ic23

infocomm

EDUCATION /
JUNE 10 - 16

EXHIBITS /
JUNE 14 - 16

**ORLANDO,
FLORIDA**

Applied Monitoring

For AV Systems (and more…)

Instructors

Fred Loucks
CTO
Level 3 Audiovisual

Jeremy Elsesser
CEO
Level 3 Audiovisual

Learning Objectives

- ✓ Identify the business cases, stakeholders, and practices involved in monitoring
- ✓ Explore the factors involved in properly designing, specifying, and implementing a monitoring solution for AV systems
- ✓ Learn how to collect good data from your AV systems and how to design AV systems that simplify this process
- ✓ Using free and open-source software, develop functional monitoring "drivers" for a working reference meeting room

Goals for this course

We’re here to answer two fundamental questions:

AM
- How are you going to get the data you need?

PM
- How do you decide what data you need in the first place?

By the end of this class…

You will know how to provide a proper uptime SLA for their AV systems for cheap or FREE using opensource software.

Housekeeping

- # on sticky note
- Sharpie: name on card
- Restrooms
- Breaks ~90 minutes
- Lunch at 12
- WiFi / Internet
- Calls / Laptops

- Laptop / Web browser
  - Node-RED (concepts)
  - Slido (polls, Q&A, etc)
- Raise hand with questions or drop them in the Slido Q&A if you want us to wait until later

slido

Please take a few minutes to share some data with the class. Review together after!

ⓘ Start presenting to display the poll results on this slide.

Thank you for sharing!

Breakout - Meet Your Pod

For each member of your pod, share the following:

1. What is your name & job role?
2. Why are you in this class?
3. What do you hope to get out of it?
4. What is something we don't know about you?

Designate a spokesperson for your pod who will share a summary of the above information with the whole group...so make sure they take notes!

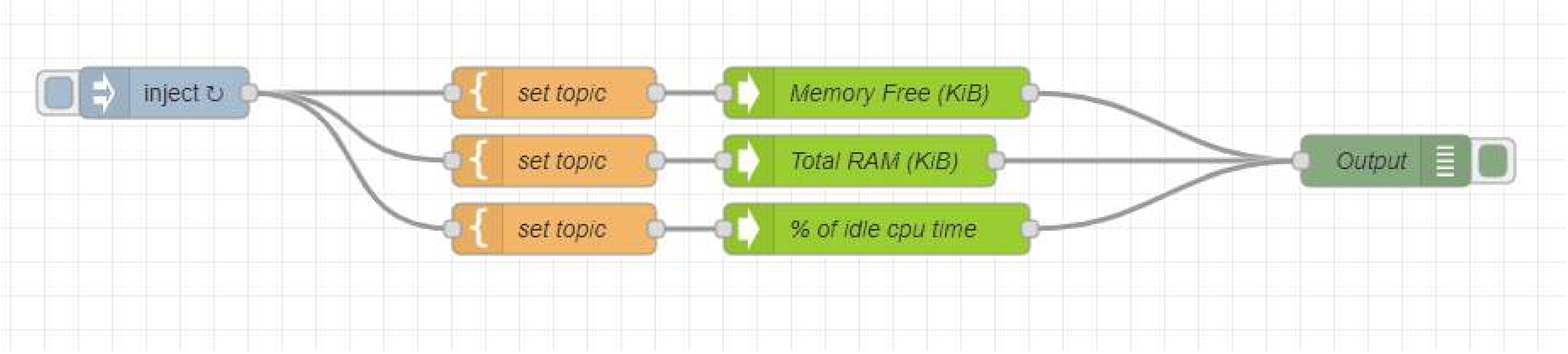
Sharing - Meet Your Pod

One representative from each group will stand and share:

- The first names of each member of the pod
- If there were common hopes / expectations for the course between the pod members
- The most interesting thing you didn't know about someone

Tools: Node-RED

- Free and Opensource Software (FOSS)
- Low-code programming editor in your web browser
- Millions of users across dozens of industries
- https://nodered.org



Intro to Monitoring

What is monitoring?

What is Monitoring?

“IT monitoring is the name for the products and processes used to determine if an organization’s information technology (IT) equipment and services are working properly and to detect and help resolve problems.”

–Splunk*

- “Is it working?”
- “Is it working frequently enough?”
- “Why isn't it working?”

***https://www.splunk.com/en_us/data-insider/what-is-it-monitoring.html**

Monitoring: going deeper

Availability Monitoring

- Keep track of basic metrics like uptime and performance.

Business Activity Monitoring

- Using the technology to keep track of how the business is performing.

Why do we monitor?

Why do you want to monitor?

- Individually, write out a few sticky notes for the reasons you want to monitor.
- We’ll keep these in front of you the whole class so you can make sure you get your questions answered.

Common reasons to mon.

- Know when something is broken
- Measure service availability
- Asset management
- System utilization
- Making data driven decisions
- …how about you?

The Ultimate Why

To better serve the business

Success =

- Knowing the business need
- Meeting the business need
- Proving the business need has been met

Businesses use KPIs to measure business performance…so we should do the same.

KPIs

Key Performance Indicators

Monitored regularly to ensure that the business performance targets are achieved and maintained.

What KPIs do we use in AV to measure our performance?

Example KPIs

- Room Uptime SLA
- Room Utilization Ratio
- Room Occupancy Ratio
- Default Password Ratio
- Customer Satisfaction Score
- Ghost Meeting Minutes
- Warranty Exposure
- Remote Accessibility Ratio
- …others?

Why isn’t everyone monitoring?

Monitoring AV is *hard*

- AV/IT segmentation
- Walled gardens (“ecosystems”)
- Lack of vendor agnostic single pane tools
- No protocol standardization
- IT & software skills gap
- Transactional business models
- Little commercial pressure
- No urgency

Difficult but not impossible!

Hacking through the jungle

First concept: Observability

Observability

"...observability is the extent to which you can understand the internal state or condition of a complex system based only on knowledge of its external outputs.

The more observable a system, the more quickly and accurately you can navigate from an identified performance problem to its root cause, without additional testing or coding."

-IBM*

*https://www.ibm.com/topics/observability

Observability in AV

To create observable *AV systems* you must be:

Educated

Choosy

Creative

Persistent

Two general requirements

Data must be available to be collected...

(API, Network)

…and we must have the means to extract, transform, and load it somewhere.

(Network, Skills, Tools)

APIs

Application Programming Interface

“a set of defined rules that enable different applications to communicate with each other”

In AV: the interface we use to communicate with and observe devices and services

Judging devices by their data

The device or system is "bad" for monitoring if...

...there is no API...

...the data you need isn't available on the API...

...the data is available but it's difficult to get.

In general, if the device is hard to "observe"!

Laying the foundation

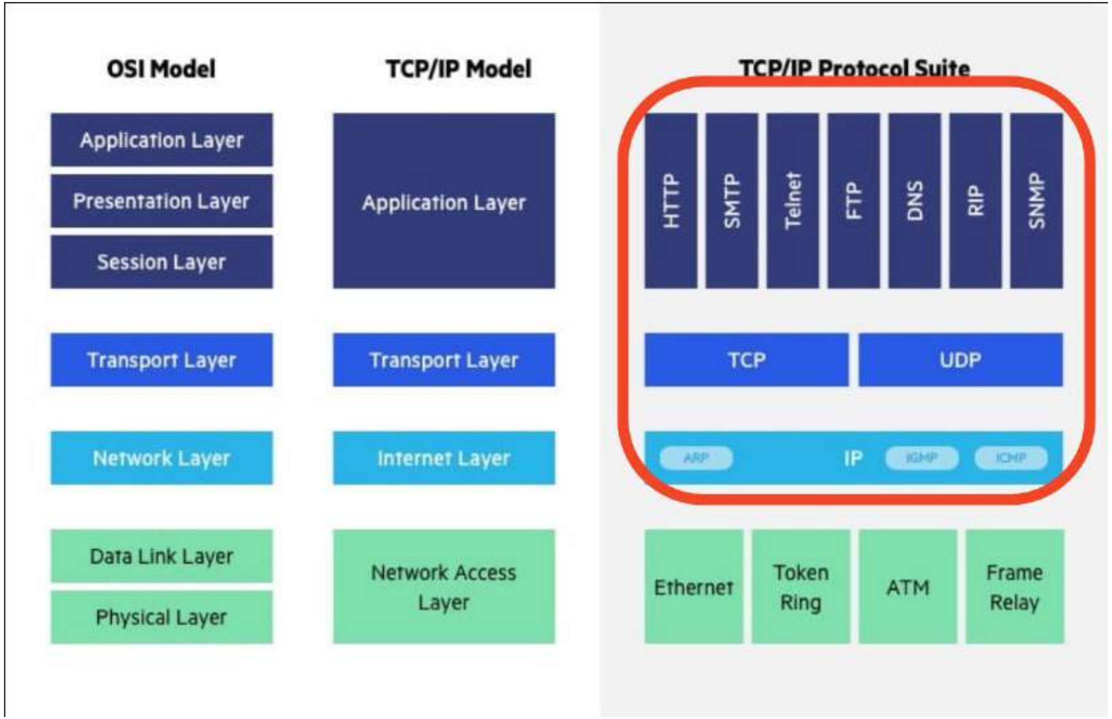
Fundamental #1

No network? No monitoring.

All monitoring we'll discuss will be *network* monitoring.

Any non-networked collection will be proxied through network-connected devices.

Fundamental #2



OSI Model	TCP/IP Model	TCP/IP Protocol Suite
Application Layer Presentation Layer Session Layer	Application Layer	HTTP SMTP Telnet FTP DNS RIP SNMP
Transport Layer	Transport Layer	TCP UDP
Network Layer	Internet Layer	ARP IP IGMP ICMP
Data Link Layer Physical Layer	Network Access Layer	Ethernet Token Ring ATM Frame Relay

