OPENAIR

Operational Network of Air quality Impact Resources

Air quality sensing device activation and deployment checklist

SR302

October 2023

openair.org.au

Table of contents

Introduction

Air quality sensing devices must be activated prior to deployment. While the exact device activation and deployment steps you follow may vary (according to the devices and/or vendor you are using), this resource will cover some of the general processes that need to be put in place.

Activation of a sensing device refers to all the steps required to prepare a device for active deployment (including onboarding a device to your network, configuration, administration, testing, assembly, and labelling). *Deployment* of a sensing device refers to the physical installation of the device, as well as the supporting activities that result in a fully commissioned, functional asset.

Who is this resource for?

This resource is intended for use by project staff tasked with planning and delivering a smart low-cost sensing network to monitor air quality. It is written with local government in mind, but is relevant to other organisations undertaking similar projects (such as community groups). It is also a useful reference for senior management who seek to understand the complexities of establishing a sensing network.

How to use this resource

This OPENAIR supplementary resource is designed to be used alongside the Best Practice Guide chapter *Air quality sensing device activation and deployment*. This resource is a more detailed, practical guide to activating and deploying smart low-cost air quality sensing devices.

Before you engage with this resource, you should already have completed the planning for your project's sensing device deployment. This is a preceding task that involves working out where you want to deploy your sensing devices to best support the needs of your use case, and confirming practical installation details. For more guidance, refer to the OPENAIR Best Practice Guide chapters *Sensing device deployment planning: high-level design*, and *Sensing device deployment planning: detailed design*.

Prior planning

Please ensure that you have worked through the following planning phases and concrete steps (as part of the OPENAIR Impact Planning Cycle[1]) before you start on your device activation and deployment.

OPENAIR Impact Planning Cycle Stage 2: Develop

- ❑ You have chosen the model and quantity of air quality sensing devices that you will use, and selected a vendor (see the Best Practice Guide chapter *Sensing device procurement*, and the supplementary resources *Technical requirements template*, and *A guide to developing technical requirements*).
- ❑ You have chosen, procured, deployed, and activated the communications technology that will support your data collection (see the Best Practice Guide chapter *Data communications procurement*).
- ❑ You have developed a data management and data sharing plan (see the Best Practice Guide chapter *Sharing air quality data*).
- ❑ You have developed a deployment location plan, including all necessary approvals. This plan should detail all deployment locations and mounting assets[2] (see the Best Practice Guide chapter *Sensing device deployment planning: high-level design*).
- ❑ You have developed a detailed device installation methodology plan, including procurement of components for specific mounting solutions (see the Best Practice Guide chapter *Sensing device deployment planning: detailed design*).

OPENAIR Impact Planning Cycle Stage 3: Implement and operate

- ❑ You have engaged an installation contractor.
- ❑ You have developed a device management and operations plan (see the Best Practice Guide chapters *Sensing device troubleshooting: common problems and how to fix them*, and *IoT system operations*).

[1] The OPENAIR Impact Planning Cycle includes six stages: **Identify**, **Develop**, **Implement and operate**, **Manage and analyse data**, **Act on evidence**, and **Evaluate**.
[2] Mounting assets are the physical infrastructure that you mount a device onto, such as a street pole.

Overview

The steps you will need to take in order to activate and deploy your devices (and full network) are summarised in Figure 1.

The amount of time and work required for device activation and deployment will vary, depending on the technology and procurement decisions you have already made. With more proprietary, 'data-as-a-service' options, the technology vendor may cover many of the listed steps. With more open technologies (particularly where you are integrating several different components), you may need to handle – or closely supervise – each of these steps yourself.

Activation	1	Onboard devices to your network and IoT platform
	2	Device configuration
	3	Establish metadata schema, device administration, and management protocol
	4	Acceptance testing
	5	Prepare hardware for deployment
	6	Co-location calibration *(optional)*
Deployment	7	Installation
	8	Field testing
	9	Deployment metadata completion
	10	Commissioning

Figure 1. Summary of activation and deployment steps

Part 1: Activation

Step 1: Onboard devices to your network and IoT platform

A sensing device needs to be programmed to connect to a specific communications network and IoT platform. This programming process is called ‘onboarding’, and should be done by your device vendor if they are also managing the IoT platform for you (this is the case for all proprietary products).

If you are connecting open-access devices with an IoT platform that is *not* associated with the device vendor, you will probably need to complete part (or all) of this step yourself, or negotiate a collaboration between the device vendor and the platform provider.

Refer to vendor instructions

Sensing devices can vary in terms of how they need to be onboarded, so always start by referring to vendor instructions. The generic process diagram in Figure 2 provides a rough guide to device onboarding.

