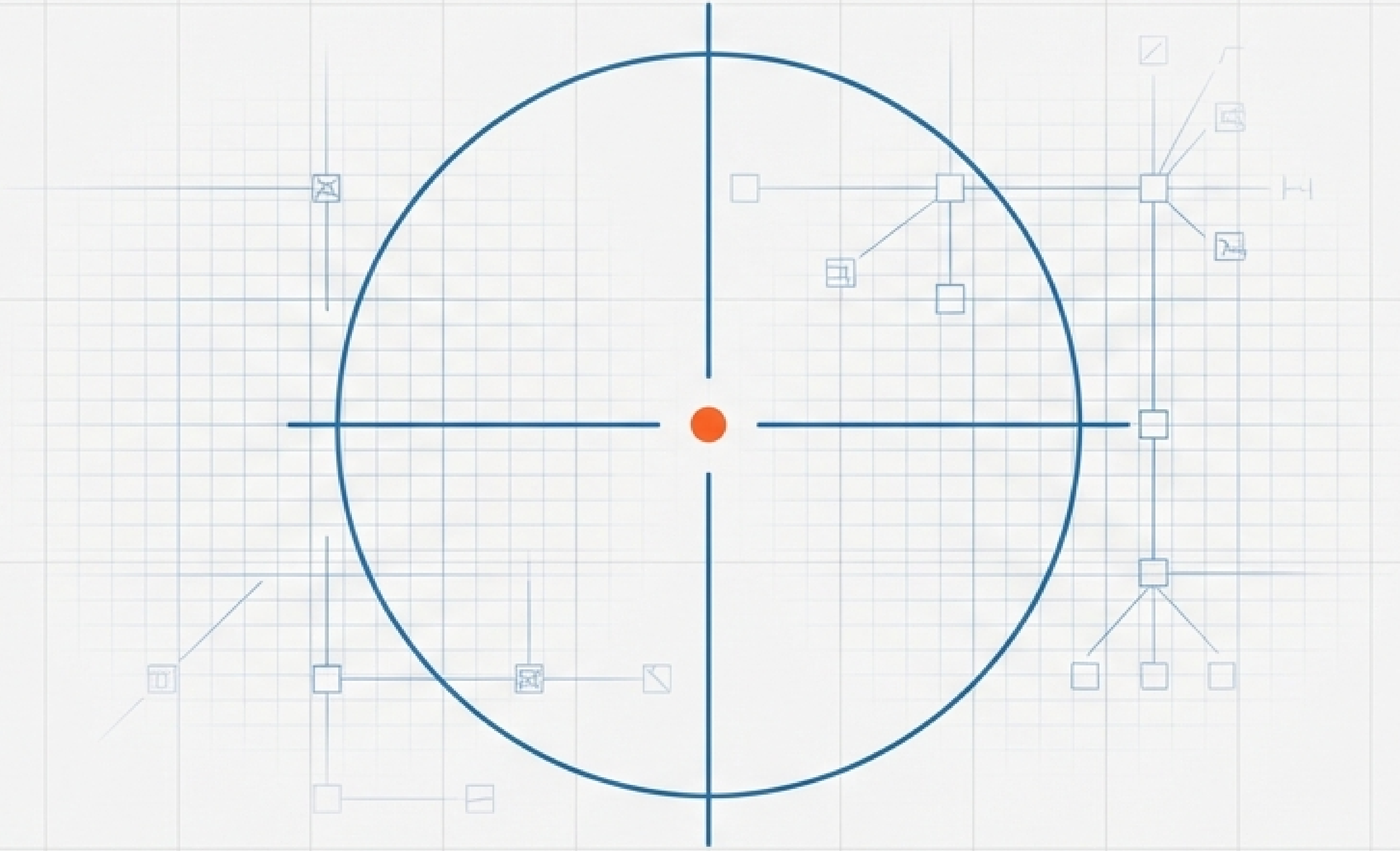
THE NETWORK ARCHITECT'S GUIDE TO STRUCTURED TROUBLESHOOTING

Moving from Guesswork to Evidence-Based Isolation