Fundamental #3

Direction

Inbound

Device <--- Collector

Outbound

Device ---> Collector

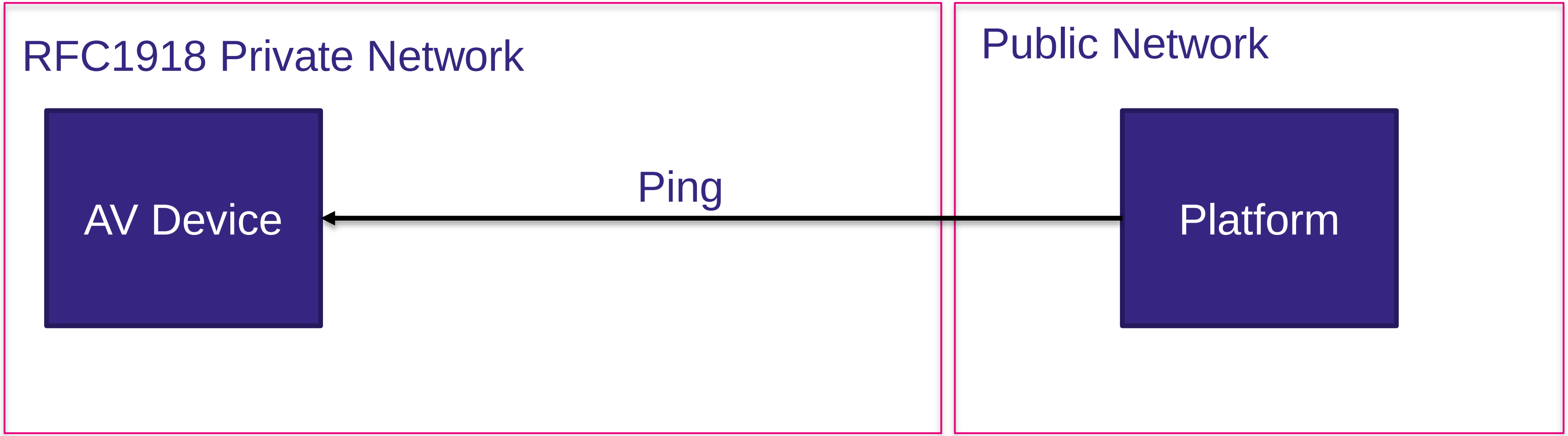
Fundamental #4

State

Stateless
Send & Forget

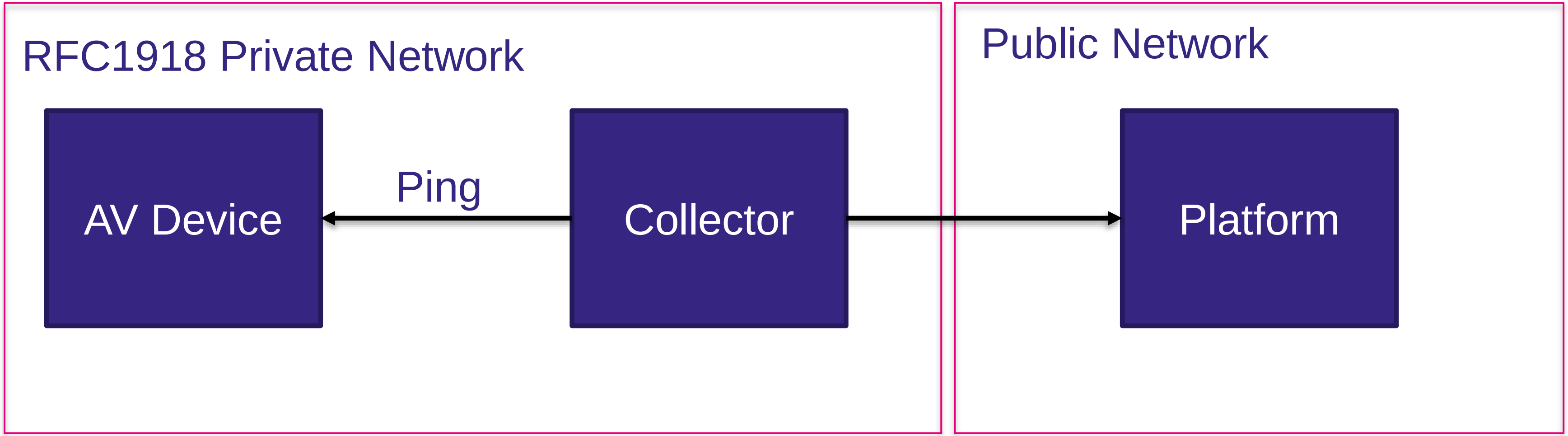
Stateful
Establish & Maintain

Spot the issue

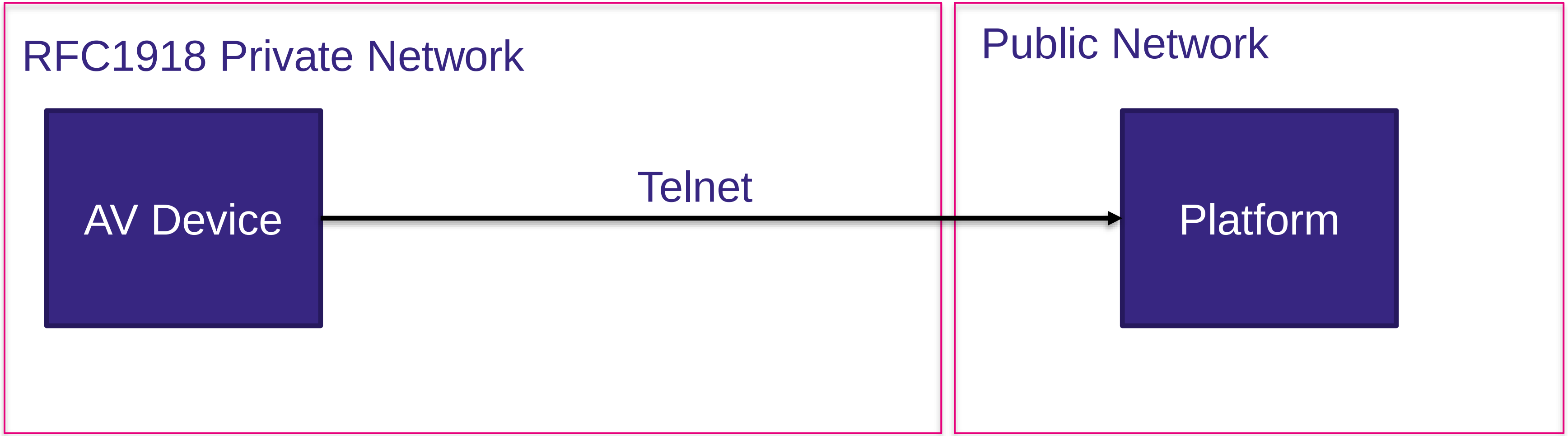


Fundamental #5

Inbound connections must have a proxy / bastion host inside the private network

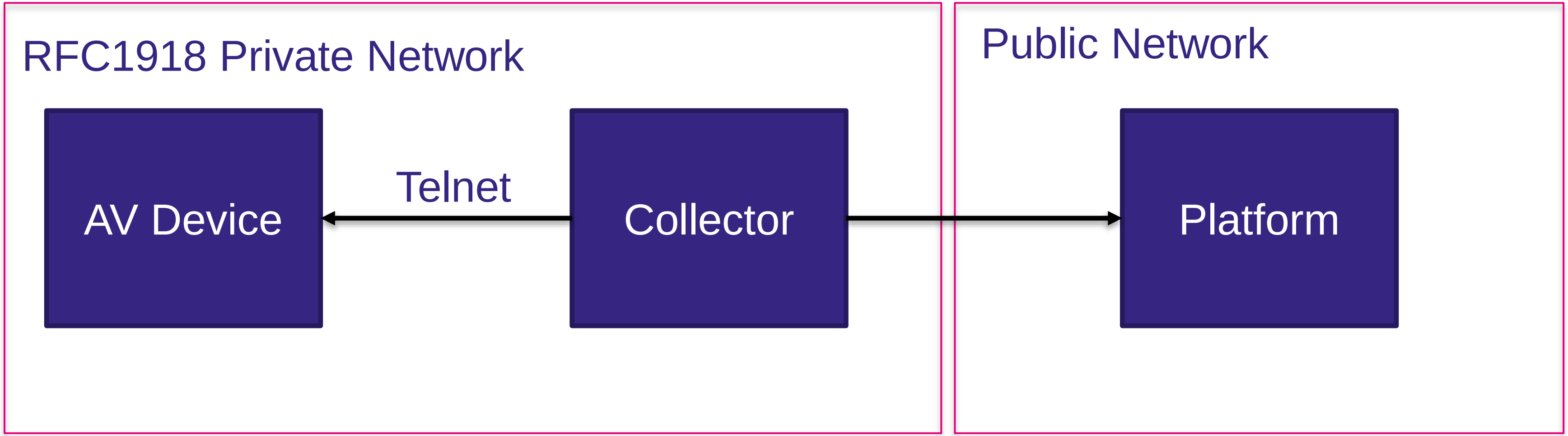


Spot the issue



Fundamental #6

Insecure protocols must not be used over the public internet or other untrusted networks.



Fundamental #7

JavaScript Object Notation - JSON

JSON fundamental data types:

- String – “some text”
- Number – 0123 vs “0123”
- Boolean – true or false (not “true”)
- Object – {“key”:”value”, “num”: 123, “foo”: true}
- Array – [123, 346, 789, “foo”, true]
- Null - <it means nothing>

Concept 2 - SPOG

How do you define SPOG?

Single Pane of Glass (SPOG)

My definition:

1 interface
in which you can view 99% of your systems,
99% of your devices, and
99% of the data you care about.

16x9

Concept 3 - ETL

ETL

Extract, Transform, Load

“A data integration process that combines data from multiple data sources into a single, consistent data store that is loaded into a data warehouse or other target system” –IBM*

- Extract data from legacy systems
- Transform data to improve quality & consistency
- Load data into target database or system

Recap

Concepts:

- Observability
- SPOG
- ETL

Fundamentals:

1. Network monitoring
2. Know your TCP/IP stack
3. Direction In/Out
4. Stateful/Stateless
5. Private networks make things tricky
6. No insecure protocols over the internet
7. JSON data types

ic23

Take a break!

infocomm

Extraction (of data from…)

Extract(TL)

- Raw data is copied or exported from source locations to a staging area.
- We can extract data from a variety of data sources
- “Anything with an API, intentional or otherwise”
- Sources could include but are not limited to:
  - AV devices
  - Cloud platforms
  - Control systems
  - Files
  - Literally anything with an API

https://www.ibm.com/topics/etl

What are some common monitoring protocols?

Protocols & methods used

- Simple checks
  - Ping
  - UDP/TCP port checks
- SNMP check
- TCP Consoles
  - Telnet & Telnet-like
  - SSH
- Push only
  - TCP/UDP push
  - SNMP trap
  - Syslog
- HTTP
  - SOAP
  - REST / RPC
  - GraphQL
  - Web scraping / driving
- IoT
  - MQTT
  - AMQP

Ping / ICMP

Internet control message protocol, layer 3, port-less.

The universal “are you on the network” check.

Why

- Reachability
- Round trip time (latency
- Packet loss

Port Checks (UDP/TCP)

Is anyone listening on this address & port?

Why

- Service availability
- A port that is closed that should be open means something is badly broken
- Simple but powerful

SNMP (the "s" is a lie)

Simple Network Management Protocol

OG – first release in 1987!

Why

- Simple if you know what you're looking for
- Robust
- Universal…but not in AV

OID - object identifier

MIB – management information base file

Consoles / TCP sockets

Ubiquitous in AV…every control system interface ever

Why

- Most common APIs in AV
- SSH is wonderful!

Why not

- Most are fundamentally insecure…telnet is **dead**
- Outside of SSH can be very brittle

HTTP/s

Hypertext transfer protocol (secure)

The language of the internet and pretty much everything.
Transport for SOAP, REST, RPC, GraphQL and web browsing

Why

- Flexible
- Stateless
- Main way custom APIs are served today
- Get anything they make available here

HTTP/s – RESTful / RPC

RESTful: Representation State Transfer (ie databases)

- CRUD: Create Read Update Delete

RPC: Remote procedure call

- If you’re calling reboot, it’s an RPC API

Generally in AV devices, we’re dealing with RPC using JSON

HTTP/s – GraphQL

GraphQL (query langage) – modern and more flexible

Still new, common on cloud data platforms…kudos to Zoom and Poly Lens

Why

- Shape the payload before you get it
- No more merging multiple responses after you get them
- Can get access to private APIs if you’re a sleuth

HTTP/s – Web Driving

Sometimes there’s just no other way…

Why

- Can monitor anything in the web interface!

Why not

- Time consuming
- Brittle…can change without notice

HTTP/s – MQTT

The future! Dominating in IoT and industrial
Concept: Pub Sub! (I’ll sketch it out)

Why

- Realtime
- Private network friendly ie secure!
- Lightweight
- Topics & JSON = flexible

Why not

- Little adoption in AV…yet

Extraction – wrap up

If its available, you can get it with some patience.

Transforming it into something useful is another thing entirely

That’s where we’ll pickup in tomorrow morning.

Reflection & Questions

Take a break!

Congrats, you can monitor just about anything!

Just because you can…

…doesn’t mean you should.

Climbing the tree

So you've want to start monitoring…

The first step is…

Stakeholder Analysis

Section Goals

Questions:

- Who are your stakeholders?
- Who are your *important* stakeholders?
- What do they care about?

