Common NFVI Telco Taskforce

Technical F2F Work Shop – January 13-16, 2020

Infrastructure Description (Manifests) Validation

Facilitator: Sridhar Rao

THE LINUX FOUNDATION

GSMA

The ‘Manifest Validation’ is a novel process – there does not exist any reference work or tools in public.
It has a major role to play in ‘automation’.

Infrastructure Description

- Machine-Readable Definition/Description of the infrastructure – Hardware and Software – using a specific schema (preferably standardized?).
- The Infrastructure: Environment In/On which 'ANY'/'Targeted' VNFs (CNFs) runs.
- Scope
  - NFVI + SDN
  - MANO
  - Underlay/Fabric?
- Consumers
  - Primary: Installers.
  - Secondary: Test and Verification Framework, Application designers, Management/Operations.
-

What really user can 'Describe'?

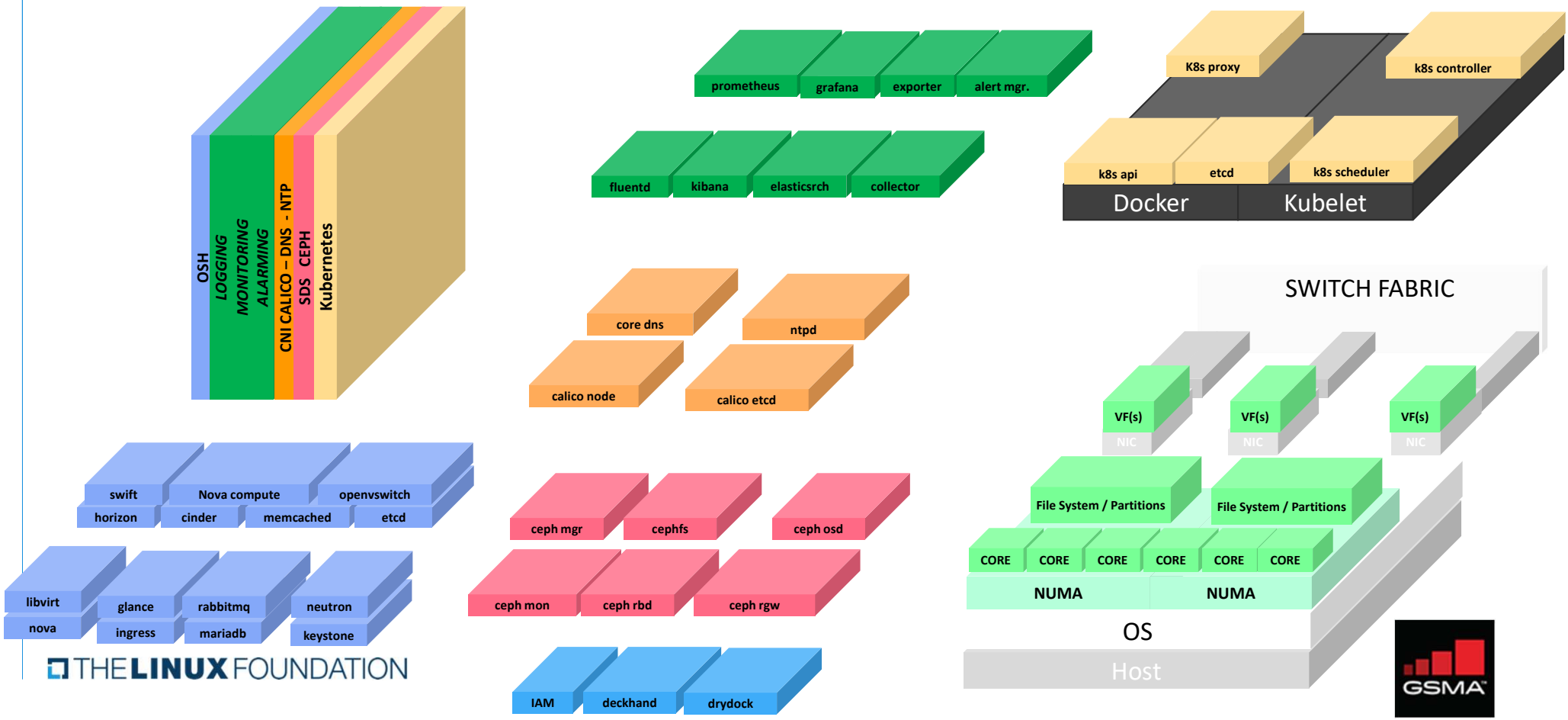
› These are organized by Installer-Specific Schema