JetBrains Mono

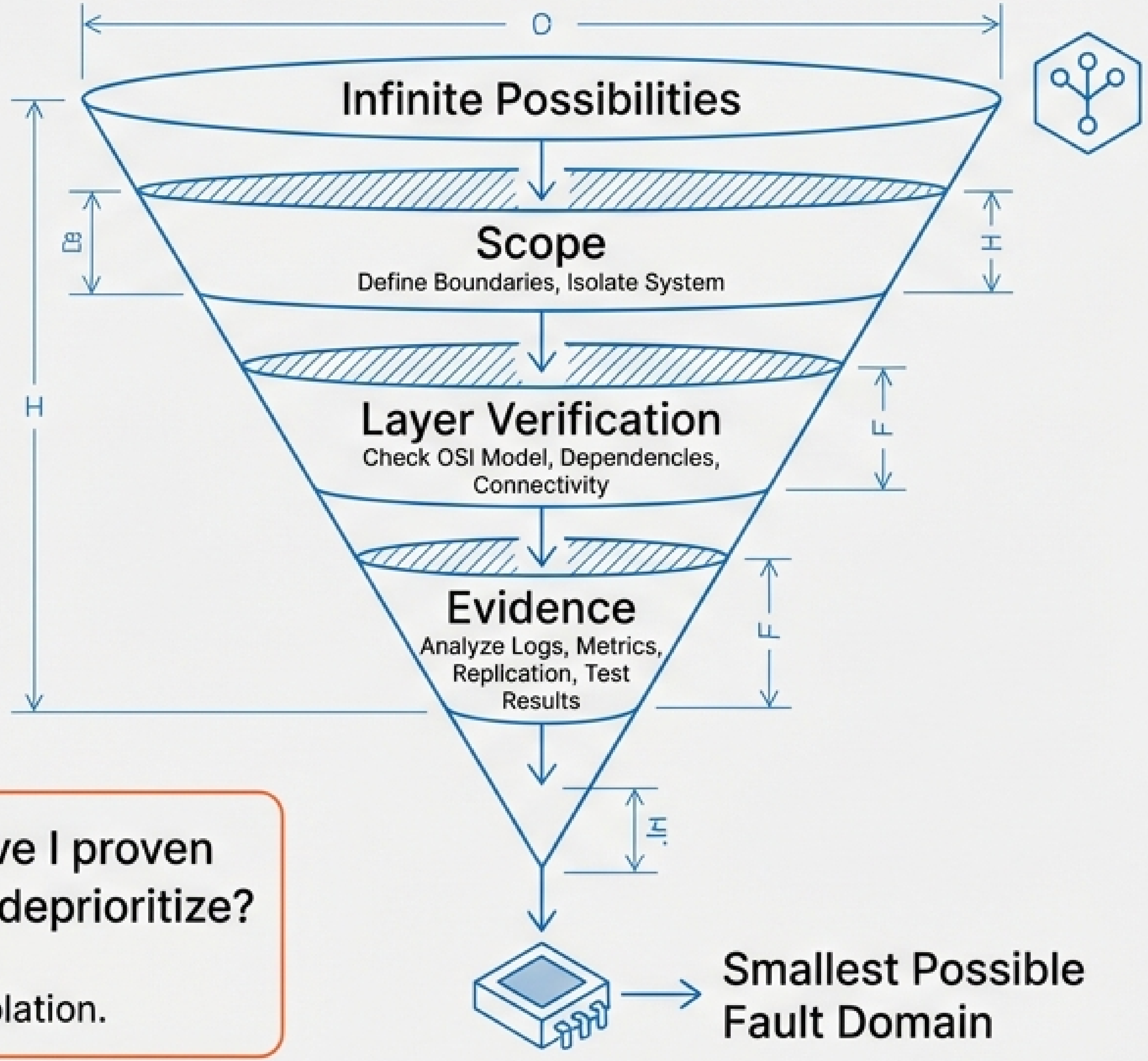
For Network Engineers, Architects, and Certification Candidates