Objectives:

- Capture & *prioritize* desired outcomes
- Set your project up for success

Stakeholder Analysis Tasks

1. Identify all potential stakeholders
2. Sort / prioritize stakeholders by power v. interest
3. Analyze 1st quadrant stakeholders
   - Motivations
   - Priorities
   - Positive/negative outlook
   - Outcomes
   - KPIs

Identify stakeholders

Example Personas

Technology / Service Manager

AV Engineer

Embedded Programmer

Support

Project Manager

IT Partners

Sponsor

Decision Maker

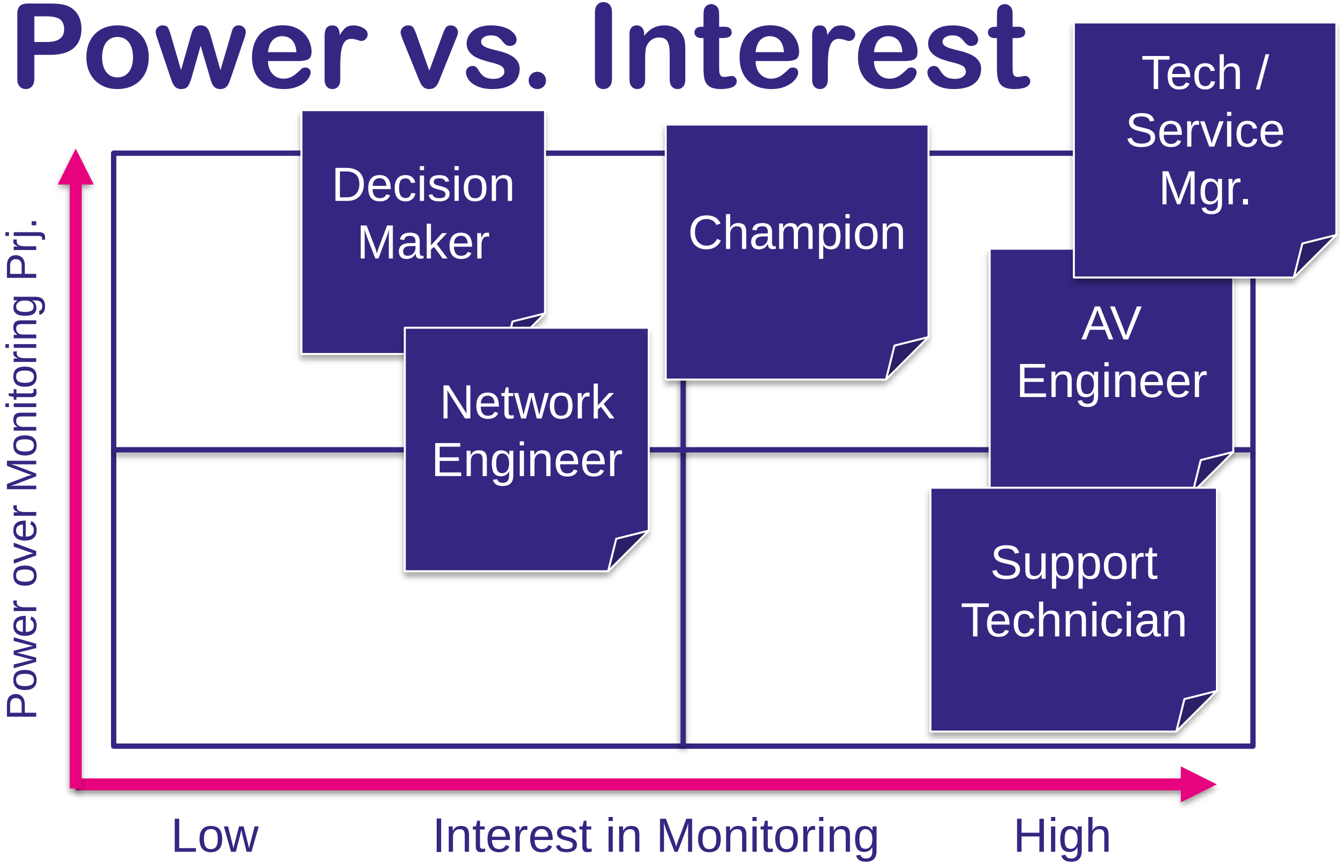
Identify Stakeholders

- Individually, take 5 minutes to ideate a list of potential stakeholder personas who would have either power over or interest in an AV monitoring project.
- Create a sticky note for each.

Consolidate Stakeholders

- As a pod, put your stakeholders on a flip chart page.
- Keep/kill/combine until there are no duplicate personas

Power vs. Interest



Power v. Interest

- Draw your own XY axis with power on the vertical (y) axis, and interest on the horizontal (x) axis
- Discuss and categorize each persona into the appropriate quadrant or somewhere in between
- Highest priority quadrant is top right

Share: Top Right Quadrant

- Who were the personas in your top right quadrant?
- Did everyone agree on the placement of the personas?
- Did you come up with any unexpected stakeholders?

Stakeholder Profile

Stakeholder	Role	Technical or Business	Internal or External	Quadrant
Frank	AV Technology Manager	Business	Internal	1 – Players

Motivated by	Priorities	View (+/-)	Features Desired	Outcomes / Success
1) Meeting room performance 2) Device / room data insight 3) Faster issue resolution 4) Maximizing existing staff	• Able to get any available datapoint from every device in the environment • Able to know space health based on device / platform health • Ability to measure uptime • Ability to see full history of data / settings for devices & spaces • Not adding new staff	Likely to have a positive view as they are sponsoring the project.	1. Ability to know at any moment the **health** of all systems 2. Ability to auto-generate tickets 3. Ability to auto-remediate issues 4. Ability to gain visibility to EVERY device in the environment 5. Deep troubleshooting capabilities	1. Users trust their meeting rooms and the meeting room service

Breakout: Stakeholder Profile

For each stakeholder in your top right quadrant answer the following questions:

- What is the stakeholder's role / title?
- Is this a technical or business stakeholder?
- Is this an "internal" or "external" stakeholders?
- Will they have a positive or negative view on the proj?
- Why is the stakeholder interested in monitoring AV?
- What power does the stakeholder have over monitoring?
- What does success look like to this stakeholder?

Share: Stakeholder Profile

- Share your favorite stakeholder profile!
  - What is the stakeholder’s role / title?
  - Is this a technical or business stakeholder?
  - Is this an “internal” or “external” stakeholders?
  - Will they have a positive or negative view on the proj?
  - Why is the stakeholder interested in monitoring AV?
  - What power does the stakeholder have over monitoring?
  - What does success look like to this stakeholder?
- Answer any questions the group as for you

Reflection & Questions

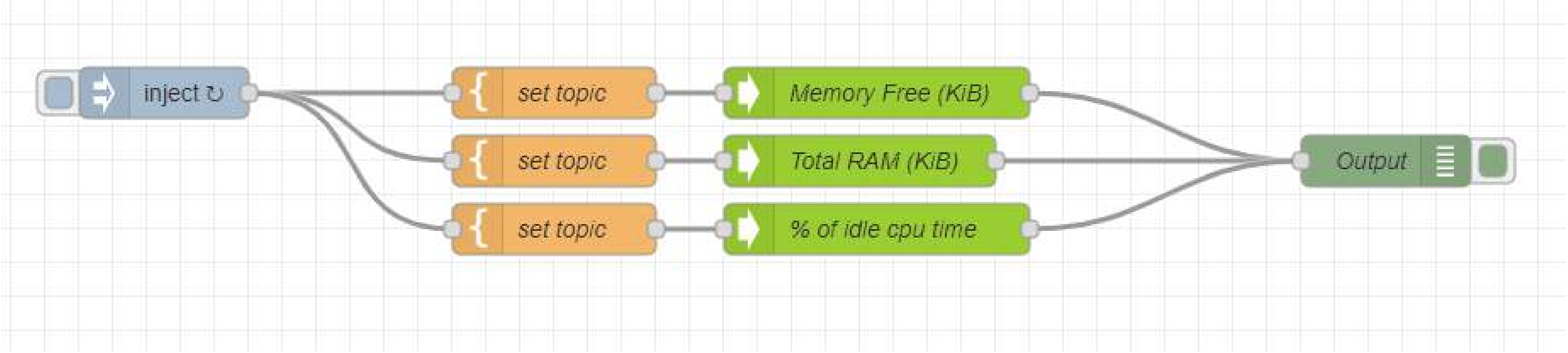
Recap

1. Monitoring fundamentals
2. ETL
3. Extraction & protocols
4. Stakeholder analysis
    1. List
    2. Prioritize
    3. Profile

Wrap-up and Prep for Day 2

Node-RED

- Free and Opensource Software (FOSS)
- Low-code programming editor in your web browser
- Millions of users across dozens of industries
- https://nodered.org



Node-RED Login

- Each of you has a dedicated instance for this class

- Connect to the class Wi-Fi network
  - SSID: Applied Monitoring / PW: 0bserveAV!
- Log into your instance
  - URL: https://studentx.class.l3av.io
  - Username: studentx / PW: 0bserveAV!

Node-RED Live Walkthrough

- Nodered.org
- The editor
  - Main workspace / flow tabs
  - Palette
  - Sidebar
  - Header
- Flows.nodered.org
- Node.js / npm

Node-RED bonus activity

- For those interested in Node-RED:
  - Extra time with Fred to go through Node-RED and how it could be used in AV.
  - Building the flows shared today.
  - Interactive Q&A.

End of Day 1!

