

# Tail-f Network Control System (NCS) – Datasheet

Multi-vendor Service Orchestration and Network Automation

## Provision Network Applications and Services - With Transactions

**For Service Providers, the network is responsible for driving revenue streams. For Enterprises, the network is critical to all business operations.**

Both organizations must implement network automation to simplify the process of provisioning and controlling network applications and services. Tail-f Multi-vendor solution, NCS, provides network service programmability to achieve important business objectives:

- Faster development and deployment of new network applications and services
- Faster turn-around times for new features
- Real-time dynamic network capacity allocation
- Better quality due to less human errors, less repetitive manual work
- Vendor independence
- Substantial capital and operating savings

### Overview

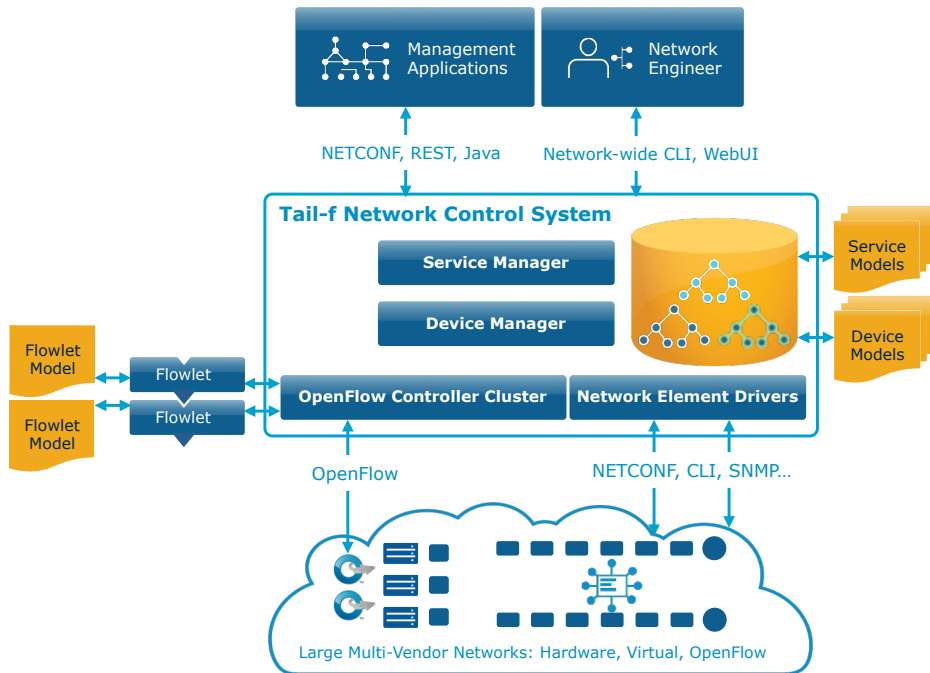
The following are examples of new network applications and services that have been developed in NCS:

- Layer 3 VPNs
- Layer 2 VPNs
- E-LINE and E-LAN
- BGP management (including peering policy management)
- Layer 2 virtual networks
- Access Control Lists (ACLs) and firewall policy rules
- Layer 4-7 Service chaining

**NCS provides a single network-wide interface to all network devices and all network applications and services, as well as a common modeling language and datastore for both services and devices.**

A transaction engine handles transactions from the operations at the service layer to the actual deployment of configuration changes in the network.

Designed to be a generic solution, NCS supports the implementation of network applications and service on a wide variety of networking devices, both traditional hardware-based devices and virtual software appliances. NCS supports OpenFlow-based applications through a built-in OpenFlow controller, Tailflow; alternatively, applications built on top of other OpenFlow controllers can be easily integrated into NCS.



- Open architecture: all interfaces are auto-rendered from declarative YANG data models
- North-bound user interfaces: network-wide CLI and Web UI
- North-bound programmatic Interfaces: JAVA, REST, NETCONF
- South-bound device Interfaces: native protocols such as CLI, NETCONF, SNMP and others

**Service-aware:** NCS provides a specification of how a network service shall be applied to the network infrastructure. This greatly facilitates the mapping of service configuration changes to device configuration commands. The entire service life-cycle is supported including creating, modifying and deleting service instances.

**Model-driven:** Services, device configurations and Openflow applications (flowlets) are specified in declarative YANG data models.

**Fail-safe:** NCS applies all service changes towards the network as an atomic change-set, using distributed transactions. This ensures that the network is always in a consistent state and can automatically recover from failed configuration changes.

**Real-time:** NCS maintains an accurate and synchronized copy of the network configuration state. Orchestration and management systems can be kept in sync with the network in real-time using the publish-subscribe APIs of NCS.



[www.tail-f.com](http://www.tail-f.com)

[info@tail-f.com](mailto:info@tail-f.com)

**Corporate Headquarters**

Korgmakargränd 2  
SE-111 22 Stockholm  
Sweden  
+46 8 21 37 40

**US Headquarters**

5201 Great America Pkwy,  
Suite 320  
Santa Clara, CA 95054  
United States  
Phone: +1 408 466 4241

**Japanese Distributor**

A.I. Corporation  
Iijima Bldg., 2-25-2  
Nishi-Gotanda  
Shinagawa-ku  
Tokyo, 141-0031  
Japan  
+81 3 3493 7981