Troubleshooting is the Controlled Reduction of Possibilities

The Amateur Approach

```
Trying Commands.
Random inputs,
hoping for a fix.
```

The Architect Approach



Infinite Possibilities

Scope
Define Boundaries, Isolate System

Layer Verification
Check OSI Model, Dependencies, Connectivity

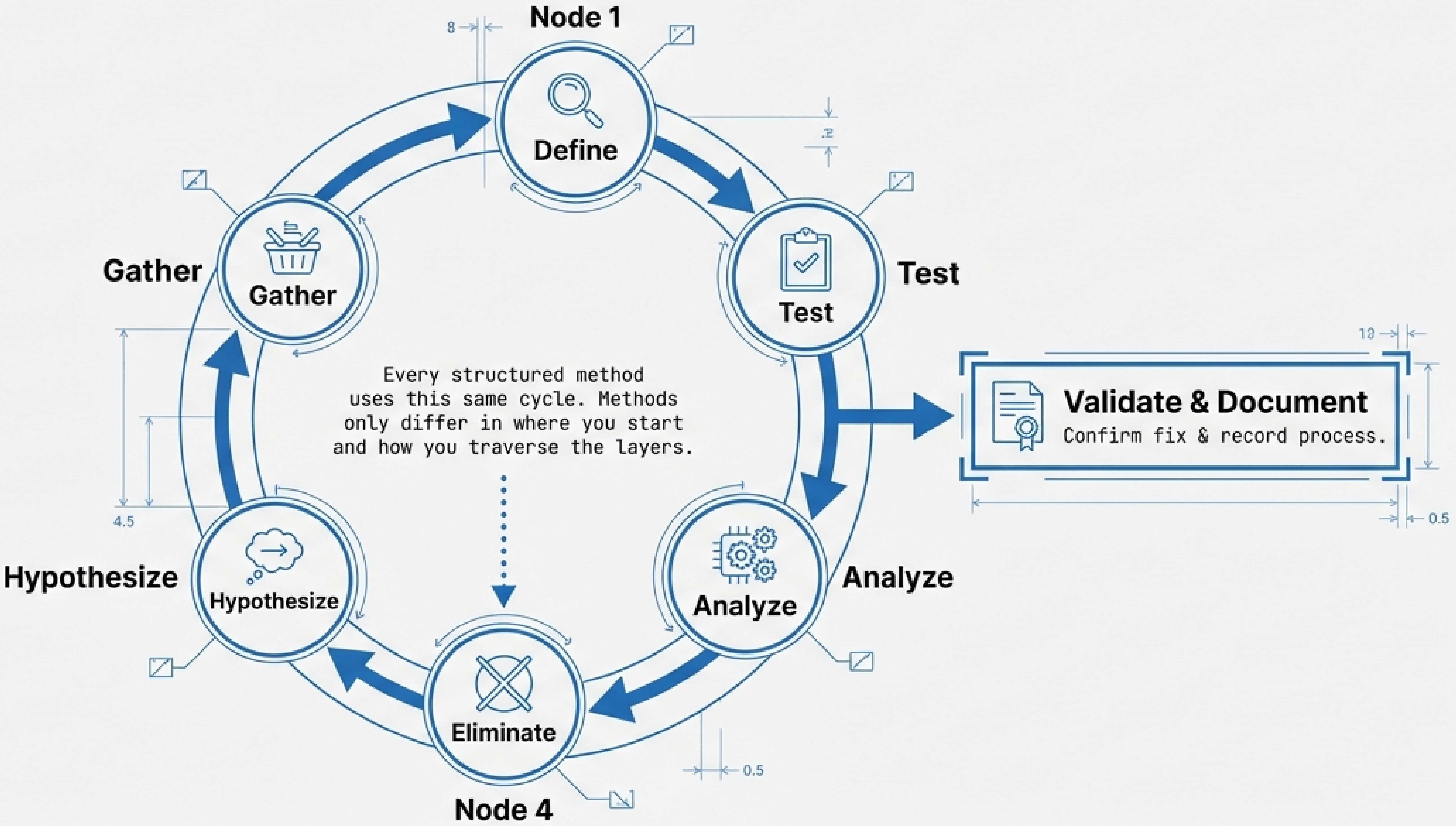
Evidence
Analyze Logs, Metrics, Replication, Test Results

Smallest Possible Fault Domain

Key Question: What have I proven works, and can therefore deprioritize?

Crucial for efficient isolation.

The Universal Troubleshooting Loop



Every structured method uses this same cycle. Methods only differ in where you start and how you traverse the layers.

Validate & Document

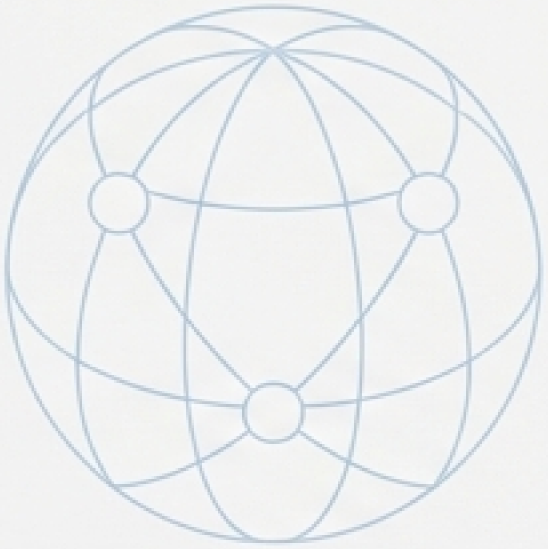
Confirm fix & record process.

Lock Down the Scope Immediately

Accurate scoping prevents wasting cycles on the wrong network layer.

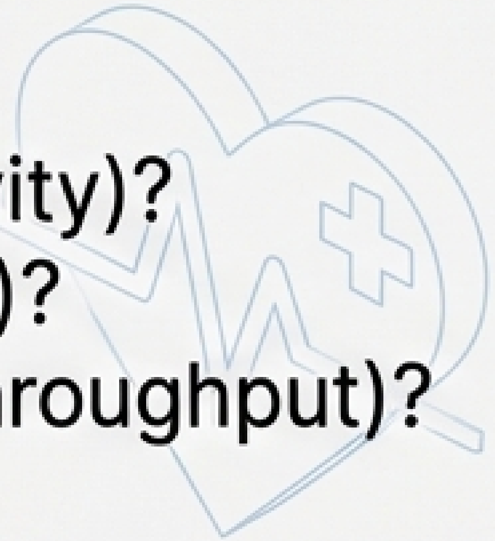
Impact

- Single host?
- Subnet/VLAN?
- WAN/Global?