Dawn of

The Second Day

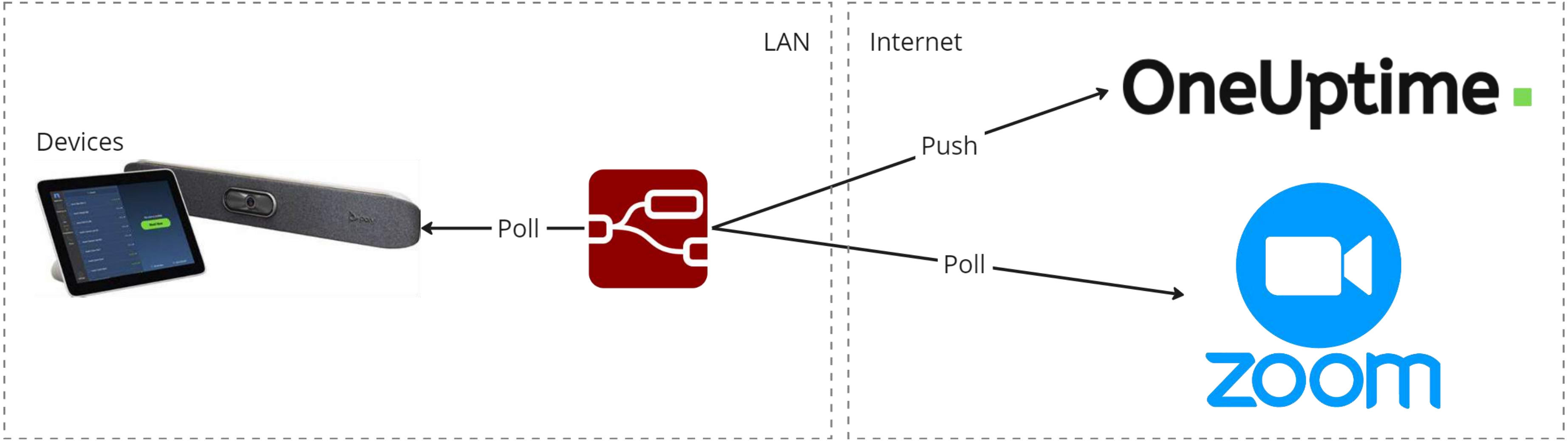
-48 Hours Remain-

Day 2

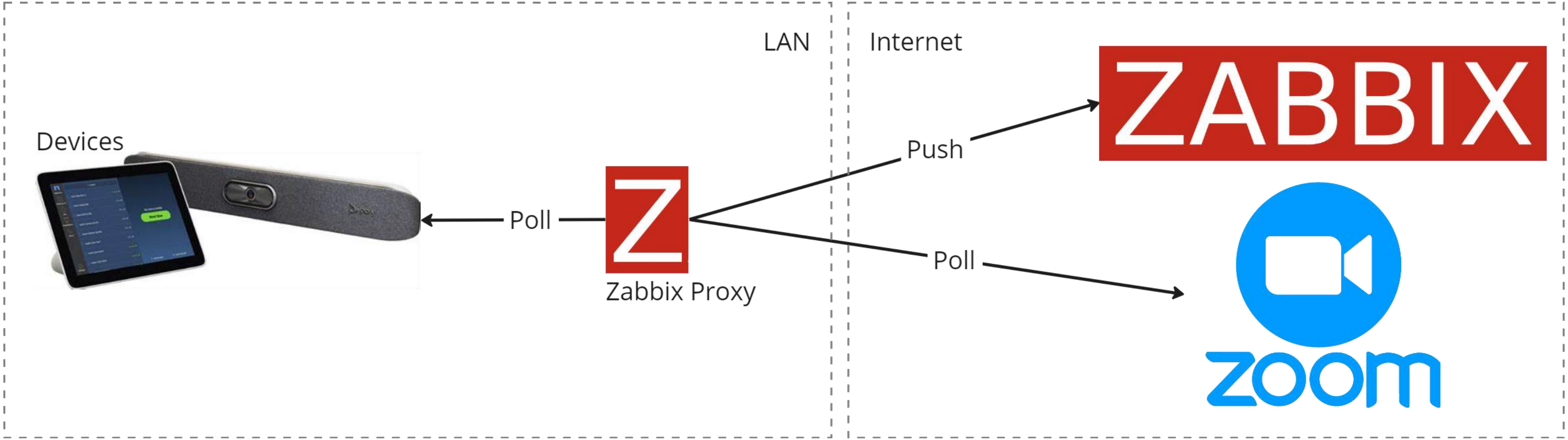
THANK YOU FOR THE FEEDBACK!!!!!!!!