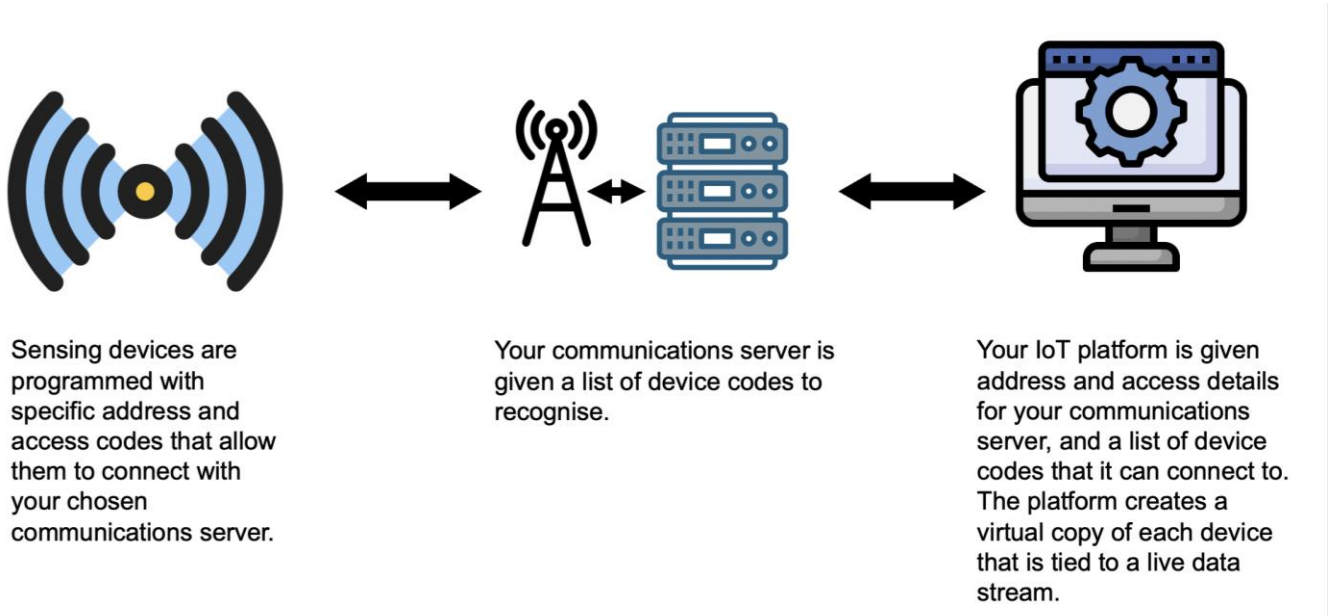


Figure 2. Summary of activation and deployment steps. Icons source: Creative Commons

Challenges and considerations

If your internal project team is responsible for onboarding your open-access devices and connecting them with an IoT platform, make sure you have sufficient in-house expertise to support this process. Make sure that you have sufficient in-house expertise to support this process.

Step 2: Device configuration

Device configuration is often done by the device vendor. However, you will still need to communicate your requirements with them. Some IoT platforms that are tied to specific device types may provide a user interface for configuring device settings.

You should ensure that device settings are configured to support the needs of your project. Common settings to be aware of include:

1. **Sampling rate.** A device will often be configured to collect multiple sample readings from a sensor across a given time period, and then report an average of those values. The sampling rate determines how many samples are included in that average. If your device is deployed in an environment that has highly variable pollution concentrations over short periods (e.g. close to a road), a higher sampling rate can help to flatten the peaks and troughs of that variability, and provide a more representative reading. Be aware that higher sampling rates use more power, which places limits on battery-powered and solar-powered devices. You may need to find a compromise between the need to optimise sampling rate to support required data quality, while also ensuring ongoing device functionality.
2. **Reporting interval.** An IoT device does not transmit continuous data. Rather, it 'reports' by transmitting a data packet at set time intervals. Reporting intervals are commonly set between 10 and 60 minutes. Short reporting intervals can provide greater temporal definition in your data, which may be critical for certain use cases. However, shorter reporting intervals also increase power demand, so you need to balance an optimal reporting interval against power requirements.
3. **Device communications settings.** Devices are often deployed in locations with marginal communications. This may be because terrain, buildings, or trees are blocking the signal, or due to the location's distance from the nearest gateway antenna. Depending on the communications technology you are using, there may be various configurable device settings that relate to how it communicates. These settings help to ensure that a device can reliably transmit and receive data. Two examples (for LPWAN technologies) are 'spreading factor' (how many times the device sends a rapid repetition of the same message), and 'transmission power' (how powerfully that message is transmitted). You need to make sure that you receive a reliable and relatively unbroken data stream without compromising battery life, or extending beyond the supply capacity of your solar-powered system.