GENERIC: Management (location, owner, etc.), Strategy, Globals, Tooling, Versioning,	HARDWARE: Vendor, Gen., BIOS, CPUs, Memory, Disks, NICs (PCI, MAC), etc.	PROFILES (HOST) Name, Disks & Partitions, N/W->NIC Mapping, OS,
NETWORK Names, vlans, cidr, routes, ip, g/w, speed, mtu, bonding, etc. cidrs for diff. n/ws, container n/w, SRIOV, etc.	NODES profile-mapping, Networks and address, metadata.	SECRETS Certificates, passphrases, publickeys, etc.
SOFTWARE Software and nodes mapping, versions, registry,	ACTIONS Bootactions- custom scripts, drivers, etc.	OTHERS Jumphosts-Defn., Network Services (NTP, DNS, etc)

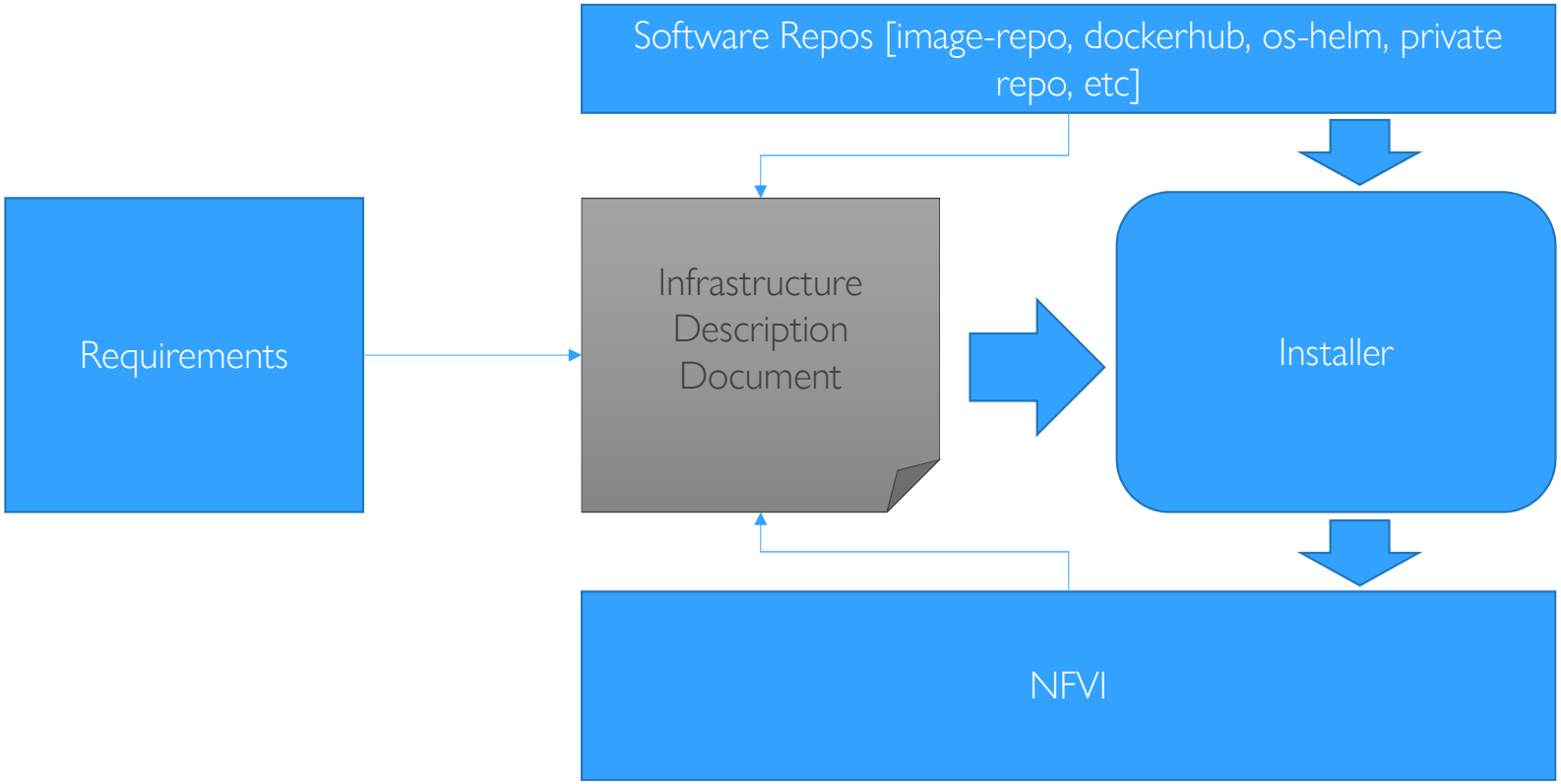
Infrastructure Description

- Example Works
  - Airship Treasuremap Manifests
  - OPNFV PDF/IDF
  - TripleO Heat Templates
  - Kayobe's YAML files.
  - Fuel Configuration in OPNFV-IDF.
  - GUI-Based configuration in compass.
  - OPNFV Apex's - inventory, network and deploy settings.
  - Kubernetes CRDs

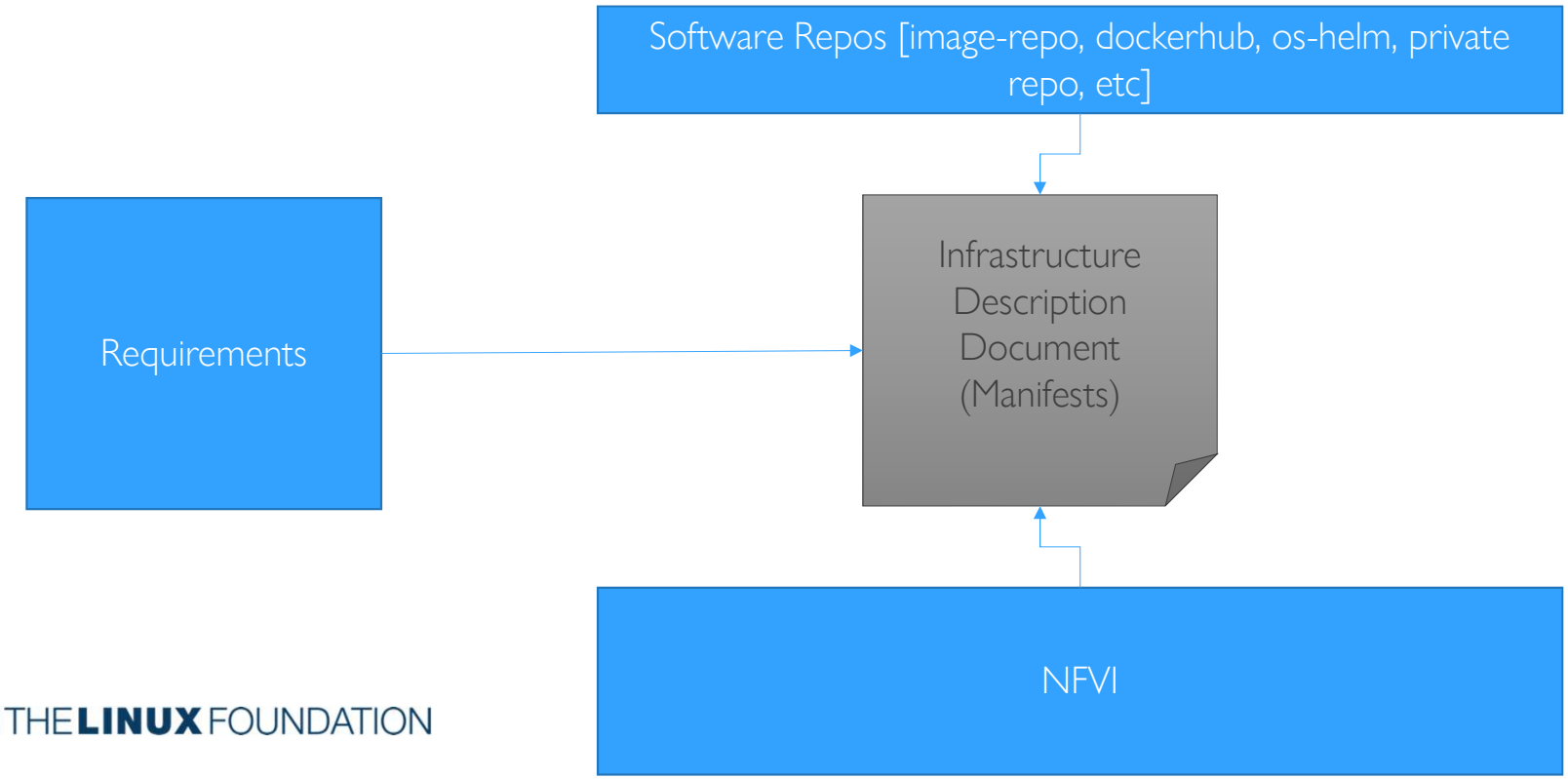
Infrastructure Description: Example (Airship)