A Note On Monitoring Tools

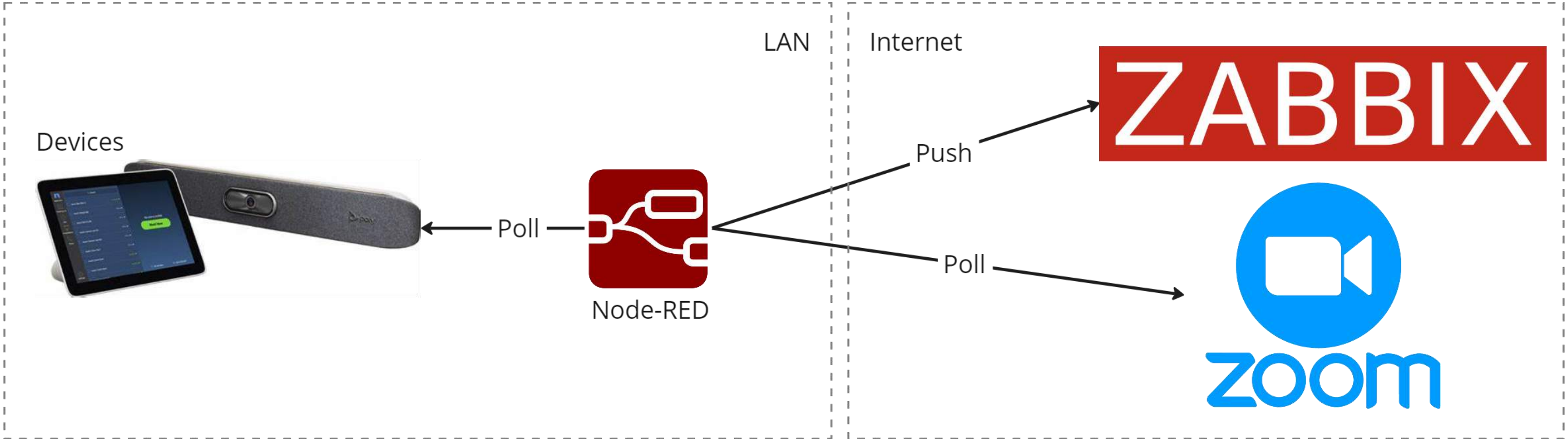
Demo Monitoring Topology



Alt Monitoring Topology 1



Alt Monitoring Topology 2


LAN

Internet

Devices

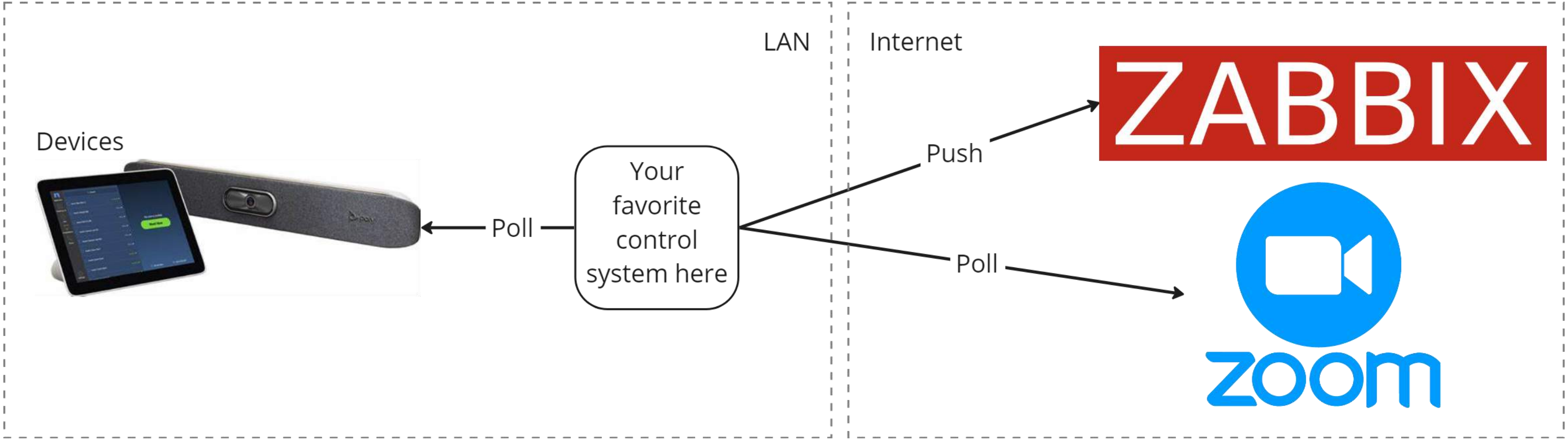
Poll

Node-RED

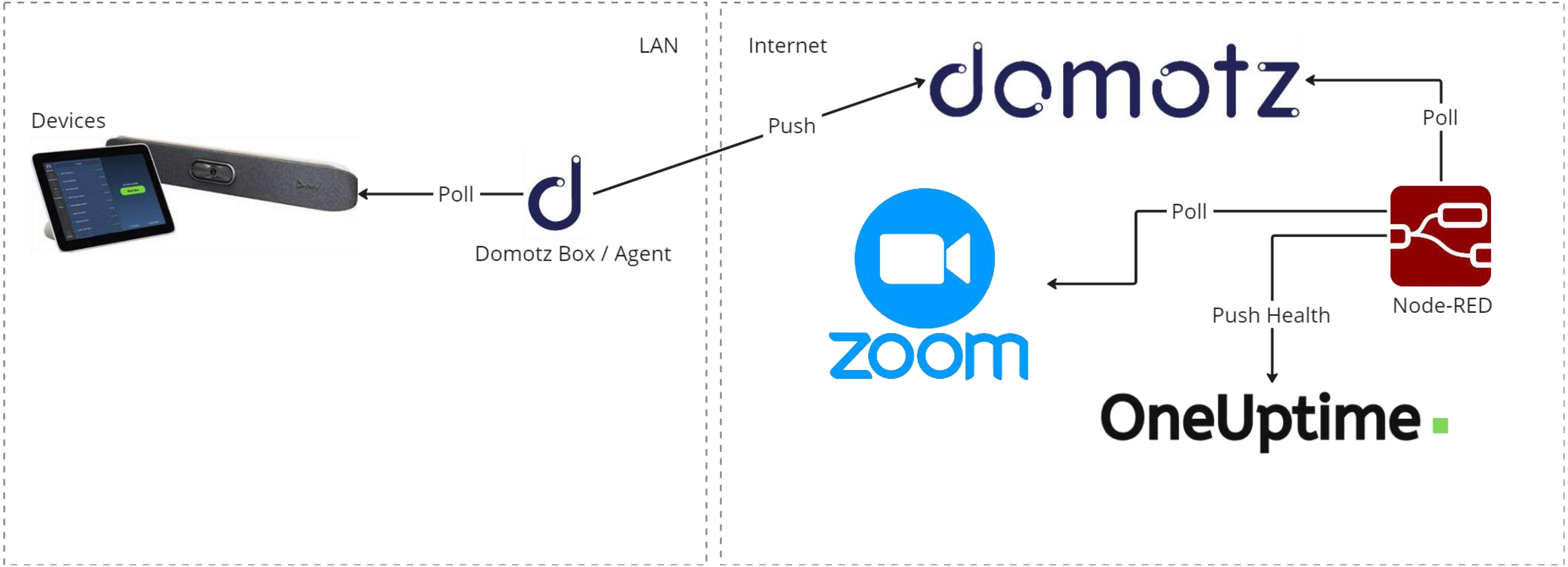
Push

Poll

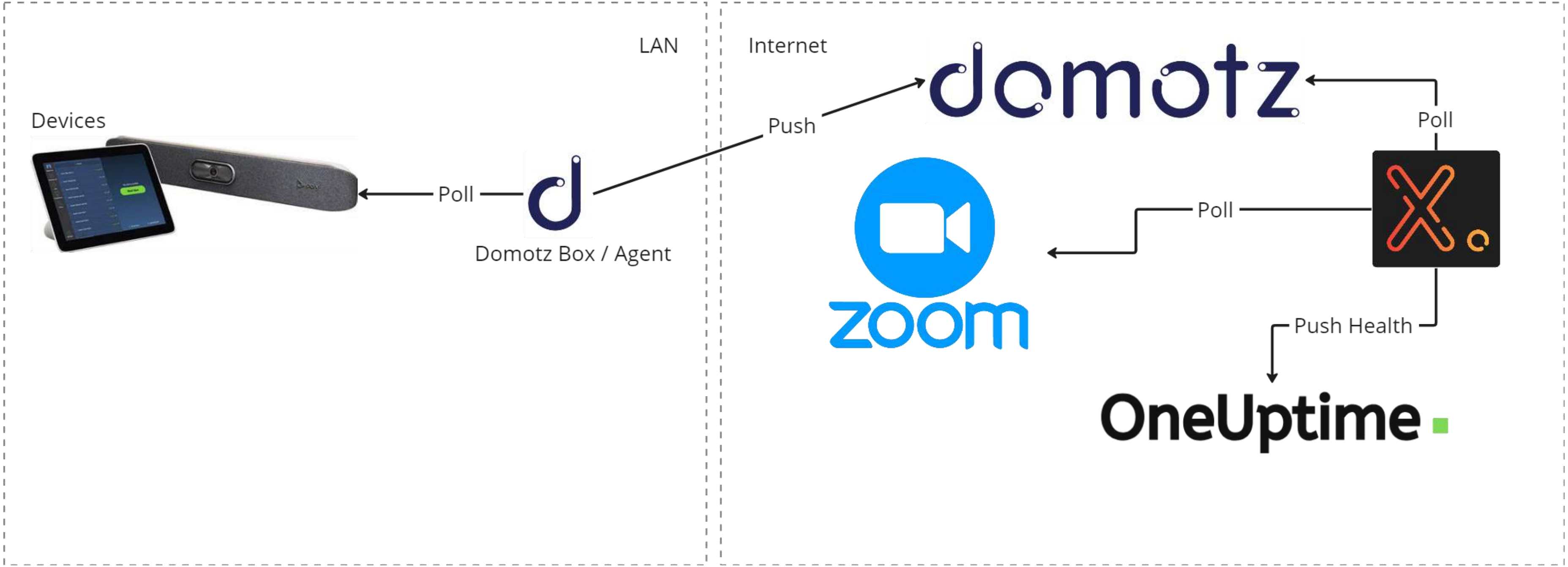
Alt Monitoring Topology 3



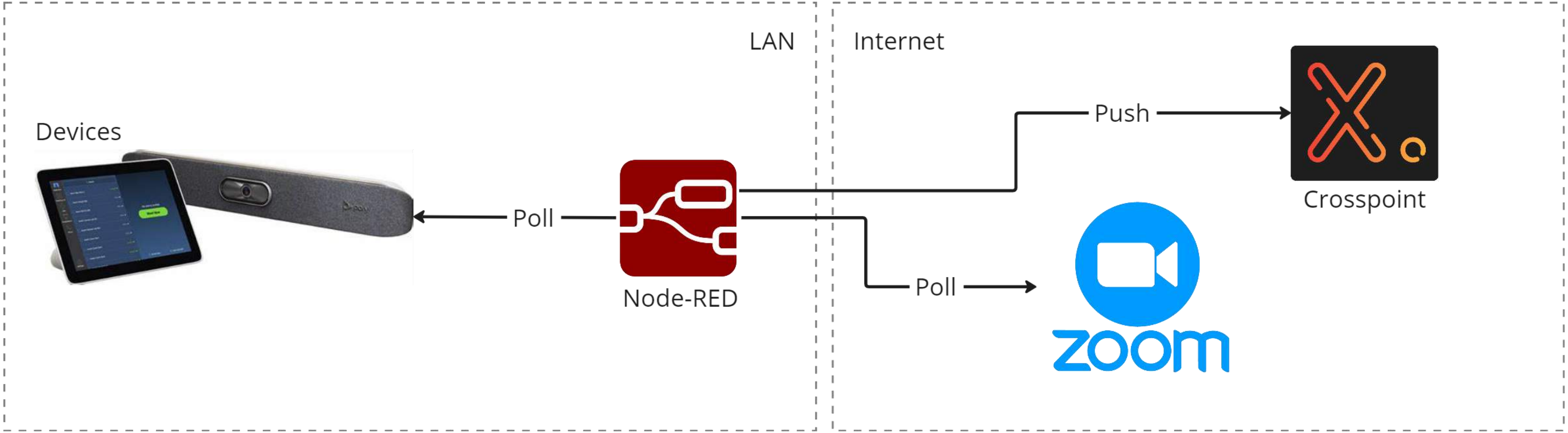
Alt Monitoring Topology 4



Alt Monitoring Topology 5

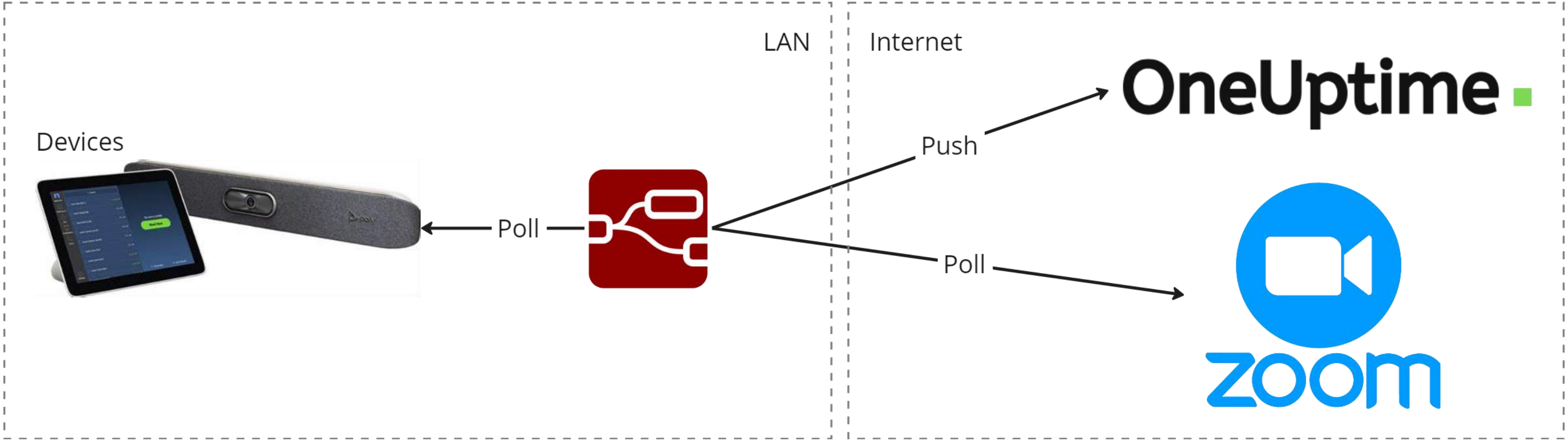


Alt Monitoring Topology 6

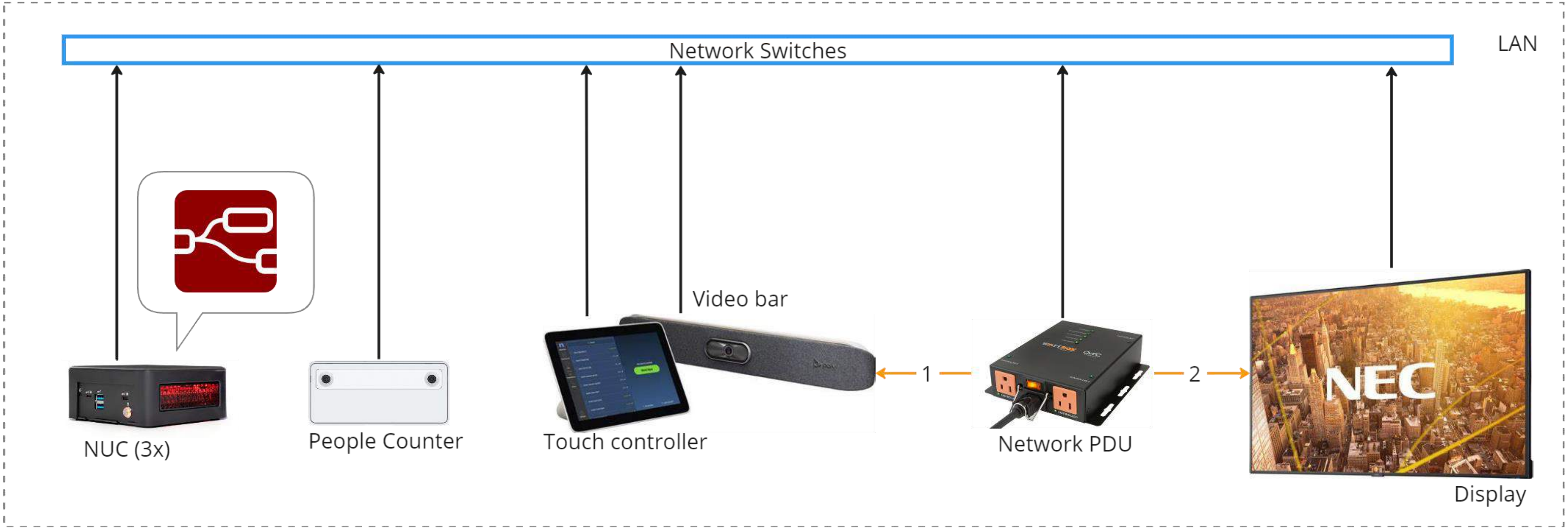


Monitoring End Result Demo

Demo Monitoring Topology



Demo System


Network Switches
LAN
Video bar
1
2
NUC (3x)
People Counter
Touch controller
Network PDU
Display

Show in Node-RED & OneUptime

Data Transformation

Sometimes our data isn’t exactly how we need it…

…so, we need tools to change our data into whatever format is required.

That way we can USE the data we’ve worked so hard to collect.

(E)Transform(L)

- In the staging area, raw data undergoes data processing.
- Here data is transformed and consolidated for
- Includes the following tasks:
  - Converting unstructured data to structured formats
  - Performing calculations, translations, or summarizations based on the raw data
  - Conducting audits to ensure data quality
  - Formatting the data to match the schema of the target data warehouse or system

https://www.ibm.com/topics/etl

Data formats

Unstructured

Raw string (text) data with no strict parsing method

“Connection to 127.0.0.1 22 port [tcp/ssh] succeeded!”

Structured

Serialized formats with a consistent pattern, allowing it to be parsed reliably

{“key”:”value”, “foo”:”bar”} or “value, value2, value3\r”

Data formats cont

Unstructured

Text streams, log files, emails, most console responses (ping, nc, snmpwalk, ssh…)

Structured

Tabular, CSV, JSON, XML

Unstructured data example

Regex

- Regex, or regular expressions, is a sequence of characters that forms a search pattern, primarily used in pattern matching within strings.
- It helps sift through data by matching specific patterns, allowing focused searches within large data sets.
- Regex validates data by ensuring it follows the expected format, improving data quality and consistency.
- It's used in search-and-replace operations within texts, allowing automated corrections or formatting changes.

Regex Syntax & Examples

Practice Question 1

Go to: regexr.com/7f8f6

Build an expression that selects the status of the connection (succeeded or refused).

1. Use the “text” tab to build the expression
2. Once you succeed on the “text” tab, check the “tests” tab and ensure both tests pass

Raise your hand when you’re done

Practice Question 2

Go to: regexr.com/7f8fr

Build an expression that selects the % packet loss metric.

1. Use the “text” tab to build the expression
2. Once you succeed on the “text” tab, check the “tests” tab and ensure both tests pass

Raise your hand when you’re done

Live Node-RED Regex examples

Regex recap

- Regex helps us deal with unstructured data by matching specific patterns, allowing focused searches within large data sets.
- Commonly used for log files, console responses, email bodies, and other unstructured text.
- Ask ChatGPT for help ;)
- Use regexr.com to test

Reflection

Take a break!

