Achievable

CCNA DUMP SHEET

Automation and programmability

How automation changes management

- **Manual CLI:** device by device: human error, drift, slow change
- **Automation:** define a task once; software applies it everywhere
- **Gains:** consistency, speed, scale without matching staff
- **ZTP (Zero Touch Provisioning):** devices configure themselves

The three planes

Plane	Job	Examples
Management	Manages the device	SSH, SNMP
Control	Decides the path	OSPF Hellos, STP BPDUs
Data	Forwards packets	Layer 2 rewrite, QoS

APIs

Northbound: to controller	Southbound: to devices
Apps request services, apply policy; device detail hidden	Controller configures, monitors, collects data
Catalyst Center (was DNA Center) REST APIs, OpenDaylight	OpenFlow, NETCONF (XML), RESTCONF (HTTP, YANG), CLI over SSH

REST and CRUD

- **Traits:** client/server, stateless, cacheable, uniform interface
- **Stateless:** each request stands alone; no client context kept

CRUD	Verb	Note
Create	`POST`	New resource
Read	`GET`	Safe, idempotent, changes nothing
Update	`PUT`	Replaces the whole resource. Idempotent
Update	`PATCH`	Changes only the fields sent
Delete	`DELETE`	Idempotent

- **URI:** `https://<host>/dna/intent/api/v1/network-device`: scheme, host, resource path

JSON

One valid object	Component
`{ "hostname": "SW1",`	Object; key is a quoted string
`"vlan": 10,`	Number, no quotes
`"is_active": true,`	Boolean, lowercase
`"contact": null,`	Null, lowercase
`"ports": ["g0/1", "g0/2"],`	Array: ordered values
`"mgmt": {"ip": "10.1.1.1"} }`	Nested object; no last comma

- **Invalid:** single quotes, unquoted keys, `True`, trailing comma

Traditional vs controller-based

	Traditional	Controller (Catalyst Center)
Control plane	On every device	Mostly on the controller
Data plane	On every device	Stays on the devices
Configure	Device by device	Policy; controller translates

Underlay, overlay, and fabric

Underlay	Physical network: switches, routers, IP reachability
Overlay	Virtual on top: VXLAN tunnels, segments, policy
Fabric	Both together, coordinated by the SDN controller

- **Cisco:** Software-Defined Access (SDA), SD-WAN

AI in network operations

Predictive AI: forecasts	Generative AI: creates
Trends, classifications	Text, code, configurations
Traffic forecasts, anomaly detection, predictive maintenance, capacity planning	Config generation, troubleshooting steps, security policy, test traffic, docs
In Catalyst Center, Crosswork	In AI Canvas

- **Machine learning:** AI subset, learns from data. Powers both

REST auth, encoding, and status codes

Basic	Username, password in Base64; use over HTTPS
API key	Unique client key in a header or query parameter
Token	Bearer token after login, often a JWT. Expires
OAuth 2.0	Delegated access without sharing credentials

- **Data encoding:** JSON most common, XML older
- **Status:** 2xx success, 4xx client error, 5xx server error
- **2xx:** 200 OK, 201 Created, 202 Accepted, 204 No Content
- **4xx:** 400 Bad Request, 401 Unauthorized (authenticate), 403 Forbidden (refused), 404 Not Found
- **5xx:** 500 Internal Server Error, 501 Not Implemented, 502 Bad Gateway, 503 Service Unavailable, 504 Gateway Timeout

Ansible and Terraform

	Ansible	Terraform
Focus	Configuration, day 1 on	Provisioning (IaC), day 0
Use	VLANs on switches	New virtual routers
Model	Push from control node	Declarative, state file
Agents	Agentless: SSH, WinRM	Agentless: provider APIs
Language	YAML playbooks	HCL
Idempotent	By design	By state management