

**netFLEX®**

*Simplify, Standardize, Personalize*

## **Network Automation Journey Map**

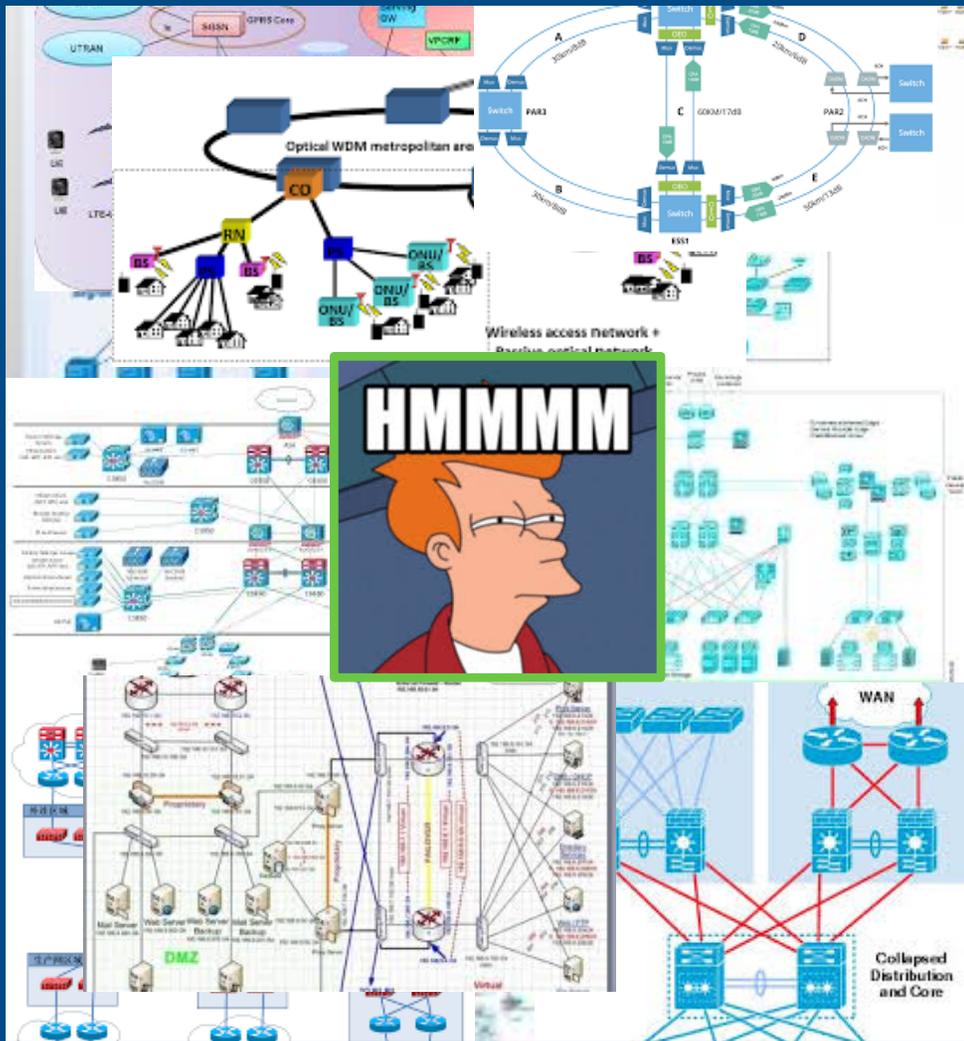
**Are your Customers Self Serve  
for their Network Services?**  
*(Digital Transformation)*

**Are human hands still  
'touching' the network?**  
*(Service Delivery)*

**Are you waiting for your network to break?  
Are you waiting for the customer to call?**  
*(Service Assurance)*

**How well do you  
know your network?**  
*(Inventory)*

**LightRiver** 



How well do you know your network?