Challenges and considerations

These settings are very technical, and should only be engaged with by your vendor. However, it is a good idea to be familiar with these settings, in case you find yourself struggling with data intermittency and device dropouts during your troubleshooting phase.

Step 3: Establish a metadata schema, device administration, and management protocol

You will need to create a master spreadsheet that lists all the devices that you will be deploying, against a list of metadata fields that capture the context in which they will be deployed. For guidance on creating a deployment metadata schema to meet the needs of your project, see the OPENAIR Best Practice Guide chapters *Sensing device deployment planning: detailed design*, and *Data labelling for smart air quality monitoring*.

The key tasks are as follows:

- ❑ Establish your metadata schema, which will consist of all fields (e.g. latitude; street name; deployment date), any associated units of measurement, and field entry validation lists (where necessary).
- ❑ Create a master metadata record that is centrally accessible by key project staff, and agree on access and editing protocols.
- ❑ Populate metadata field entries in your master document for all devices, based upon your deployment plan.
- ❑ Engage with the administrators of your IoT platform and data platform (these two may be the same or separate), and agree on:
    - what metadata from your master document will be captured within these platforms (as opposed to remaining only within the master document)
    - how the metadata schema will be adopted (including the structure and storage of metadata, API design, metadata visualisation, and its integration into the functional aspects of the platforms)
    - how metadata field updates will be made, and by whom.

Challenges and considerations

All metadata should serve a purpose. When you develop your metadata schema, the following approaches are strongly recommended:

1. You should have a detailed device deployment plan in place before you start to develop your metadata schema. Ideally, you should have visited all of your device deployment locations, taken detailed photographs, gained approvals, and defined solutions for mounting, power supply, and communications. By engaging directly with these practicalities, you will rapidly determine which deployment metadata fields are relevant to your project context and data use case. See the OPENAIR Best Practice Guide chapter *Sensing device deployment planning: detailed design* for further guidance.
2. Deployment metadata is critical for interpreting sensor data. If you are planning to address certain research questions using your data, these should shape your choice of appropriate metadata fields. For example, if you plan to compare air quality measured by sensing devices at

two different locations, you could include ‘deployment height’ as a metadata field (e.g. one device is deployed two metres off the ground, and the other is deployed at four metres).

3. You should engage with your data users to help you define a data schema that is practical and useful for your organsaition. Each metadata field should meet an end user need. End users in this context include the people who will be responsible for operating your sensing network (device and network management), as well as the users of the data you are producing.

Step 4: Acceptance testing

Acceptance testing is critical for effective verification and troubleshooting after a device is purchased. It is conducted prior to device deployment, and aims to confirm two things: firstly, that the device functions correctly with respect to power usage, communications, and data capture; and secondly, that reported data looks ‘sensible’ (in line with what you might reasonably expect). Once confirmed, any issues that arise following deployment should be attributable to the location, or related issues (e.g. damage during installation).

To conduct acceptance testing, identify a secure indoor location with continued access for people, that will not be disturbed for several days. It must have constant, reliable connectivity using your chosen communication technology (e.g. be within range of a nearby LoRaWAN gateway). Solar devices that lack a direct power input will need a location in a north-facing window, with solar panels connected.

Note that if your device includes meteorological sensors (e.g. to measure wind or rain), then acceptance testing may need to be done outdoors, using a suitable roof, balcony, or other secure outdoor area.

- ❑ **A. Verify device functionality**
    - The device power supply functions as expected.
    - Voltage output is stable and constant.
    - Solar and battery recharge/discharge cycle is reliable.
    - The device connects reliably to the communications server, and maintains the connection.
    - The device reports data in accordance with configured reporting interval (e.g. once every 15 minutes).
    - Data packet content is complete. This means that all the information expected to be contained in each data packet received from the device is indeed present.
    - The device continues to send data packets in accordance with the set reporting interval, for a period of at least three days. Device availability (which measures the number of data packets received vs expected) should be at least 75%.

❑ **B. Verify that data looks sensible**

- The device reports data for key parameters within the bounds of what is physically possible (e.g. a temperature reading of 40°C is hot but possible, whereas a reading of 500°C is not possible, and indicates a technical fault).
- The device reports data that is sensible relative to the environment (e.g. temperature is reported as 22°C, which is sensible for an air-conditioned office), and/or matches a reference instrument, where available (e.g. temperature matches the reading from a wall thermometer in the office by +/-1°C).

This is also an opportunity to confirm the flow of data from your devices into your IoT platform and data store. You can check the visualisation of test data through the platform, and engage with the platform provider regarding customisation of the interface (if desired).

