraps it in **VXLAN** — stamped with the destination's location, the virtual network, and the group tag — and forwards it across the routed underlay to the far edge, which unwraps it and

enforces the group-based policy. Traffic destined outside the campus exits through a **border node**. All the while, **Catalyst Center Assurance** is watching, correlating, and ready to tell you where something broke.

That single sequence is also your troubleshooting map. When an endpoint "cannot get on the network," you walk the same steps in order — did it authenticate at ISE, did it get a group, did the edge register it with the control plane, is the overlay path healthy, is policy permitting the flow — and you find the break. Chapter 10 turns this walk into concrete playbooks; for now, just notice that the fabric's structure and its troubleshooting method are the same thing seen from two directions.

A first health glance on a fabric edge — is the control plane relationship up? (illustrative, fictional output)

```
Edge-1# show lisp session
Sessions for VRF default, total: 2, established: 2
Peer                    State     Up/Down      In/Out
192.0.2.11                Up        3d04h      128/141     ! control plane node
192.0.2.12                Up        3d04h      126/139     ! control plane node (redundant)

Edge-1# show device-tracking database | include 10.40.
10.40.0.57   aa11.bb22.cc33   GigabitEthernet1/0/12   VLAN 1040   REACHABLE   PLANT-FLOOR
```

▲ Common Problem — Old DNA Center tutorials will show you the wrong menus

A lot of freely available material was written for early DNA Center releases, and both the GUI and the fabric hardware have moved on. Menus have been renamed and reorganized under Catalyst Center, and features that once needed CLI are now driven from the controller. When a guide tells you to click something that is not there, the usual cause is version drift — check which release the guide targets before assuming you have done something wrong. Treat the *concept* as stable and the exact click-path as something to confirm against the release in front of you.

▲ Watch out — Not every switch can be a fabric node

SD-Access fabric roles run on specific, fabric-capable platforms — principally the Catalyst 9000 switching family — and the fabric and automation features are gated behind the appropriate software licensing tier. Part of your design work in Chapter 3 is confirming that Northforge's existing hardware and licenses can play the roles you need, and identifying what must be upgraded or added before the build. Discovering a wiring closet full of switches that cannot join the fabric on cut-over night is exactly the surprise this planning exists to prevent.

◆ From the Field — You will live in the GUI, but the CLI still tells the truth

Day to day, you will design and operate the fabric from Catalyst Center, and that is the point — intent in one place, pushed consistently everywhere. But when something misbehaves, the switch CLI and ISE's live logs remain the ground truth: a quick check of the LISP session, the device-tracking table, or an ISE authentication record often finds in seconds what a dashboard only hints at. Learn to trust the controller for building and the CLI for confirming.

Chapter Recap

SD-Access is the campus fabric solution, run by two brains over a fabric of role-playing devices. **Catalyst Center (formerly DNA Center)** designs, automates, provisions, and assures; **ISE** authenticates, profiles, groups, and holds group-based policy; the two share state over pxGrid. The fabric devices play roles — control plane node (the location directory), border node (the doorway out), edge node (where endpoints connect), fabric WLC and APs (wireless into the fabric), and extended nodes (reach into places like the plant floor). Endpoint onboarding — authenticate, group, register, encapsulate, enforce — is both how the fabric works and how you troubleshoot it.

1.4 Meet the Project: Northforge Industrial

Every design decision in this book is anchored to one company, so that the trade-offs are concrete rather than abstract. This section is the project brief you would assemble after your first week of discovery at Northforge — the sites, the endpoints, the network as it is today, and the mandate you have been given. Keep it in mind; later chapters constantly refer back to "what Northforge needs."

The company and its sites

Northforge Industrial makes precision components for the automotive and aerospace supply chain. Around 1,800 employees are split between the office and the shop floor, and the network has to serve both worlds at once — quiet, cloud-heavy office traffic alongside sensitive, latency-critical, safety-relevant industrial traffic. There are four kinds of site:

- **Headquarters campus (HQ).** Two office buildings — Admin and Engineering — plus an attached production hall, **Plant 1**. This is the largest, most complex site and the center of the project. A brand-new building, **Plant 1B**, is under construction next to it and will be the book's **greenfield** fabric.

- **Plant 2.** A second manufacturing site in another city, smaller than HQ but with the same office-plus-shop-floor mix. It will become its own fabric site and later be joined to HQ across the WAN.
- **Distribution warehouse (DW).** High-bay racking, wall-to-wall wireless, handheld scanners, and a warehouse-management system. Heavily wireless, light on wired users.
- **Sales offices.** A handful of small offices, some of them a single wiring closet, that need the same policy as everyone else without a rack of fabric gear — candidates for a compact, single-box fabric design.

The endpoints you must serve

The reason SD-Access fits Northforge so well is the sheer variety of things that plug in, each needing a different level of trust. Cataloguing them now makes the segmentation design later almost write itself:

- **Corporate users** — laptops and desktops, mostly cloud and Microsoft-365 traffic, increasingly wireless and mobile across buildings.
- **Voice** — IP phones and soft clients that need priority and their own reachability rules.
- **Engineering workstations** — powerful machines that need to reach both corporate services and, in controlled ways, the plant-floor data systems.
- **Plant-floor OT** — PLCs, human-machine interfaces (HMIs), robotic cells, machine-vision cameras, and industrial sensors. Latency-sensitive, rarely patched, long-lived, and never meant to touch the Internet or corporate IT directly.
- **Building and physical security** — IP cameras, badge readers, HVAC and building-management sensors. Low-trust IoT that should reach its own controllers and nothing else.
- **Guests and contractors** — visitors and third-party machine vendors who need Internet or a single specific system, and must be isolated from everything else by default.