# How well do you know your network?

## How would you answer these questions today?

- Is your Network Inventory (*Physical Resources and Logical Services*) maintained by human-data-entry? In spreadsheets? In traditional OSS/Inventory platforms?
- When service delivery gets busy with turning up new service, do disconnects get worked (and reflected as such in your OSS/spreadsheets/similar, or the network itself)? If not, how do you figure this out after?
- Do your network and capacity planners have the most accurate information to know when to add/grow network, leverage that which is already in place, & reuse/move inventory when able?
- If you have begun a network automation journey – where is your inventory/config data coming from? What is your automation fall-out if the data is wrong?
- Are you still managing your Elite/SLA-managed customers with spreadsheets and PDF's?
- When the 'big pipe breaks', how good is your understanding of correlation/topology to know the true customer impact or find the issue at all?
- If a Network Element fails, and can't be restored, do you have timely back-ups to restore?
- What are you using to reconcile static OSS/similar data repositories to get to 'the truth?' (day one, and then after)
- If you have figured out all of the above – how is this *network-as-the-truth* data shared with other consuming applications?



**Are you waiting  
for your network  
to break?**

**Are you waiting  
for your customer  
to call?**

# Are you waiting for your network to break?

# Are you waiting for your customer to call?

*How would you answer these questions today?*

## Reactive?

- Are your NOC technicians primarily 'eyes on glass' – watching alarms, logs, similar?
- Do you assign technicians to SLA-managed customer networks as dedicated 'watchers'?
- Is the vast amount of network health and performance data collected not usable because you struggle to make it 'actionable' to a technician or the customer?
- Do you have to wait for customer reports (calls, other), before starting corrective action?
- Do you have to watch (or fire up) multiple screens to see trouble in your network?
- Do you have to open manual trouble tickets to launch corrective action?
- When you finally respond, do many of your 'troubles' turn out to be 'no trouble found'?

## Proactive?

- Can you proactively ID issues so you can correct the issue *before* they become customer troubles?
- Can you custom-define (personalize) your proactive analytics to decide what you act on, and what the outcome should be (i.e. to ticketing, dispatch, auto-remediate, other).

## Autonomous?

- Can your systems launch trouble tickets without waiting for humans to see trouble?
- Is your network self-repairing, so troubles are cleared as they occur?
- Is your network and service health data programmatically shared with other consuming applications?
- When your network is running smoothly, can your customers self-serve to confirm?



# Are human hands still 'touching' the Network?

# Are human hands still 'touching' the Network?

## *How would you answer these questions today?*

- Are your technicians/engineers still logging directly into NE's for provisioning and change? Likewise, into disparate shelf GUI's or EMS' ?
- For either, are they using standard configs, templates, or engineering designs to turn-up or change service?
- Are the engineering designs created by human guesswork, coming from disparate static-inventory sources or worse?
- Is the design implementable? That is... is capacity actually available at time of provisioning and activation on the network?
- If the provisioning ends up different from design, are records updated?
- During repair and maintenance situations, are technicians manually changing the network to restore service? If so, are records updated?
- If you update records, do you do it manually, typing into databases or worse?