At the end of acceptance testing, you should have a device that you know is operating correctly under ideal conditions. It sends data correctly and reliably, and – at a basic level – this data looks trustworthy. This means that any problems that emerge following device deployment can most likely be attributed to the location (e.g. marginal communications coverage). Acceptance testing is not the same as co-location calibration (covered in **Step 7**), which aims to verify data quality more precisely.

Challenges and considerations

When you do an acceptance test, the beginning of your data record for a device will not be tied to its final deployment location. For this reason, it is vital that you keep an accurate record of the deployment date, allowing you to create a cut-off point in the data record. This ensures that any data from the acceptance test is not accidentally incorporated into future data analysis.

WHAT IS A DATA PACKET – AND HOW DO I VERIFY IT?

A data packet is a 'report' made by a sensing device at recurring intervals. It is transmitted through the communications server (regardless of the technology used) and enters the IoT platform, where it is received by a piece of software called a 'packet decoder' (which converts it into human-readable information).

It is possible for a device to experience a fault (either due to physical damage, or a bug in its firmware or configuration). This results in incomplete data packets being sent, or zero/null/negative values being sent for certain fields within the data packet. As part of testing, you should verify that your data packet content is complete, and is appearing as expected.

There are a few ways to do this. For devices that use LPWAN communications (e.g. LoRaWAN; NBIoT; Sigfox) you may be able to access the communications server directly through a user interface, and verify the raw data packet. Most device manuals provide an example of a data packet, so you can compare what you see to what you *should* be seeing. This is the best practice approach for verifying data packets because you are making an assessment of the raw data, rather than decoded data (where the fault may lie with the decoder, rather than with the device).

If you cannot access your communications server, or if you would like a simpler and more rapid means of verifying data packets, there is another option that is somewhat imperfect, but still helpful (in some cases). Regardless of the communications technology used, device data should be arriving in your IoT platform, and being interpreted into human-readable information. Refer to the device manual, and find the list of data that is supposed to be received in each packet. Can you see it all appearing in your IoT platform? Check values over time, and if there are flat zero, null, or negative values, this may indicate an issue.

The reason this approach is considered 'imperfect' is that data in your IoT platform that appears to be missing (or is otherwise strange) may not be an accurate reflection of device functionality. You are viewing the output of a decoder, and decoders can have faults – in fact, this is quite common. Thus, you may identify what appears to be a device issue, but is actually a decoder issue. Best practice recommends that you check data packets at the level of the communications server whenever possible, so treat verification in the IoT platform as only a secondary backup method.

Step 5: Prepare hardware for deployment

During this phase, you will need to take the following steps:

- ❑ Assemble devices, power supply, and mounting components to be ready for deployment.
- ❑ Ensure that device-specific metadata is up-to-date in your *master metadata record*, as this may change from your initial deployment plan, especially as you finalise the details.
- ❑ Add weather-proof labels to devices (e.g. name; serial number; owner; contact number).
- ❑ Prepare detailed installation documentation and instructions for use by an installation contractor. The more detail you can provide, the lower the chance of mistakes being made, saving you time and expense later.

Recommended documentation includes:

- A generic installation methodology document, detailing each major type of planned installation (e.g. city-owned light poles). Include schematics with details of mounting assemblies, height above ground, etc.
- Annotated photos of each individual deployment site. These should include map coordinates, and clearly identify where and how a device is to be installed.
- A checklist of all parts associated with each installation (e.g. steel straps; nuts and bolts).
- Additional master reference documents that itemise all devices and their locations, and include a form for capturing ‘as-installed’ metadata variations (e.g. confirming actual installed height above ground).
- Prepare device installation packages for handover to your installation contractor. It is recommended that you box up each assembled and labelled device with its own specific installation instructions, and clearly label the box.

Challenges and considerations

Some installers may not be familiar with the task of sensing device deployment. The more time and effort you can invest up front (by producing clear, detailed documentation and instructions), the lower the chance of mistakes and complications later on. Annotated photographs are highly recommended.

Be aware that batteries are often difficult to send via regular mail delivery services. For this reason, devices are often shipped without batteries, requiring you to procure and install them yourself. If you need to courier or mail complete device deployment packages to an installer (including batteries), this could delay your project. Ideally, try to engage a local installer who can collect packages in person. If you need to send batteries in the mail, check with your courier company regarding their policy.

Step 6: Co-location calibration (*optional*)

This step may or may not be required for your project. Please consider the following information carefully to support your own assessment of whether this step is necessary.