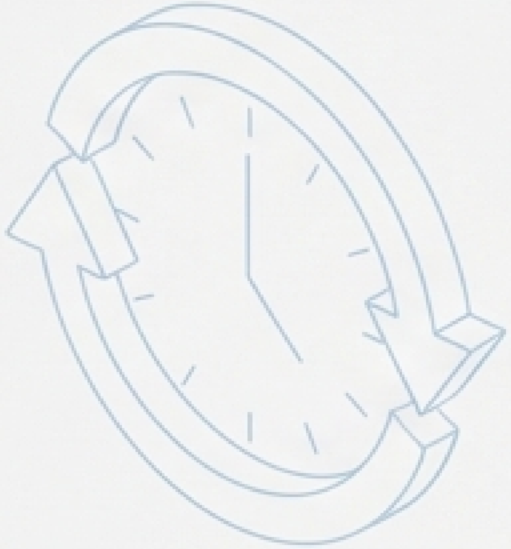
Symptom

- Hard down (No connectivity)?
- Intermittent (Flaps/Drops)?
- Performance (Latency/Throughput)?



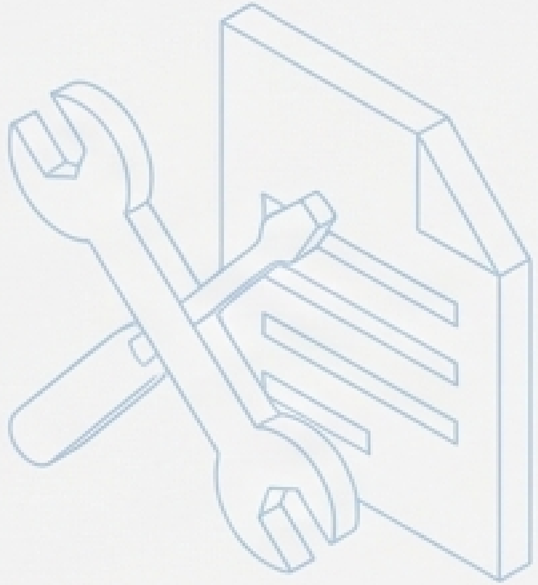
Reproducibility

- Consistent?
- Time-based?
- Load-based?

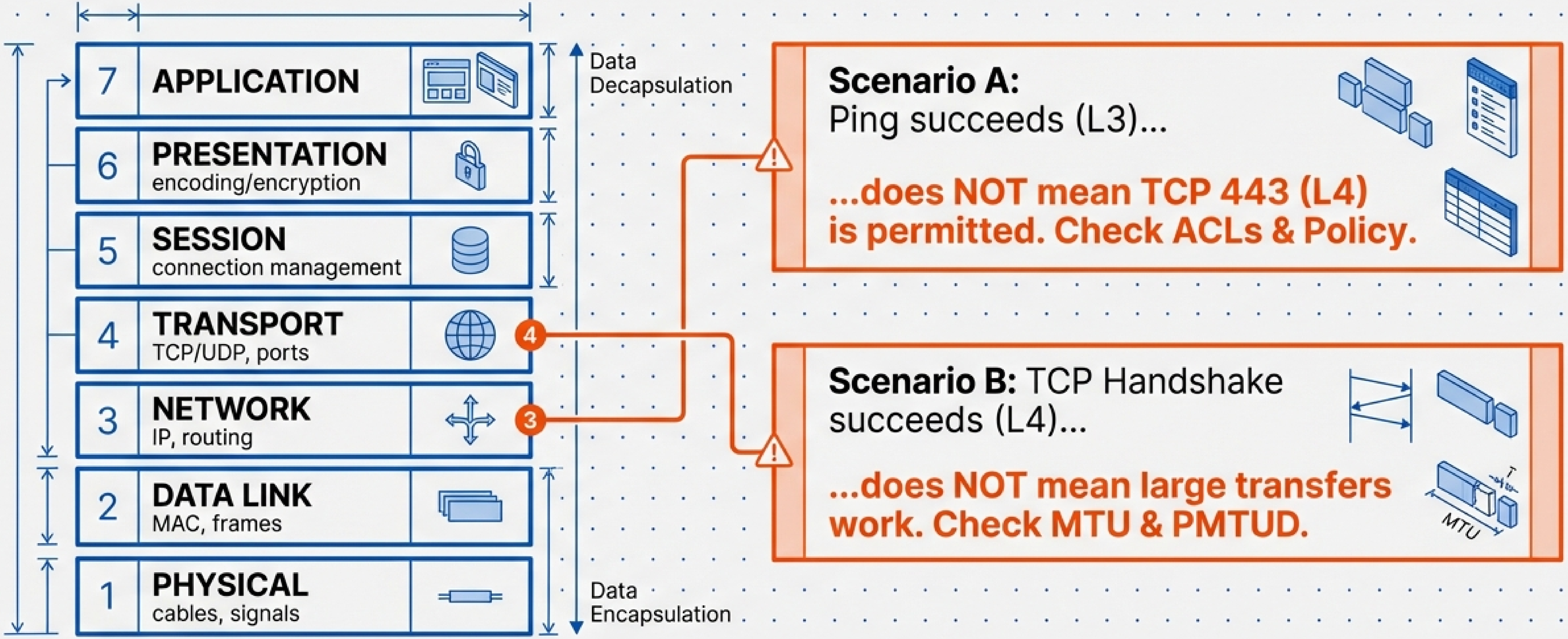


Recent Changes

- Config change?
- Code upgrade?
- Cabling?
- Policy push?