# Are your Customers Self Serve for their Network Services?

*(Digital Transformation)*

# Are your Customers Self Serve for their Network Services?

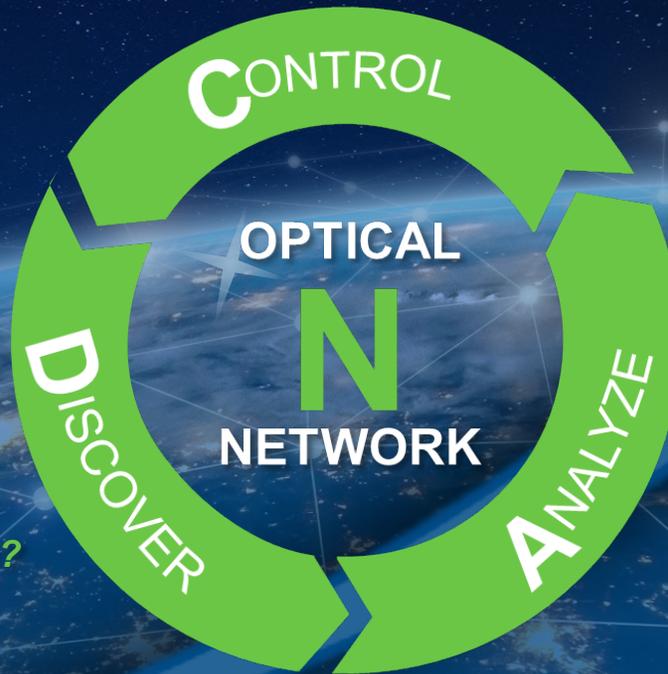
*(Digital Transformation)*

## *How would you answer these questions today?*

- Can a customer see availability of your (their) network, services, and capacity available?
- Do your customers have access to their Managed Network and see end-2-end network layout, performance, alarming, broader?
- Can they custom-define (personalize) their analytics, for thresholding and actioning (as they would define upfront, and likewise after)?
- Can you demonstrate the quality of your network by allowing your customers access to network health data (beyond base telemetry)?
- Do your customers have the ability to self-test/triage, across their purchased services?
- Can they provision (activate) service on-demand? Likewise, as scheduled or as tied to predefined policy (analytics-driven)?
- Can you offer multiple private virtual optical networks to your customers and trading partners across your core network (Optical VPN, or similar)? Is this also self-serve enabled (through UI, or API)?
- Do your Sales Professionals have similar access to the above, to act on the customer's behalf?

**Solution to the  
punchlist questions...**

*Are human hands still  
'touching' the network?  
(Provisioning Automation)*



*How well do you  
know your network?  
(Inventory)*

*Are you waiting for your  
network to break?*

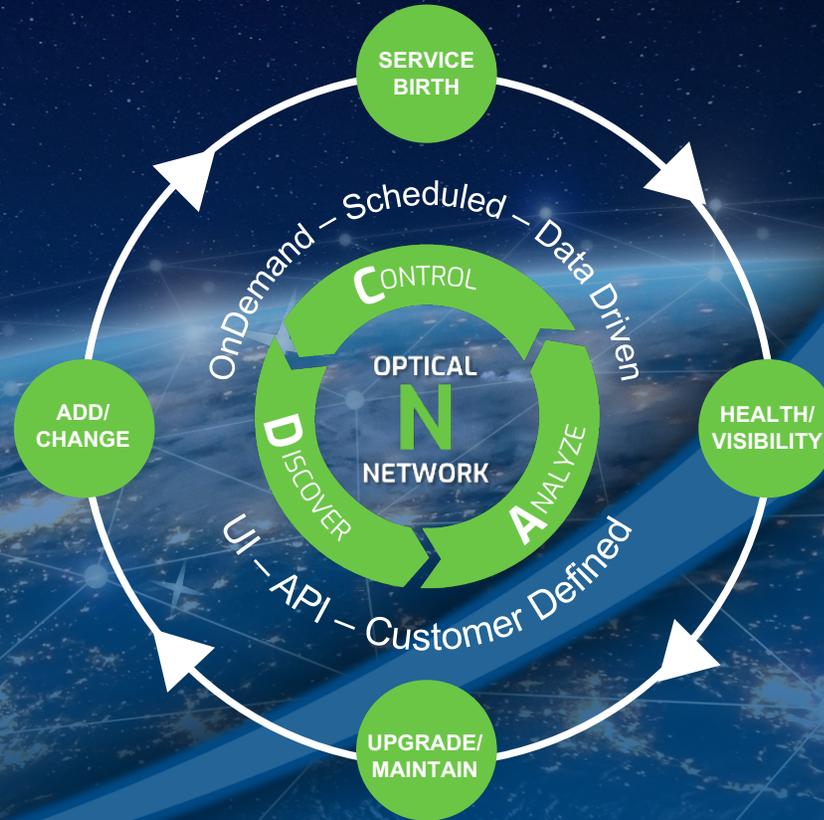
*Are you waiting for the  
customer to call?*