Co-location calibration involves temporarily deploying sensing devices in close physical proximity to an authorised air quality reference station that is situated in (or close to) the area where you intend to deploy your devices. This is done to understand how the data from your smart low-cost sensing devices compares to data from highly trusted reference equipment, and how local environmental conditions interfere with sensor performance. This comparison allows for the determination of correction factors, which support calibration of the device, and more accurate (and usable) data output.

There are three main factors to consider:

1. Why devices vary in performance (and why calibration co-location matters)
2. Whether you need to co-locate your devices to achieve your aims
3. Practical approaches to undertaking device co-location, if required.

Practical approaches to undertaking device co-location

If you *do* decide to co-locate, take the following steps:

- ❑ Identify regulatory air quality monitoring stations in your area, and contact the relevant state authority to discuss your project, and secure access.
- ❑ Ensure that you have approved use of an appropriate and secure space for device co-location, with adequate accessibility and power supply.
- ❑ Ensure strong communications coverage. You may need to install a local gateway (e.g. LoRaWAN or Wi-Fi).
- ❑ Visit the site to confirm a specific, detailed mounting solution that will position your device in an experimentally appropriate position. Seek guidance from the regulatory authority on how best to achieve this.
- ❑ Cost, procure, and install the planned mounting solution.
- ❑ Physically install your co-located devices.
- ❑ Allow a period of at least three days to confirm and verify data, prior to the main data collection period.
- ❑ Collect co-location data for at least one month.
- ❑ Apply correction factors to each individual device (a process carried out either by you, or your vendor).
- ❑ Retrieve device, and prepare for main deployment.

For in-depth advice, refer to the U.S. Environmental Protection Agency resource *How to Evaluate Low-Cost Sensors by Collocation with Federal Reference Method Monitors*.

Challenges and considerations

Co-location calibration takes time (at least a couple of months of planning and delivery). It can be a significant expense, and may or may not be a critical step for your project. The following information boxes will help you to understand why co-location can be important, and whether it is something you need to do. For more detailed discussion of this topic, see the OPENAIR Best Practice Guide chapter *Sensing device calibration*.

WHY DEVICES VARY IN PERFORMANCE (AND WHY CO-LOCATION MATTERS)

1. Individual sensor variability
Sensors can vary in their performance due to small inconsistencies in the manufacturing process, meaning that no two sensors (or devices that contain them) will perform in exactly the same way. This variability is most pronounced for gas sensors. Co-location can support the creation of device-specific correction factors.

2. Device design
The overall performance of different devices that incorporate the same basic sensing components can vary considerably because of device design (hardware layout, electronics, and software settings). Co-location allows you to assess the actual performance of a sensor as part of a complete device system, under local conditions, and against a trusted reference.

3. Local environmental conditions
Environmental conditions (notably, temperature and humidity) impact the performance of sensors, and can vary by locality. Co-location enables you to understand the relative impact of these variables, and to apply locally specific correction factors for them.

DO I NEED TO CO-LOCATE?

Your decision should be based upon consideration of the following factors:

1. Does your data use case demand accurate gas sensing?
   - ❑ If YES, you should co-locate.
2. Does your data use case demand *highly* accurate data (of any type)?
   - ❑ If YES, you should co-locate your devices, regardless of which parameters you are measuring.

     NOTE: *For particulate sensing, it is accepted practice to co-locate a representative sample of devices (three at a minimum) to obtain a locally specific correction factor to be applied to your other devices.*
3. Is co-location realistically achievable for you?

Time, budget, or access constraints may present challenges. If so, you may need to rethink your project aims, so that co-location is not critical for achieving them.

Part 2: Deployment

Step 7: Installation

The following steps should be taken when installing a sensing device:

- ❑ Arrange for collection of prepared devices (and accompanying documents) by your chosen installation contractor. Ensure that you have an agreed process for contractors to deliver 'as-installed' documentation and photographs back to you after installation is complete. Make time for briefing discussions, to ensure that everything is clearly understood. Overseeing and approving one complete installation (particularly if many installations are planned) is also advisable.
- ❑ Engage installers to physically install the devices.
- ❑ Instruct installers to take photographs of each installed device. These are a vital part of the 'as-installed' documentation for your device network, and should include:
    1. A close-up photograph that captures the detail of how the device is mounted.
    2. A context shot that captures the broader spatial setting. Make sure that your contractor has a suitable camera, and be clear about your photography requirements during the briefing. Taking demonstration shots with your contractor can be a good idea.
- ❑ Require installers to complete the 'as-installed' metadata form provided (e.g. date; time; height; orientation).

Challenges and considerations

There is a risk of installers failing to complete installations according to approved methodology, or failing to document installations correctly, which can result in significant delays to your project. To avoid mistakes, make extra time for quality control at the start.