Structured Data

- Also sometimes called “semi-structured data”
  - When compared to database schemas, I agree. For our purposes, structured is adequate.
- Methods for parsing structured data depend upon the data structure.
- Primary data structures we’ll deal with are:
  - JSON
  - XML
  - CSV

CSV

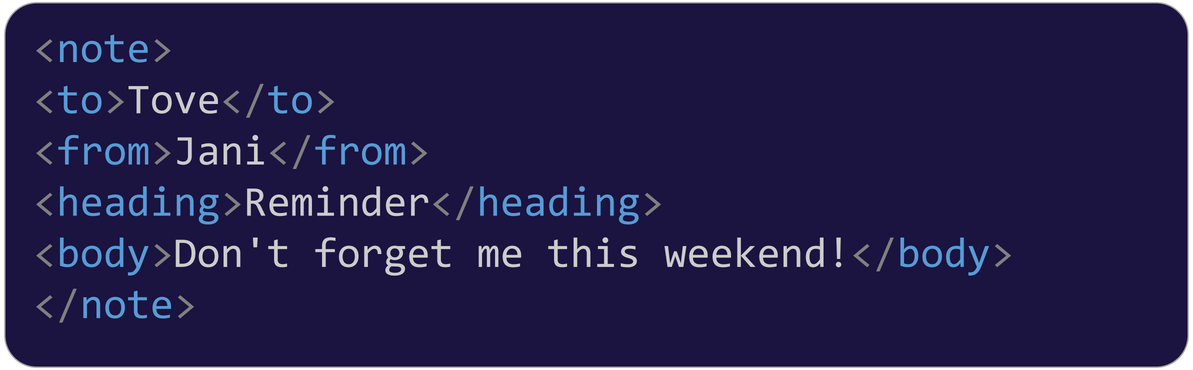
- Comma Separated Values

```
Name,Email,Phone Number,Address
Bob Smith,bob@example.com,123-456-7890,123 Fake Street
Mike Jones,mike@example.com,098-765-4321,321 Fake Avenue
```

Node-RED example CSV

XML

- eXtensible Markup Language



```xml
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Node-RED example XML

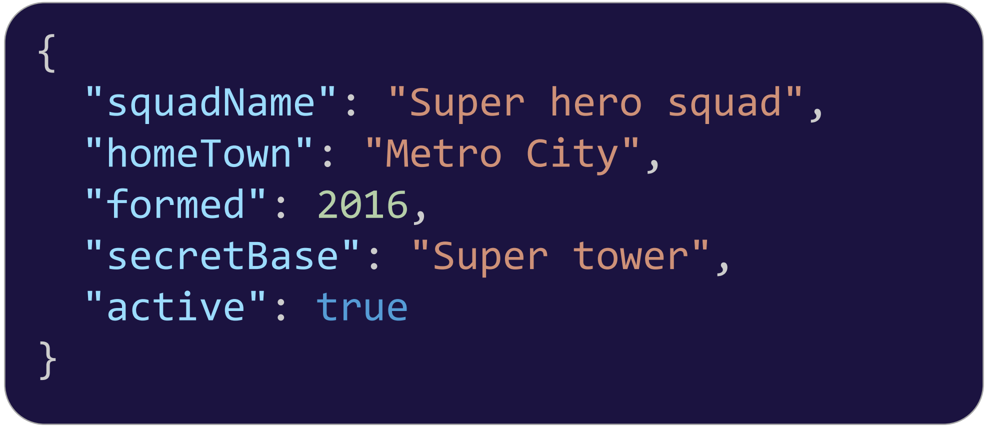
YAML

- YAML Ain't Markup Language

```yaml
---
time: 19:04:12
player: playerOne
action: strike (miss)
speed: 110
thoughts:
    - "That was fast"
    - "I didn't see it"
    - "Where did it go?"
```

JSON

- JavaScript Object Notation