Mapping Tests to the OSI Model



7 APPLICATION
6 PRESENTATION encoding/encryption
5 SESSION connection management
4 TRANSPORT TCP/UDP, ports
3 NETWORK IP, routing
2 DATA LINK MAC, frames
1 PHYSICAL cables, signals
Data Decapsulation
Data Encapsulation
4
3
MTU

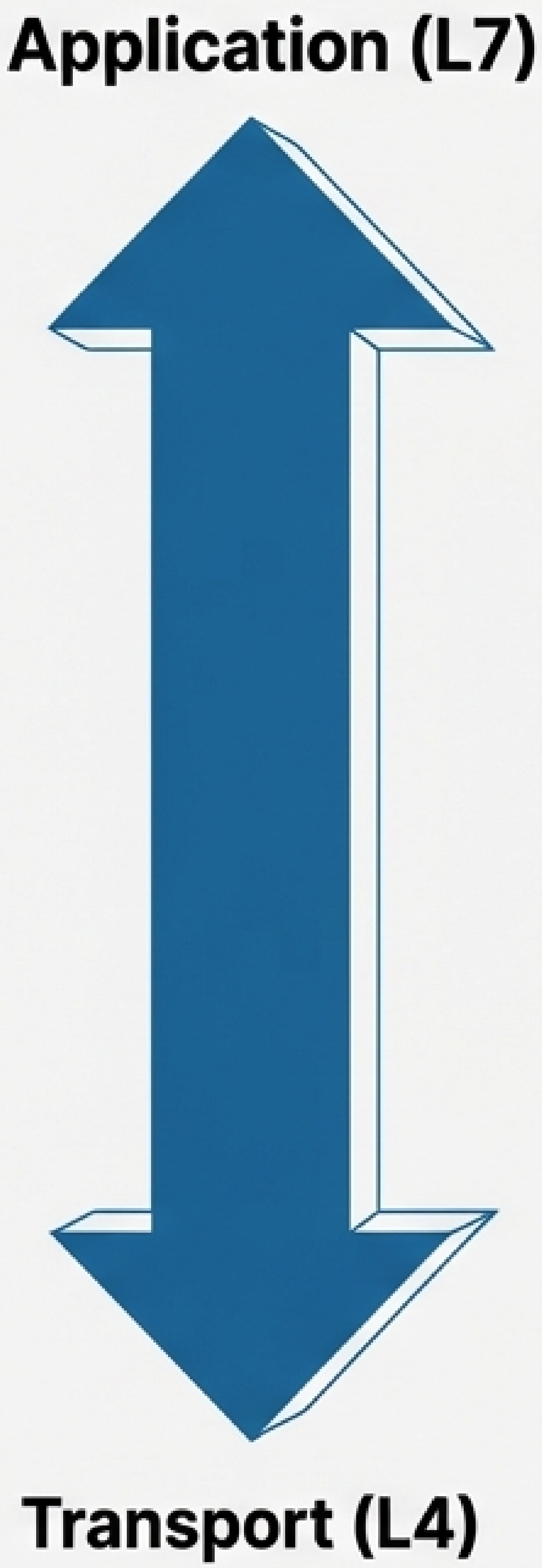
Scenario A:
Ping succeeds (L3)...

...does NOT mean TCP 443 (L4) is permitted. Check ACLs & Policy.

Scenario B: TCP Handshake succeeds (L4)...

...does NOT mean large transfers work. Check MTU & PMTUD.

Engineering Insight: A successful test only proves THAT specific thing works, not that the whole system is fine.



Method 1: The Top-Down Approach

When to use

Helvetica Now Display

- User reports App symptoms (Web/Email/DNS).
- Lower layers (Link/Routing) appear clean.

Goal: Prove issue is above Layer 4.

If L4 session establishes, focus on: DNS, TLS/Certs, Proxy, Server Health.