The ideal approach is to undertake an initial trial installation, where you are present on-site with the installer. Go over everything step-by-step, and make sure that both parties are happy with the process and the result. This is also an opportunity to discuss installation records, and any documentation photographs you require. If it is not possible for you to be present for a trial installation, aim to have a dedicated briefing session with your contractor to discuss the components and paperwork.

Step 8: Field testing

- ❑ Following device installation, allow a period of at least a week[3] to verify device installation, device operation, and data quality in the field. There are several steps to this process, summarised in Figure 3.
- ❑ Keep notes on any deviations (e.g. if a device was installed at a different location, or in a different way, to what was originally planned). These notes will need to be reconciled with your metadata record.

This verification process described above focuses on devices, and their ability to send data as far as your communications server. This is not the same as checking for them on your IoT platform, because there can be issues with data decoding or presentation at the level of the IoT platform (which may have nothing to do with device functionality, or data quality).

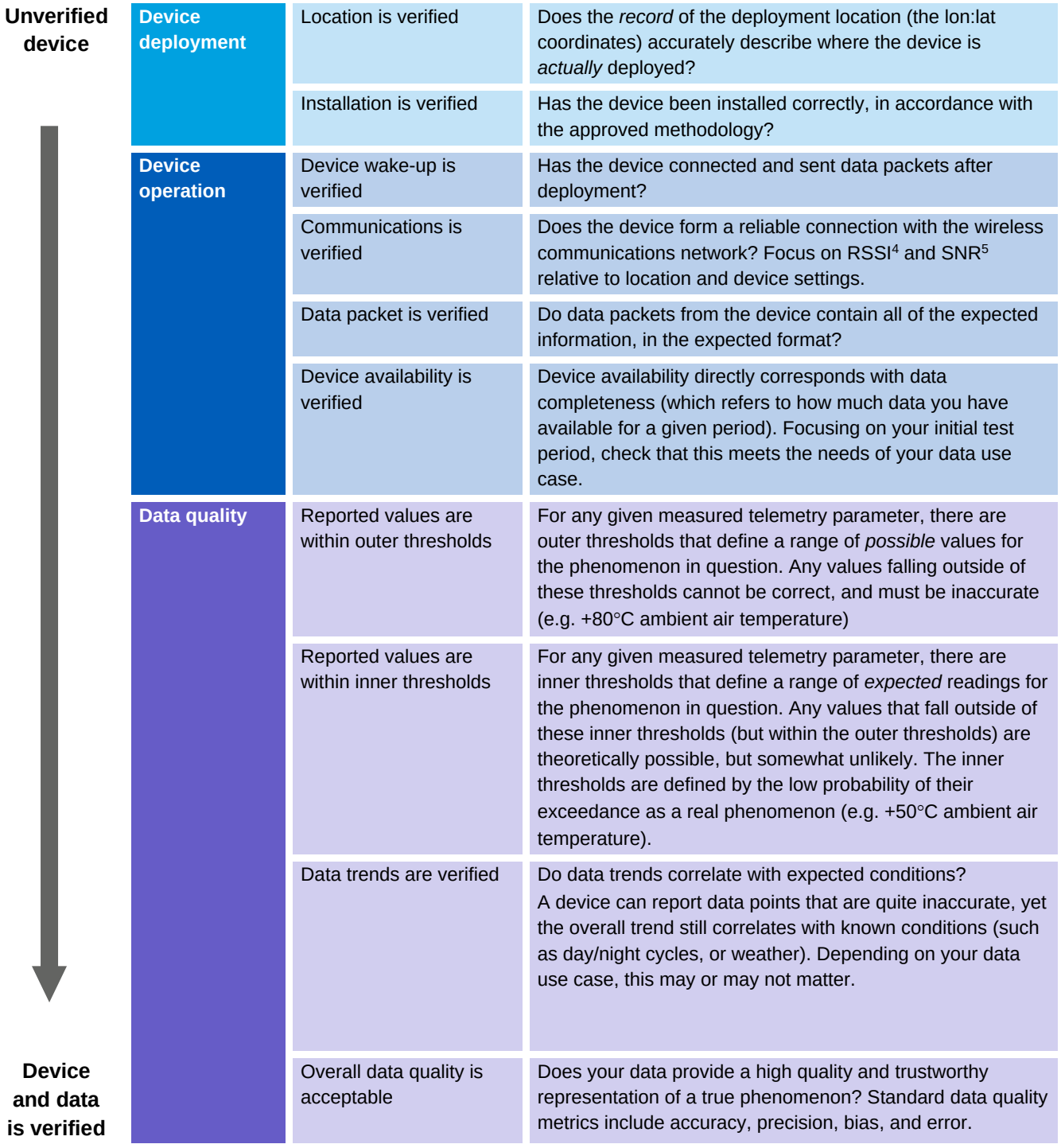
Basically, you want to avoid confusing device-related issues with issues occurring further up your data stack. In practical terms, it is acceptable to use your IoT platform to support device and data verification. However, you should do so with a degree of caution, and a clear awareness that issues identified may be platform-related (rather than device-related). You should first aim to verify your devices and data quality, and then verify IoT platform functionality.