```
{
  "squadName": "Super hero squad",
  "homeTown": "Metro City",
  "formed": 2016,
  "secretBase": "Super tower",
  "active": true
}
```

Parsers & Query Tools

- Parsing JSON, XML, and CSV are generally trivial, native processes to most software applications that handle data.
    - Node-RED has native nodes for all 3.
- Querying or Transforming these data types is where things get interesting.
    - For CSV, we have…Excel!
    - For XML we have XQuery and XPath
    - For JSON we have JSONata and JSONpath

JSON + JSONata

- JSON is the standard data format for monitoring
    - Our goal will always be to get other data formats transformed into JSON
- Thus JSONata makes a great standard query language
- JSONata is a lightweight query and transformation language for JSON data
- Sophisticated query expressions with minimal syntax
- Built in operators & functions user-defined functions
- Format query results into any JSON output structure

JSONata syntax and examples

Practice Question 1

- Go to: https://tinyurl.com/p2dt3nb3
- Copy the data from there to try.jsonata.org
- Build an expression that selects the volume

- **Bonus, build an expression that selects the volume but only if not muted

Practice Question 2

- Go to: https://tinyurl.com/3rkjm2ut
- Copy the data from here into try.jsonata.org
- Build an expression that selects the names of each Zoom Room as an array (list)

Practice Question 3

- Using the same data as before
- Build an expression that shows a list (array) of the rooms with their room names and their current issues, but nothing else:

[{“room”:”some room name”, “issues”: [“issue”]}, …]

Live Node-RED examples

Advanced Transformations

- Sometimes we transform to make data more consumable or usable.
- Sometimes we transform because we need to combine multiple datapoints into something else entirely…like:
  - System Health
  - Confidence Score
- And that is what we'll do tomorrow morning!

Reflection

Lunch!

Successful Outcomes

Section Objectives

Questions:

- What does it mean when the stakeholder says x?

Objectives:

- *Clearly* defined outcomes
- SMART Goals & KPIs to support those outcomes
- Buy-in from stakeholders

Tasks

1. Define the desired outcome
2. Identify goals that will lead to this outcome
3. Identify KPIs to measure progress
4. Write it down and get stakeholder buy-in

Digging into the Outcome

- What is the problem? (5x whys)
- What is the business impact?
- Is there an ROI for solving the problem?
- How will we achieve the outcome?
  - Set SMART goals to move progress forward
  - Select KPIs that will measure our progress
  - Set a threshold for what success looks like

Example Outcome – “Trust”

“I want our users and executives to trust their meeting rooms…to trust the meeting room service.”

- Frank the Technology Manager

Outcome Definition

In your pod, analyze Frank's desired Outcome:

- What is the problem? (dig deep, 5 whys)
- What is the business impact?
- Is there an ROI for solving the problem?
- If we were all on the same team, how will we achieve the outcome?
    - Goals (SMART)
    - KPIs (Measurements for goals and beyond)
- **ASK FRANK FOLLOW UP QUESTIONS**

Sharing: Outcome Definition

Present your success plan to “Frank”.

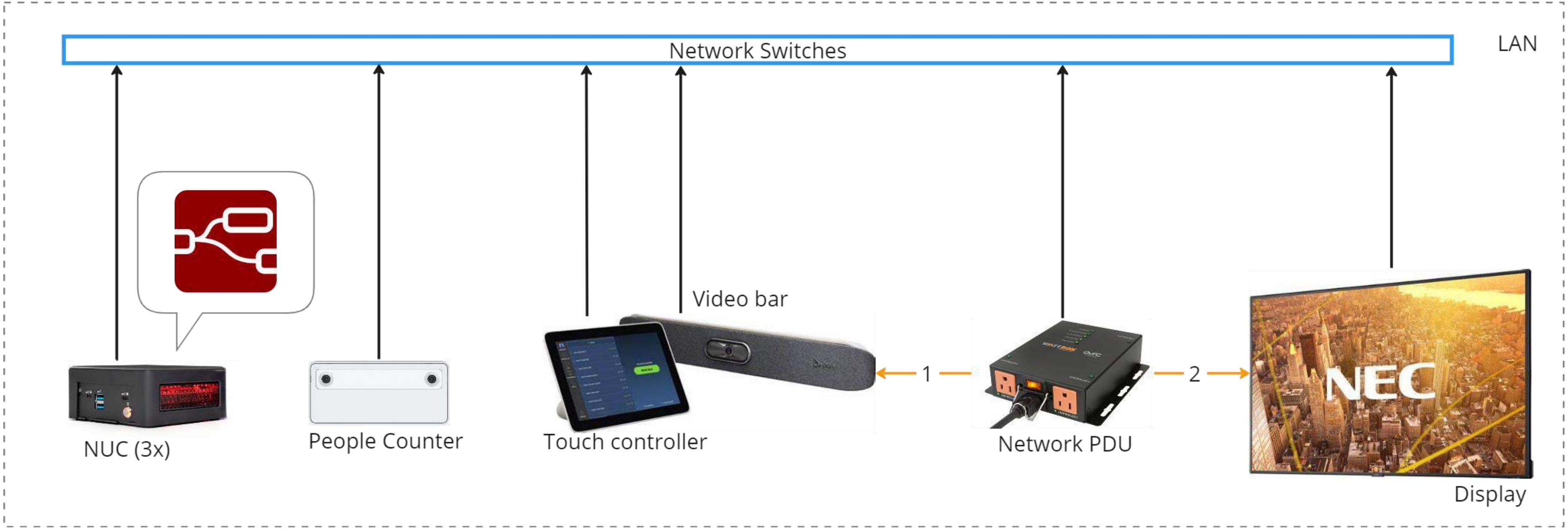
- What is the problem?
- What is the business impact?
- Is there an ROI for solving the problem?
- If we were all on the same team, how would we achieve the outcome?
  - Goals
  - KPIs

KPI Definition

For any KPI you must address:

- Variables
- Methodology
- Mission/Purpose (tie it to stakeholder outcome)
- Possible states / values of the KPI (up/down? number?)
- Example of the KPI

Demo System


Network Switches
LAN
Video bar
NUC (3x)
People Counter
Touch controller
1
2
Network PDU
Display

Health KPI Example Walkthrough

Breakout: Health KPI States

In your pod, discuss what device / system states would trigger our health KPI states

- Consider several states per device in the system
- Think beyond devices…what about Zoom?
- If we could change our design a bit could we get even more important data?
- Reminder: system is a video bar, touch panel, display, and people counter

Sharing: Health KPI States

Share with the group the states you added

- If the display is offline, that triggers an “Outage”, etc
- Are there any devices or services missing in order to provide accurate health?
- Do you think you covered all of the potential states of the system to determine health?

Take a break!

API Research

API Research

Finding API data can be tricky:

- Manufacturer’s websites / docs
- Support
- KB articles
- Google / stack overflow / YouTube
- Developer tools / console
- SNMP walk
- Sleuthing in general

States -> API Calls

For every state in the health KPI you need:

- API interface
- API call
- Auth
- etc

Breakout: API WB-300-IP-3

For each of the states you selected, determine if there is an API available for that data:

- What interface will we collect the data over?
- Are we able to find adequate documentation on the devices?
- Take your time, divide and conquer, see what you can find. Google, call support, phone a friend.
- NEC ME551, Poly X30 / TC8, Axis P8815-2

Sharing: API Research

Share what your pod’s experience was like trying to find this information:

- Did you find all the data you needed? Most? Some? Any?
- What resources were most useful? Where were the best sources of data?

Specify systems based on how manageable they are…how observable they are…and save yourself a lot of time.

Reflection

End of day!

Dawn of

The Final Day

- 24 Hours Remain -

Day 3

API Research & Health Status continued

API Research

After finding API documentation…

- Does this API have the data I need?
- What command do I used to get that data?

API Research gone wrong

What happens if the API doesn’t contain the data you’re looking for?

- Are there other public APIs you can try?
- Can you get the data from a different device in the system?
- Is there a private API you can find?
- Can you respecify the device?
- Do we *reeeeally* need that datapoint?

Let’s find…

- Current connection status to Zoom
- Current display power status
- Current display input status
- Current touch panel pairing status
- Current occupancy

Composing system state

For each of the states in the system

- What state are we trying to get?
- What command will get it?
- Extract the data
- Transform it to our desired format (test!)
- MERGE it with the rest of the system’s state values

The OUTPUT is a composite system state object…the snapshot of the system in a single place.

Can all devices be pinged?

Is the system connected to Zoom?

Is the display powered on?

Is the touch panel paired?

How many people are in the room?

We have composite system state!

“Everything” we need to know about our system in one place.

Now we can **test** our composite system state for system **health**!

Health Truth Table

Highest severity status takes priority!!!

Devices	Operational (base)	Degraded (mid)	Critical (top)
any	Ping ok	Ping some loss	Ping 1%+ loss
Display	Power On		Power Off
Display	Input 1		Not input 1
Display	7142 listening		7142 not listening
Video bar	Using Zoom		Not in zoom mode
Video bar	Touchpanel online		Touchpanel offline
Zoom	Status=noissue		Status!=noissue
People counter		Over occupancy	

We now know, based in the data we get & our expressions if our system is **healthy**.

Next we have to send that somewhere…

Loading Data

(ET)Load