Infrastructure Description: Manifests



Validating Manifests: What are we validating against?



Validating Manifests Vs Validating Deployment

› Overlaps does exist….

Category	Manifest Validation	Deployment Validation
Scope	Applies to only a subset of requirement	Validation covers all the requirements
Source	Documents	Physical Systems
Phase	Pre-Deployment [Cloud and Application]	Post-Deployment
Approach	Manual/Scripts	Test-Frameworks, Automation Scripts
Stakeholders	Cloud Architect, Application designers and Testers	

Validating Manifests: Why and How

- Why
  - Pre-Installation Checks for RM/RA
  - Minimize/Eliminate deployment failures.
  - Drive test-automation
  - Consistency Check for efficient automation
  - 'Handoff' to RC
- How?
  - Manual
  - Automated: Scripts

Ex: Considering CNTT Requirements

Requirements (chapter-5)

Requirement	Basic	NI	CI	POD-10
Number of CPU (Sockets)	2	2	2	2
Number of Cores per CPU	20	20	20	22
NUMA	N	Y	Y	Y
Simultaneous Multithreading/Hyperthreading (SMT/HT)	Y	Y	Y	Y
GPU	N	N	Y	Y (glxinfo | grep "direct rendering")
Local Storage HDD				
Local Storage SSD	Y	Y	Y	Y
NIC Ports	4	4	4	4
Port Speed	10	25	25	10
PCIe slots	8	8	8	8 (dmidecode -t slot)
PCIe speed	Gen3	Gen3	Gen3	Gen3
PCIe Lanes	8	8	8	8
Cryptographic Acceleration	N	O	O	N
SmartNIC	N	O	O	N
Compression				

Requirements (chapter-2)

- Categories
  - Opensource
  - Cloudnativeness
  - Scalability
  - Resilience
  - Availability
  - Compute
  - Storage
  - Network
  - Acceleration
  - General
  - API
  - Automated Deployment
  - CI/CD+
  - Integration+
  - Monitoring
  - Zoning
  - Compliance
  - Networking*
- Requirement-Level
  - Must
  - Should
  - May

- Pod-10 (ex. Validation)
- [separate xcel sheet]

THE LINUX FOUNDATION

GSMA

Validation Approach & Classification of Requirements

Validation Approach	Requirement Type
Testing Framework	APIs, Openstack-Features, Operations,
Automation Scripts	Configurations and Settings, System Capabilities,
Manual	Rest..

Thanks

THE LINUX FOUNDATION

GSMA