Challenges and considerations

Be careful about using your IoT platform as the sole means of verifying device functionality. While it is reasonable to start there, any apparent issue with a device needs to be followed up at the level of your communications server (which sits at the lowest level of your data stack, and should have an accessible user interface of its own).

By reviewing device status and data packets at the level of the communications server, you can rule out any faults occurring within the IoT platform (e.g. a bug in a data decoder or data visualisation element). You can also check whether the fault lies within the communications layer itself (e.g. a gateway or server outage). Ensuring access to multiple levels of your data stack allows you to pinpoint where problems are occurring, and is therefore critical for troubleshooting and ongoing operational management of your sensing network.

[3] One week is the recommended minimum period for data collection to verify device functionality. However, please note that the verification and troubleshooting process can take much longer than this.



Device deployment	Location is verified	Does the *record* of the deployment location (the lon:lat coordinates) accurately describe where the device is *actually* deployed?
	Installation is verified	Has the device been installed correctly, in accordance with the approved methodology?
Device operation	Device wake-up is verified	Has the device connected and sent data packets after deployment?
	Communications is verified	Does the device form a reliable connection with the wireless communications network? Focus on RSSI[4] and SNR[5] relative to location and device settings.
	Data packet is verified	Do data packets from the device contain all of the expected information, in the expected format?
	Device availability is verified	Device availability directly corresponds with data completeness (which refers to how much data you have available for a given period). Focusing on your initial test period, check that this meets the needs of your data use case.
Data quality	Reported values are within outer thresholds	For any given measured telemetry parameter, there are outer thresholds that define a range of *possible* values for the phenomenon in question. Any values falling outside of these thresholds cannot be correct, and must be inaccurate (e.g. +80°C ambient air temperature)
	Reported values are within inner thresholds	For any given measured telemetry parameter, there are inner thresholds that define a range of *expected* readings for the phenomenon in question. Any values that fall outside of these inner thresholds (but within the outer thresholds) are theoretically possible, but somewhat unlikely. The inner thresholds are defined by the low probability of their exceedance as a real phenomenon (e.g. +50°C ambient air temperature).
	Data trends are verified	Do data trends correlate with expected conditions? A device can report data points that are quite inaccurate, yet the overall trend still correlates with known conditions (such as day/night cycles, or weather). Depending on your data use case, this may or may not matter.
	Overall data quality is acceptable	Does your data provide a high quality and trustworthy representation of a true phenomenon? Standard data quality metrics include accuracy, precision, bias, and error.

Figure 3. Complete device and data verification process

[4] Received Signal Strength Indicator
[5] Signal to Nosie Ratio

Step 9: Deployment metadata completion

- ❑ Update the metadata in your *master metadata record* with 'as-installed' information. This might include deviations from planned installation instructions (e.g. a device was deployed 0.5m higher than planned because a road sign was in the way).
- ❑ Update the device metadata record following field testing, using notes on any deviations (see Step 8).
- ❑ Updates to your *master metadata record* will likely need to be captured by your IoT or data platforms. Ensure that these updates are carried out in accordance with the agreed editing process (e.g. inform key parties of any updates).
- ❑ Create a supplementary 'as-installed' document to capture photographs of each device installation.
- ❑ Use a digital mapping program for capturing 'as-installed' device locations and metadata, to create a baseline record that is independent from your IoT platform (which will update over time). Google Maps is a useful, free, and appropriate tool for doing this if you do not have access to an internal GIS platform.

Challenges and considerations

For even a modestly sized sensing device network, the time required to fine-tune a data schema, and populate 'as-installed' metadata, should not be underestimated. Device deployment can involve a steep learning curve, particularly if you are new to it. Your metadata schema is likely to require updates (e.g. new fields, validations, or associations between fields) based on local realities that will only become apparent through practical engagement with the deployment process.

These updates may then need to be implemented within your IoT and data platforms, which may require extensive technical support. The process of capturing metadata entries for all the fields can also be lengthy, as it is contingent upon completion of the verification and troubleshooting process (which can itself last for weeks, or even months). You should anticipate this by allowing extra time in your project delivery plan.