Execution

Helvetica Now Display

Endpoint Checks

Helvetica Now Display Medium

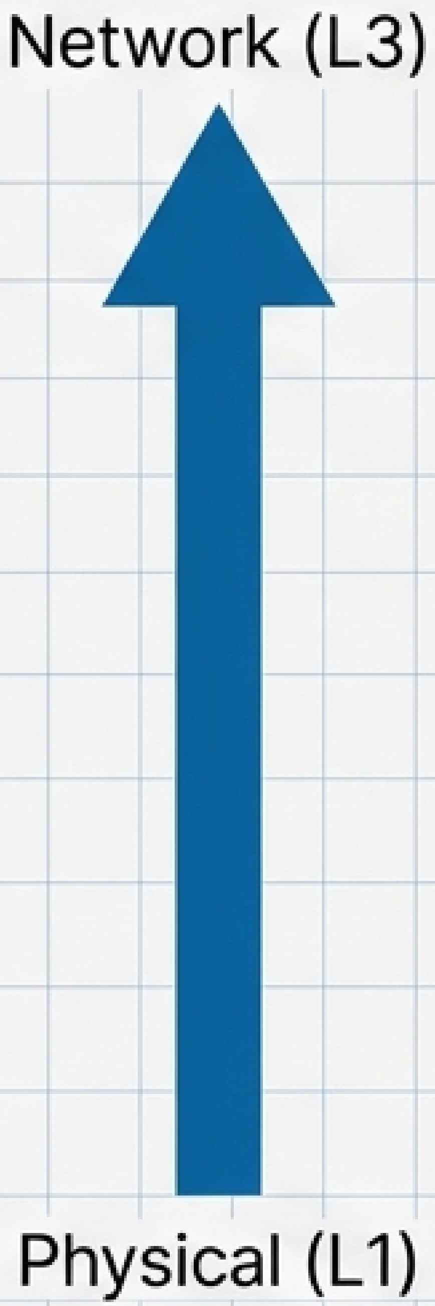
```
DNS Resolution
TCP Port Reachability
HTTP Status
```

Network Commands

Helvetica Now Display Medium

```
show ip nat translations
show access-lists
show logging
```

Helvetica Now Display
Method 2: The Bottom-Up Approach



When to use

- ✓ Link flaps, Error increases, Cabling changes, “Slow network” reports.

Eliminate physical faults before troubleshooting routing protocols.

Execution

L1 (Physical)

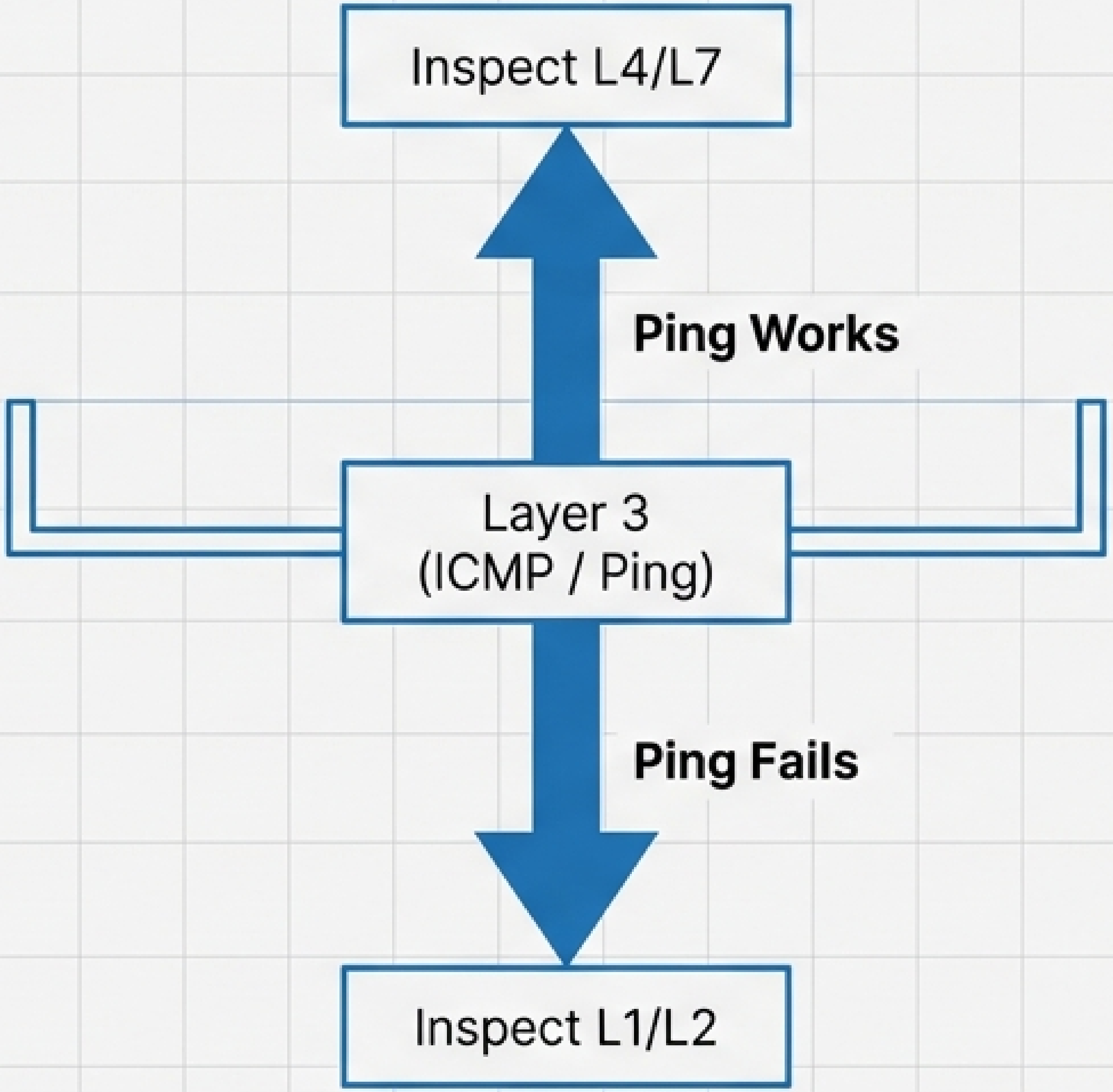
```
show interfaces status
show interfaces counters errors
(Look for CRC, Input Errors, Late Collisions)
```