The network as it is today (brownfield)

Northforge's current campus is a textbook three-tier design that has grown organically for a decade. Each building has its own access and distribution layers; VLANs are stretched where it was convenient; STP and HSRP hold it together; and segmentation is a large, fragile collection of IP ACLs on the distribution switches. Crucially, IT and OT are **not** cleanly separated: a few plant VLANs were bridged into the corporate network years ago so that a historian server could collect production data, and no one has dared unpick it since. Configuration is per-switch and manual, guest access is improvised, and visibility is whatever an engineer can piece together by logging into devices one at a time.

Site	Today	Target design
HQ — Plant 1B (new)	Empty; under construction	Greenfield fabric site, built as SD-Access from day one
HQ — Admin, Eng, Plant 1	Three-tier, VLAN/STP, IP ACLs, IT/OT partly bridged	Brownfield migration to a single HQ fabric site
Plant 2	Three-tier, standalone	Its own fabric site, later joined to HQ over the WAN
Distribution warehouse	Flat, heavily wireless	Fabric site with fabric-enabled wireless
Sales offices	Single closet each, manual	Compact single-box fabric (“fabric in a box”)

The mandate

Stripped of buzzwords, Northforge's leadership has asked for six outcomes. Every one of them maps directly onto an SD-Access capability, and together they define what “done” means for this project:

1. **Separate IT and OT for real** — a hard boundary between the plant floor and corporate IT that a compromised office machine cannot cross, plus finer rules within each world — and do it **without** re-addressing the whole network.
2. **Onboard any endpoint consistently and automatically**, wired or wireless, so that adding a production line or a floor of desks is a policy action, not a week of manual switch edits.

3. **Contain IoT and OT** so that a compromised camera or sensor cannot move laterally to anything of value.
4. **Give guests and contractors isolated access** by default, with third-party vendors reaching only the one system they came to service.
5. **Provide central visibility and faster troubleshooting**, replacing box-by-box guesswork with a single source of truth and real telemetry.
6. **Deliver both a greenfield and a brownfield path** — stand up Plant 1B as fabric from day one, and migrate the live HQ campus onto the fabric with minimal disruption to a running factory.

> ✔ **Design Tip — Start from the policy, not the boxes**
> The most common way these projects go wrong is to treat SD-Access as a switching upgrade — buy the hardware, build the fabric, and only then wonder what the segmentation should be. Do it the other way around. Northforge's real deliverable is item 1 on the mandate: a clear map of which groups exist (Corporate, Voice, Engineering, Plant-OT, Building-IoT, Guest, Contractor) and which may talk to which. Nail that policy intent first, and the fabric, the virtual networks, and the address pools become straightforward expressions of it. Design the badges before you rebuild the doorways.

> ◆ **From the Field — Why a manufacturer is the ideal first fabric**
> Office-only campuses can coast on “everyone is roughly corporate.” A manufacturer cannot: the gap in trust between a finance laptop and an unpatched PLC is enormous and consequential, which forces you to use every part of SD-Access properly — macro-segmentation to wall off OT, micro-segmentation to contain IoT, identity-based onboarding for devices that cannot log in, and assurance to prove it all works. If you can design Northforge's fabric, an ordinary office campus will feel easy. That is exactly why we chose it.

Where the book goes from here

With the problem, the ideas, the platform, and the project in hand, the rest of the book follows the natural arc of a real deployment. Chapter 2 opens the machinery — underlay and overlay, LISP, VXLAN, TrustSec, and the fabric roles — so the mechanics stop being acronyms. Chapter 3 turns Northforge's mandate into a High-Level Design. Chapter 4 stands up the controllers and the underlay; Chapter 5 brings sites online, greenfield and brownfield. Chapters 6 through 8 layer on

segmentation, wireless, and security. Chapter 9 extends the fabric across sites and links it to the WAN, and Chapter 10 operates, automates, and troubleshoots it. You will finish with a network you could hand to Northforge's team and a method you could repeat anywhere.

Chapter Recap

Northforge Industrial is a mid-size manufacturer with an office-plus-shop-floor HQ campus (including a new building for greenfield), a second plant, a warehouse, and small sales offices, carrying everything from finance laptops to unpatched PLCs on a flat, manually operated three-tier network with IT and OT dangerously intertwined. Leadership's six-point mandate maps one-to-one onto SD-Access. The guiding discipline is to design the policy first and let the fabric express it. From here, the book follows the deployment arc from design to day-two operations.

◆ ◆ ◆

END OF FREE SAMPLE

You have read Chapter 1 in full. The complete book continues the Northforge Industrial project from design to day-two operations:

- Chapter 2 — How the fabric works: underlay/overlay, LISP, VXLAN, TrustSec, and the roles
- Chapter 3 — Designing the fabric: discovery, roles, segmentation, addressing, transit, identity, HLD
- Chapters 4–5 — Standing up the foundation and bringing sites online (greenfield + brownfield)
- Chapters 6–8 — Segmentation & group policy, fabric wireless, and securing the fabric
- Chapter 9 — Multi-site, transit, and the SD-Access ↔ SD-WAN handshake
- Chapter 10 — Operating, automating, and troubleshooting the fabric
- Appendices — reference design, command reference, deployment checklists, and a glossary

144 pages · original diagrams · a complete, worked implementation

SDA Fundamentals

Get the complete book at

leanpub.com/sda-fundamentals