Step 10: Commissioning

Commissioning is the final step of the deployment process, culminating in sign-off by your project team or organisation, and an official shift into an 'operations' phase. Your organisation will likely have existing policy relating to this, but there are a few general steps you should follow in this stage:

- ❑ Compile and publish (internally or externally) all 'as-installed' documentation. Ideally, this should include a photographic record and installation notes, paired with your *master metadata record*.
- ❑ Sign-off on all completed work by contractors.
- ❑ Inform all relevant stakeholders that full network deployment has been completed, and circulate 'as-installed' documentation (as appropriate).

NO SENSING NETWORK IS SET-AND-FORGET

Commissioning your new sensing network is a major achievement.

Once it is up and running, you will need to actively manage your network, regardless of the technology choices you have made, or the complexity of your deployments. Devices need to be regularly checked to ensure optimal functionality, and routine maintenance is advisable to avoid failures.

The following OPENAIR resources provide further guidance on this topic:

- Best Practice Guide chapter *Sensing device troubleshooting: common problems and how to fix them*
- Supplementary resource *Sensing device troubleshooting: extended guide*
- Best Practice Guide chapter *IoT system operations*.

Additional resources

U.S. Environmental Protection Agency* | *How to Evaluate Low-Cost Sensors by Collocation with Federal Reference Method Monitors.

Associated OPENAIR resources

Best Practice Guide chapters

Air quality sensing device activation and deployment

This Best Practice Guide chapter provides guidance on activating and deploying smart low-cost air quality sensing devices.

Sensing device *deployment planning: high-level design*

This Best Practice Guide chapter explores the high-level design of a smart air quality monitoring network. It provides general guidance for selecting where to deploy devices, what to mount them on, how to mount them, and how to support their operation.

Sensing device deployment planning: detailed design

This Best Practice Guide chapter explores the detailed design of a smart air quality monitoring network. It builds upon high-level design activities, and provides guidance for planning and documenting the details of specific device deployments.

Sensing device procurement

This Best Practice Guide chapter provides guidance on the selection and procurement of smart low-cost air quality sensing devices. It explores critical considerations relating to the design and functionality of devices and the quality of the data they produce, supporting procurement choices that are appropriate to the needs of a project and organisation.

Data communications procurement

This Best Practice Guide chapter explores the various communications technologies that can support smart low-cost air quality sensing, and provides advice on selecting technologies that are appropriate to a project and organisation.

Sharing air quality data

This Best Practice Guide chapter provides guidance on the sharing of air quality data. It explores the process by which a local government might assess data to determine its shareability, and presents a series of practical options for implementing data sharing.

Sensing device troubleshooting: common problems and how to fix them

This Best Practice Guide chapter introduces a framework of common problems that can arise with smart low-cost air quality sensing devices and the provision of useful data. It includes some practical information to help diagnose issues, fix them, and mitigate against reoccurrence.

IoT system operations

This Best Practice Guide chapter provides guidance on the technical operation of an air quality monitoring network as a complete IoT system (comprising multiple devices, communications systems, software/platforms, databases, and digital services). Effective operation of these systems ensures a reliable supply of air quality data, and ensures that data is stored, accessed, and used in accordance with the needs of a project and organisation.

Data labelling for smart air quality monitoring

This Best Practice Guide chapter provides guidance on data labelling for smart air quality monitoring. It provides advice on developing and implementing a project data schema (which defines all of the telemetry and metadata that will be used in a project).

Sensing device calibration

This Best Practice Guide chapter provides guidance on the calibration of smart low-cost air quality sensing devices. It discusses calibration, co-location, decision-making, and developing and following a plan.

Supplementary resources

Technical requirements template

This template is an extended, step-by-step tool that supports the development of technical requirements for a smart air quality monitoring project. These requirements define the details of technologies (sensing devices, platforms, and services) that can meet the specific needs of a project, and are intended to support procurement decision-making.

A guide to developing technical requirements

This resource is a companion guide to the technical requirements template.

Sensing device *troubleshooting: extended guide*

This resource presents an extended, systematic list of problems that can arise with smart low-cost air quality sensing devices and the provision of useful data. It includes practical information to help diagnose, fix, and mitigate each type of issue.

Further information

For more information about this project, please contact:

Peter Runcie

Project Lead, NSW Smart Sensing Network
Email: peter@natirar.com.au

This supplementary resource is part of a suite of resources designed to support local government action on air quality through the use of smart low-cost sensing technologies. It is the first Australian project of its kind. Visit www.openair.org.au for more information.

OPENAIR is made possible by the NSW Government's Smart Places Acceleration Program.

Document No: 20231027 Air quality sensing device activation and deployment checklist Version 2 Final