- In this last step, the transformed data is moved from the staging area into a target data warehouse.
- Typically, this involves an initial loading of all data, followed by periodic loading of incremental data changes and, less often, full refreshes to erase and replace data in the warehouse.
- For most organizations that use ETL, the process is automated, well-defined, continuous and batch-driven.

https://www.ibm.com/topics/etl

You can load to…

- Status pages (OneUptime)
- Service monitoring tools (Zabbix, Splunk)
- Databases (SQL, Elastic)
- Etc.

Driven by project / org requirements!!!

Prerequisites to loading

- Destination
- Expected data format
- Type of API (HTTP, SQL, etc)
- Objects required to be in place prior to pushing

Load = pushing data via the platform’s API

Status Pages

Status pages are great for:

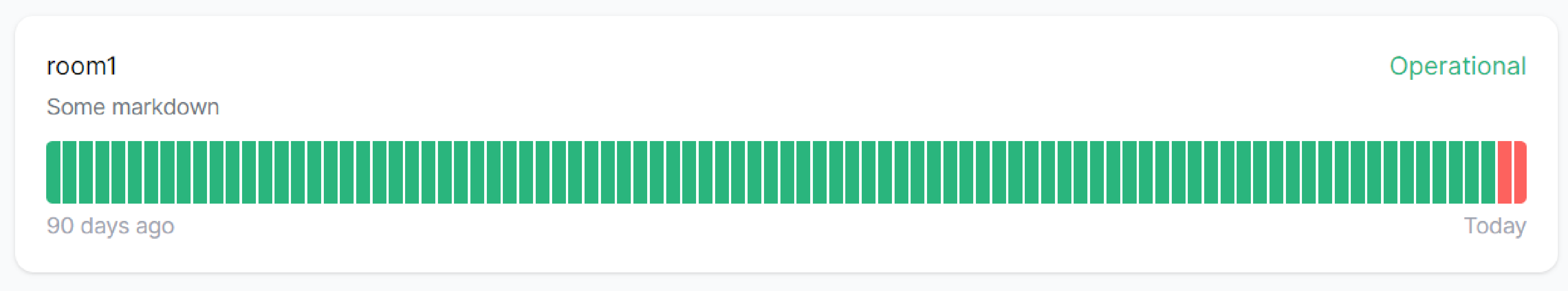
- Showing service (system) status over time which can build trust
- Allowing users to subscribe to status updates
- Creating private pages for authenticated users only
- (Integrators) Decoupling your customer’s status pages from your monitoring tool

We’re going to use a status page from…

OneUptime

Bringing it all together

1. Who cares and why
2. Goals / KPIs
3. Data to collect
4. Device / platform APIs
5. Extracting data
6. Transforming / cleansing
7. Transform to health status
8. Load to platform!



room1

Operational

Some markdown

Bonus: Auto Remediation

Once we **know the status**…

…we should try to solve things **before** reporting an outage.

Health Truth Table

Highest severity status takes priority!!!

Devices	Operational (base)	Degraded (mid)	Critical (top)
any	Ping ok	Ping some loss	Ping 1%+ loss
Display	Power On		Power Off
Display	Input 1		Not input 1
Display	7142 listening		7142 not listening
Video bar	Using Zoom		Not in zoom mode
Video bar	Touchpanel online		Touchpanel offline
Zoom	Status=noissue		Status!=noissue
People counter		Over occupancy	

Reflection

Lunch!

We’ve been building a house…

Picking the “Right” Tools

Crystal clear requirements

Not all tools are created equally! You must build accurate requirements based on:

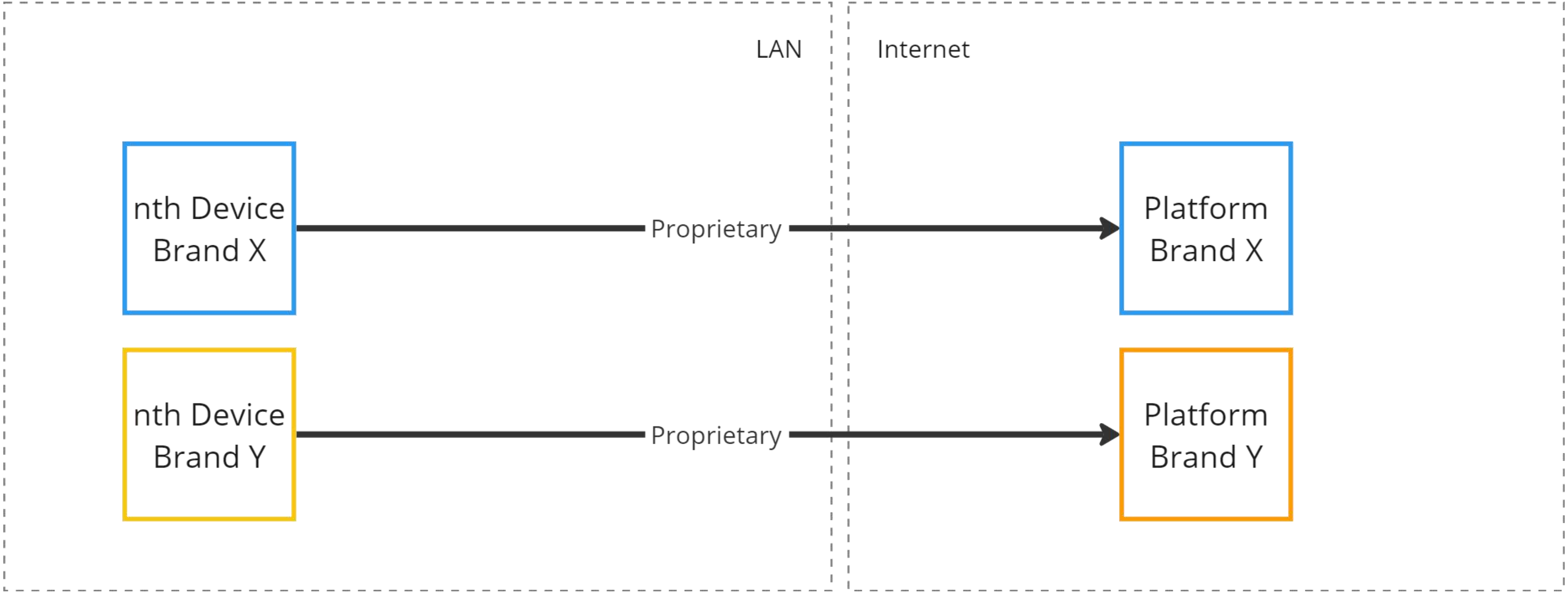
- The business requirements of your stakeholders
  - Outcomes, goals, KPIs!!
- Your knowledge of AV Monitoring practices
  - Protocols, topologies, legacy data extraction & transformation, system/service health monitoring
- The technical details of your systems
  - Devices, platforms, any middleware you may use, etc.

Review of common platform features & functions

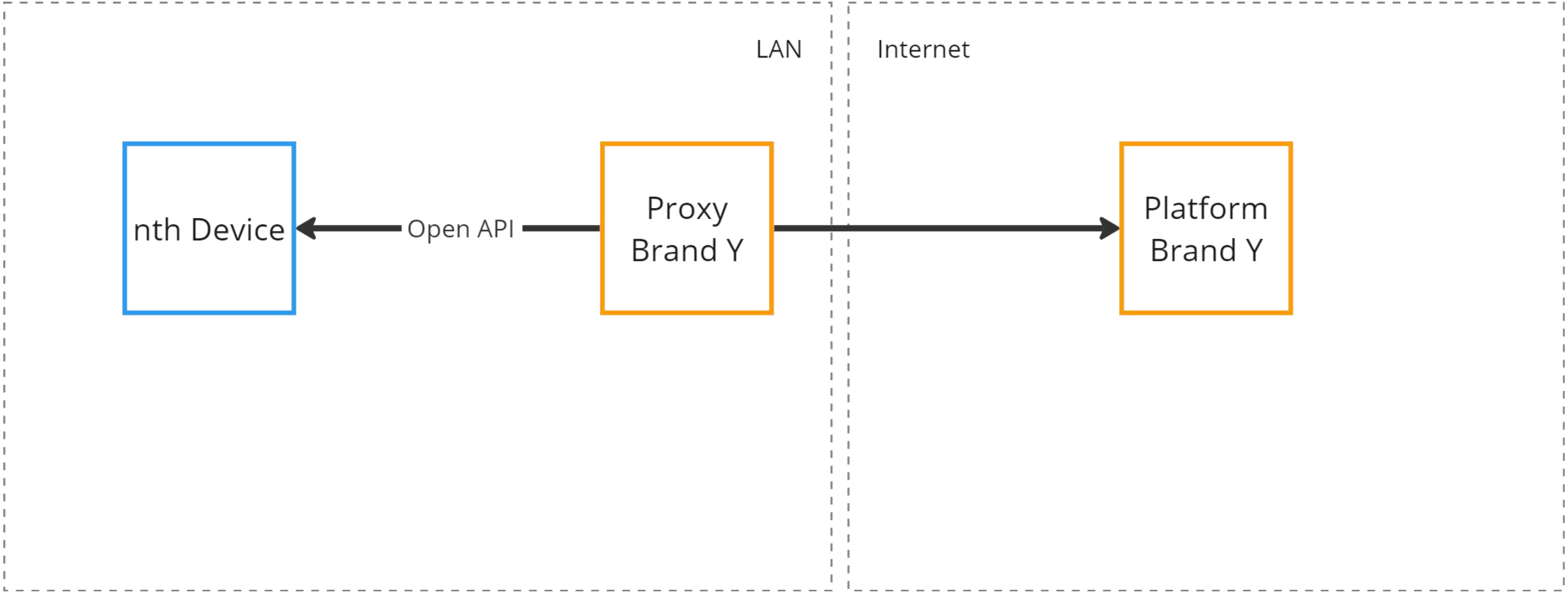
Types of monitoring tools

- Vendor point tools
- Traditional control system-based
- All-in-one control & monitoring products
- IT network monitoring
- IoT platforms
- Status pages
- Database + Dashboard
- Middleware + All of the Above

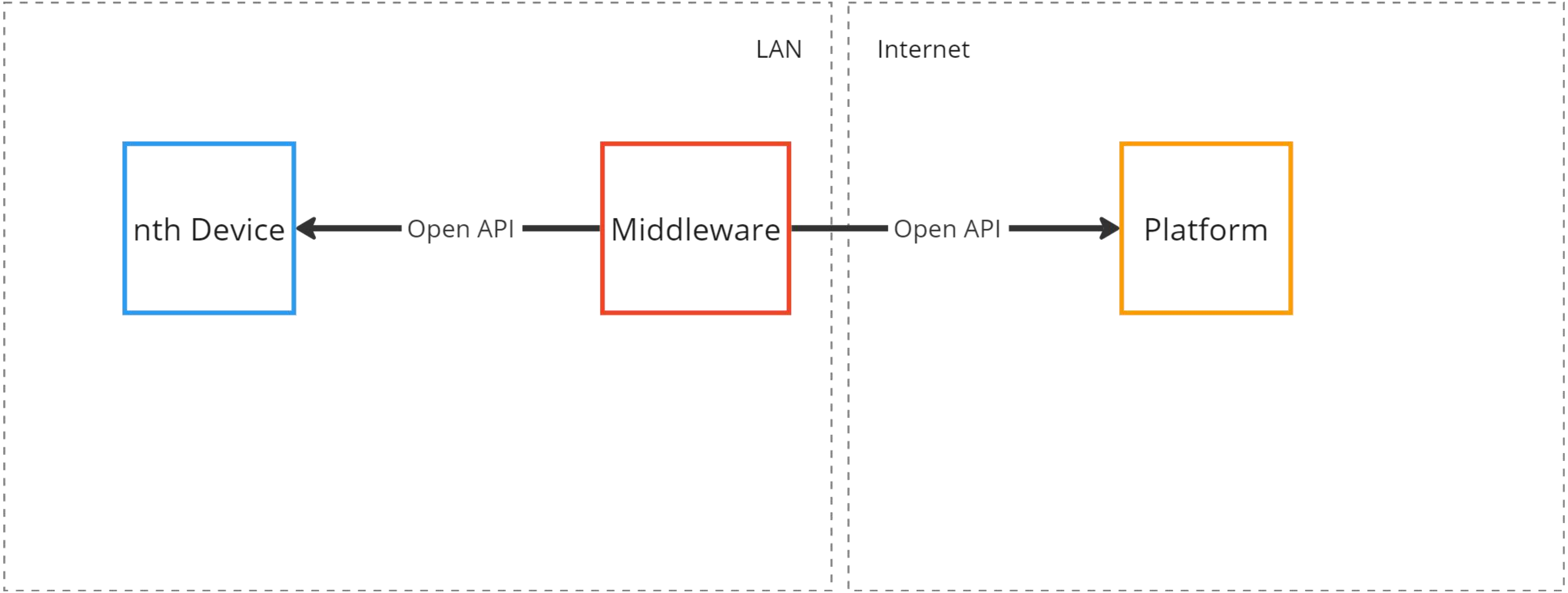
Topologies: Direct



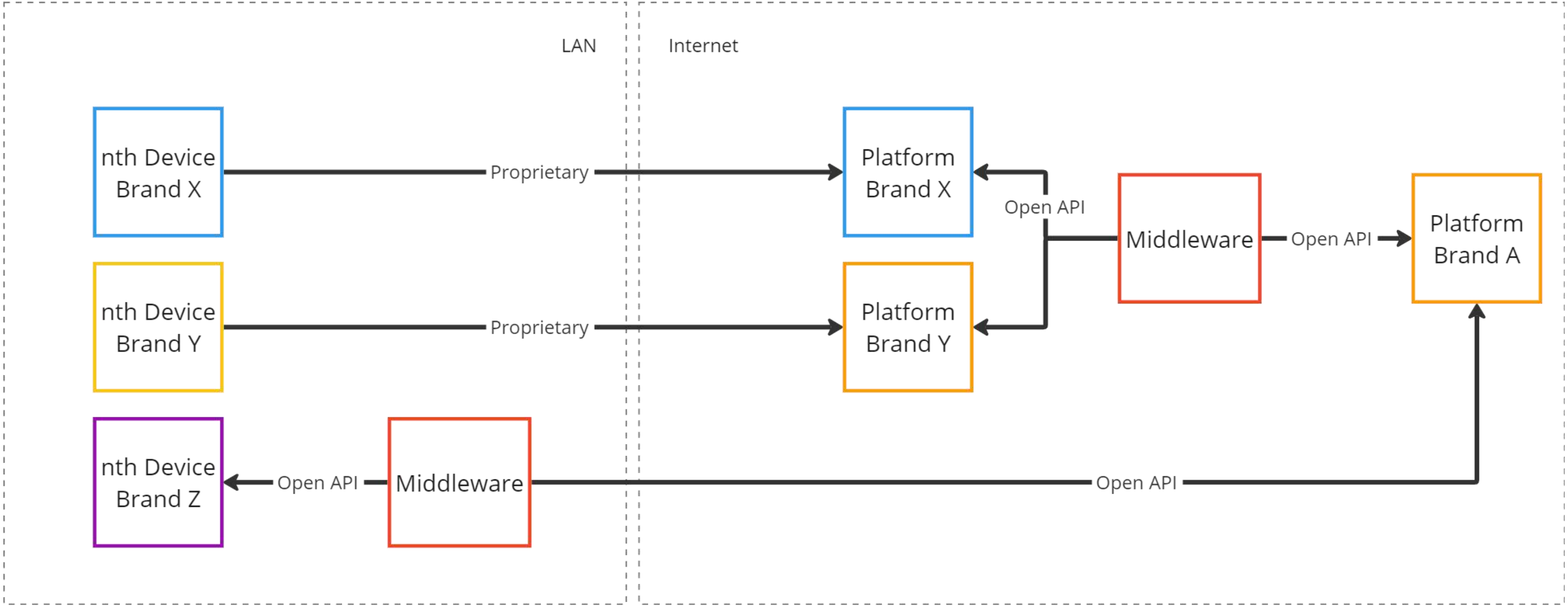
Topologies: Proxied



Topologies: Middleware



Topologies: Middleware+



Features 1 – Extraction Methods

- Ping, port checks, SNMP checks & traps, HTTP/s, Telnet/SSH checks, MQTT topic, custom scripts (JS, bash, etc), OS agent
- Bulk collection…run one command, extract MANY metrics from it (light touch)
- Indirect collection…collect data from child devices (ie serial devices via processor)
- Ingest endpoint / API…allow data to be pushed in from anything!

Features 2 – Discovery

- Device discovery
- Metric (“item”) discovery
- Configurable discovery

Features 3 – Collector

- Privately hosted on-premise (or cloud) collector. Ideally in AV should be a flexible agent that can run on many OS and maybe even an appliance.
- Encrypted connection to platform
- Outbound communication from Collector only…never never never inbound / port forward.

Features 4 – Device Mgmt

- Assignment of hostname / IP
- Asset inventory data
- Device secrets
- Device groups
- Device / group variables
- Bulk actions
- Metrics & variable via drivers/templates (scale)
- Parent/child device relationships

Features 5 – Remote Mgmt

- Execute remote commands / scripts
- Centrally store remote commands
- Proxied browser access to remote web/ssh/rdp
- TCP tunnel access to any device port publicly
- On-demand VPN connection
- Use of switches & PDUs to manage connected devices

Features 6 – User Mgmt

- Role-based access control
- Multi-tenancy (integrators)
- User preferences
- Native auth + SSO auth preferred
- MFA or delegate via SSO

Features 7 – Service Monitoring

- SLI/SLO/SLA tracking & reporting
- Based on device/system state conditions, trigger service availability changes (up, down, etc)
- A service is a room/system

Features 8 – Triggers/Alerts

- Custom triggers, raise issues if threshold is breached
- Send alerts to users or other apps on certain conditions
- Tunable alerts to prevent fatigue

Features 9 – Data Vis/Report

- Custom dashboards
- Custom reports
- Extendable widgets
- Scheduled reports
- Networking mapping (auto)

Features 10 – Data retention

- Store varying telemetry for varying times
- Perform regular housekeeping of expired data
- Store trend data beyond when telemetry is aged out

Features 11 – Extensibility

- Open API
- API-first (anything you can do in the interface…)
- Telemetry ingest / assign to devices
- Native integrations
- Community integrations
- Pluggable architecture
- ITSM integrations

Features 12 – Support/Training

- Community
- Standard / paid support
- Structured training and cert programs
- Solid documentation

Additional Considerations

- Price
- Maturity
- Power
- Community
- Flexibility
- Support
- Activity
- Security
- Your IT team
- Integrations

Specifying your requirements

Requirements

Should be driven by your knowledge + your stakeholder needs / outcomes / KPIs.

- Description
- For the monitoring solution or for the AV system?
- How critical to the success of the project / program / ongoing practice
    - Low -> Critical

Example 1

Description	Mon. Platform or AV Dev/Sys	Importance
The monitoring platform must be flexible and allow for collection of arbitrary data from various sources • Networked AV devices (displays, processors, video bars, etc) • Network infrastructure • SaaS platform APIs (Zoom) • On-Prem platform APIs (Active Directory)	Platform	Critical

Example 2

Description	Mon. Platform or AV Dev/Sys	Importance
The monitoring tool must be able to: • Track SLAs / SLOs / SLIs • Recalculate SLAs on a recurring basis (flexible) • Exclude planned maintenance windows • Track the SLA down to the individual room • Nest SLAs, providing a roll-up SLA • Map telemetry from various sources to SLI states • Clearly define what telemetry state will raise what issue and map to what SLA • Apply an SLA to every room	Platform	Critical

Breakout:

In your pods:

- Document several requirements for a monitoring solution based on what you’ve learned so far.
- Use Frank’s Outcome (“trust”) & the Health KPI to feed some of your requirements.
- Describe the requirement, if the requirement is for the monitoring solution or for the AV system/platform, and how critical it is (1-5 critical)

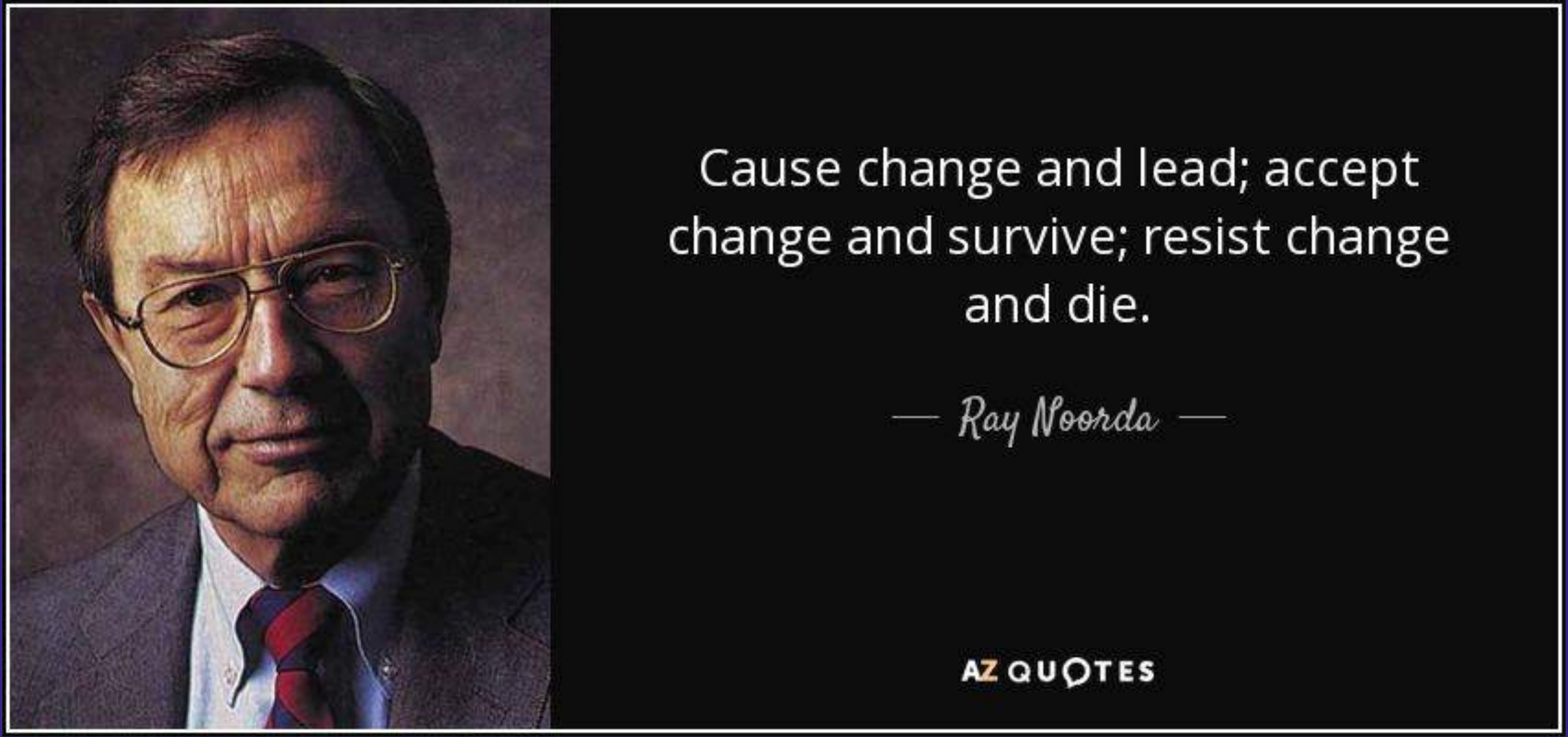
Breakout:

Choose 1-3 requirements to share with the group:

- Description
- Monitoring platform or AV system / service
- Importance

What drove the requirement? Frank’s “trust” outcome? The Health KPI? Something else?

Reflection



Cause change and lead; accept change and survive; resist change and die.

— Ray Noorda —

“Progress, not perfection.”

THANK YOU!

HAVE A GREAT SHOW!!!!!