L2 (Switching)

```
show vlan brief
show interfaces trunk
show spanning-tree interface
```

Method 3: Divide-and-Conquer

The best first move for general incidents.



Staged Testing

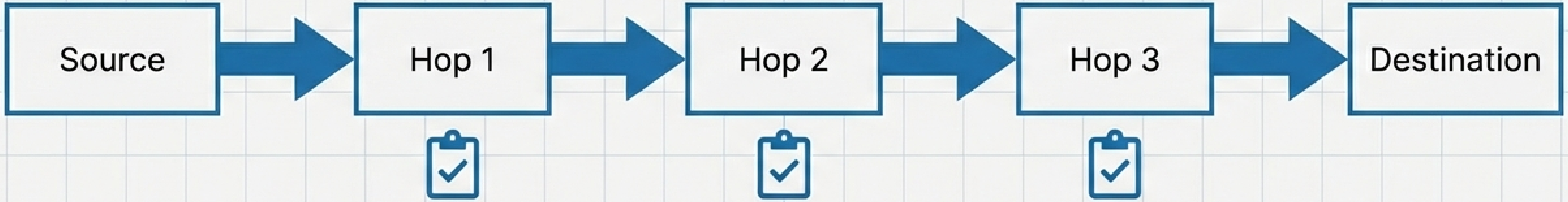
```
1. ping gateway
2. ping next hop
3. ping WAN edge
```

⚠ Critical Nuance

A single failed ping != Routing Broken.

- Reasons: ICMP Filtering
- Asymmetric paths
- VRF Mismatch

Method 4: Follow-the-Path



Use Case:

Best for Multi-hop problems (Campus + WAN + DC), Asymmetric paths.

Process:

1. Identify expected path (Traceroute).
2. Verify each hop: Interface health, L2 adjacency, L3 forwarding.

Command Toolkit

```
traceroute | show ip route | show cdp neighbors detail | show ip nat translations
```

Edge Tactics: Swap & Compare

Swap Components

For Hardware/PHY Faults (Single User/Port)

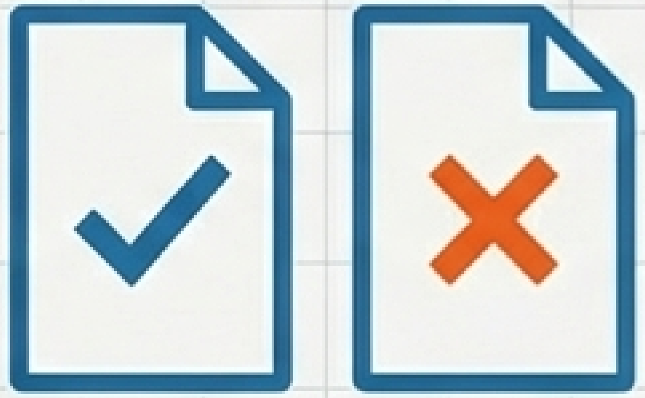
⚠ The Rule

Swap one variable at a time (Cable, Port, SFP).

Validation

Document change and re-test.

Perform Comparison



For Templated Sites (Baseline vs. Broken)

Action

Compare working config against broken config.