*(Assurance Automation)*

**netFLEX®**

*Simplify, Standardize, Personalize*

# Are your Customers Self Serve for their Network Services? (Digital Transformation)

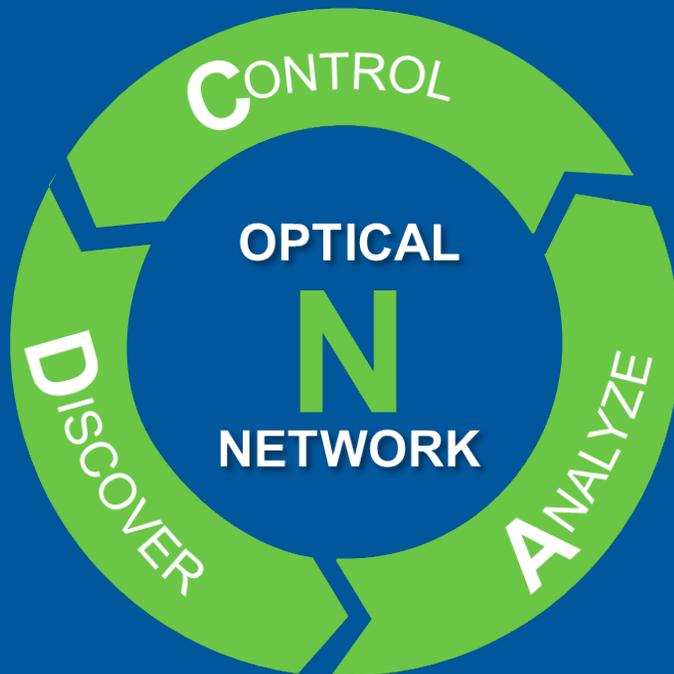


## Life-Cycle Service Automation

(UI, API, Custom Defined --- OnDemand, Scheduled, DataDriven)

**THE LIGHTRIVER RESPONSE:**

# netFLEX PROGRAMMABLE NETWORKING



## netFLEX Reference Framework

- **SDN:** Optical Domain Control for end-2-end Network Abstraction
- **Only Multi-Vendor Solution:** that includes Discovered Inventory, Actionable Analytics, & Control Automation
- **Network FLEXibility:** across every functional area
- **Programmability:** of Visibility & Control, as the Operator & Customer would Define
- **Digital Experience:** enablement of UI, API, and Program-and-Go for Every Functional Area

For each functional area, and in total – Industry Best!

**netFLEX®**  
*networkFLEXIBILITY*

## THE ONLY SDN OPTICAL DOMAIN CONTROL SOLUTION THAT REFLECTS...

**SupplierFLEXibility**... dozens of suppliers

**Network-elementFLEXibility**...hundreds of network elements & solutions supported

**TechnologyFLEXibility**... WDM, DCI, ADM, DCS, OTN, CE, Packet Optical, whitebox/open source

**GenerationFLEXibility**...decades of supported solutions

**DomainFLEXibility**...all domains (supplier, solution, tech....) supported for D-A-C capabilities

**DXFLEXibility**...Digital Experience – full Analytics, Inventory, & Control UI/UX and API extensibility

**SDN Abstraction... Make it all look the same!**



### AWARD WINNING LIGHTRIVER FACTORY BUILT NETWORK

No better way to truly 'know' configuration and engineering spec's for automation – then to engineer-and-install thousands of nodes annually



### LIGHTRIVER LABS

Multiple Facilities dedicated to engineering, test, and certification of field deployments & automation.

No other network FLEXibility test and validation approach of its kind.



### LIFECYCLE ENGINEERING

Commitment to all of the above for the life of the individual technology platform - the hardest part !

**netFLEX®**

©2019 LightRiver Technologies

**LightRiver** 