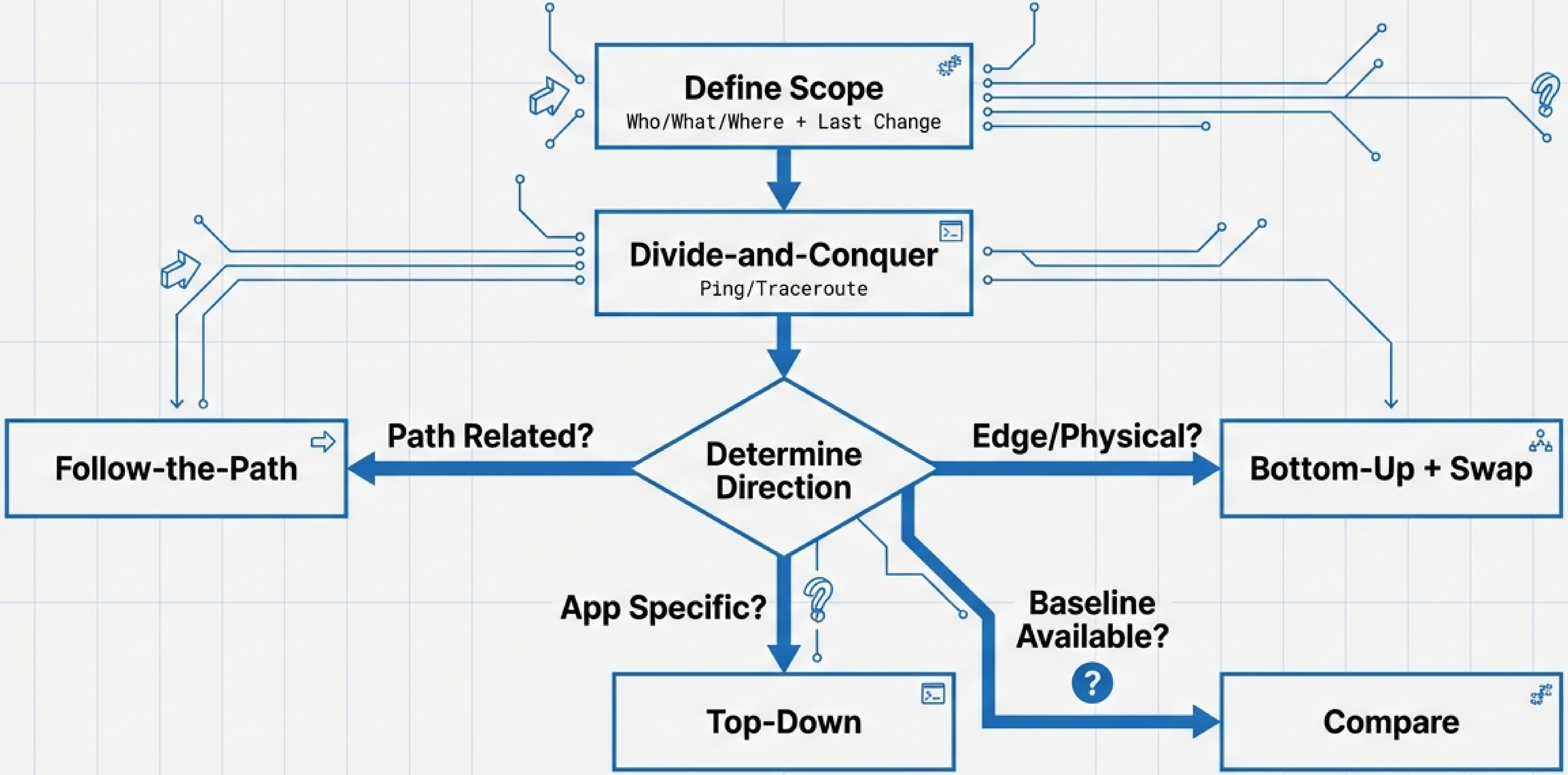
Commands

```
show run interface | show run |
section | show version
```

The Strategic Selector: Choosing the Right Method

Symptom		Primary Method
App Issue / DNS / Auth	→	Top-Down
Link Flaps / CRCs / “Slow”	→	Bottom-Up
“Isolate Fast” / Unknown	→	Divide-and-Conquer
Multi-hop / Packet Loss	→	Follow-the-Path
Single User / Hardware	→	Swap Components
Templated Site Issue	→	Perform Comparison

The First 5 Minutes: A Practical Runbook



CLI Toolkit: Layers 1 & 2 (Physical & Data Link)

Layer 1 (Physical)

- `show interfaces status` → (Check for Err-Disable)
- `show interfaces counters errors` → (Check for CRC, Input errors)
- `show logging | include down` → (Check timestamps)

Layer 2 (Switching)

- `show vlan brief` → (Check membership)
- `show interfaces trunk` → (Check native VLAN/allowed list)
- `show spanning-tree` → (Check blocking states)
- `show mac address-table` → (Check learning)

CLI Toolkit: Layer 3 & Policy Edges

Layer 3 (Network)

```
show ip interface brief
```

→ (Correct addressing)

```
show ip arp
```

→ (Resolution to next hop)

```
show ip route
```

→ (Forwarding decisions)

```
show ip cef
```

→ (Forwarding plane verification)

Policy & Edges

```
show access-lists
```

→ (Hit counters)

```
show ip nat translations
```

→ (Translation creation)

```
show policy-map interface
```

→ (QoS drops)

Exam Heuristics & Final Thoughts

Prompt: “Application not working”

→ **Top-Down**

Prompt: “CRC / Duplex / Link Flaps”

→ **Bottom-Up**

Prompt: “Fastest Isolation”

→ **Divide-and-Conquer**

Prompt: “Multiple Routers / Isolate along route”

→ **Follow-the-Path**

VALIDATE. DOCUMENT. REPEAT.

Every successful troubleshoot ends with evidence, not guesses.