

Field note

Localising airport systems: what to move, and who should own it

Localisation is usually argued as sovereignty. The stronger case is economics: every vendor at an airport duplicates the same local layers, and consolidating them once across vendors and airports is where a national programme finds its scale.

September 2026 · 5 min read · For operators and authorities; vendors and suppliers

The local share of every airport system

An airport system is never wholly cloud, however it is sold. Each deployment carries a local share: implementation and delivery, field support, the LAN and WAN it runs over, integrations with the airport's other systems, the hardware and touchpoints in the terminal, and release management once it is live.

Walk through a mid-sized airport and count how many times that local share has been built. The operational database vendor has its own monitoring, remote access, patching and backup tooling. So does the passenger processing vendor, the baggage vendor and the display vendor. Each has a service desk, field engineers, spares and out-of-hours cover. In some airports each has run its own cabling and edge network where one airport-managed network would have served them all. None of this is anyone's fault. It is what happens when systems are procured one at a time.

Vertical scale and horizontal scale

Cloud and community providers win vertically. One product line serves many airports in many countries, so a feature built for one ships to all, and the platform beneath it is shared across the whole customer base. The economics are strong for the provider and, for a single airport buying a single system, strong for the buyer too.

An in-country platform partner wins horizontally. One infrastructure, network, hosting, security, monitoring and field-support environment serves many vendors across many airports in one country. For a single airport those economies barely exist. Across an airport group, with a mandate behind it, they match the cloud provider's and add what cloud cannot supply: sovereign hosting and operation, resilience when the connection to the world is cut, local touchpoint support, and one local counterparty for every vendor.

That is the case for localisation in commercial terms. Sovereignty and regulation set the floor; the duplicated local share is where the money is.

Decide layer by layer

The mistake is to argue about whether a system should be localised. Systems are not localised; layers are. Facilities, network, hosting, platform, application, data, integration, and the operations and security tooling that runs them all: each is a separate decision, with a separate answer about who should own it and how it should be shared.

For each layer the question is whether it is shared across operators and vendors, segregated by operator on common infrastructure, or dedicated to one system. The lower layers are usually the easiest to share and the first a local partner can credibly own. The application and data layers are where the argument is won or lost, and where the answer differs system by system. Record the answer from evidence; do not assume it.

Keep the three states apart

As-is, to-be and the path between them are three different pieces of work, and each should be accepted on its own evidence. The as-is is fact: what runs where, under which contract, supported by whom, until when. The to-be is recommendation: the localisation position and operating model for each system and layer. The path is sequence: what moves first, what waits for a renewal, what is designed onto the platform from the start. Mixing them lets a weak recommendation hide behind a strong baseline, and a plausible sequence hide behind an untested recommendation.

Operating models sit on a spectrum

At one end the airport owns and operates the system itself. Then managed service, where a provider runs it on the airport's behalf. Then the in-country services entity, a model large airport groups in Asia have used for years, where a local company owns the shared layers and contracts vendors beneath it. At the far end the global vendor platform with in-country hosting and local support tiers.

Each point on that spectrum has an answer on licence, hosting, data, support, integration risk and exit. A programme that has not written those six answers down for each candidate model has not chosen a model; it has chosen a label.

What moves a global vendor

Two things move a global vendor to change how it deploys: a documented regulatory requirement, and a mandate from an operator or group large enough to matter. A single airport's preference does neither. It will get a polite response and the standard deployment. A national programme that wants vendors on its platform by choice rather than by mandate has to make the platform the quicker, lower-risk route to a compliant deployment, with the local layers already in place. That is a product proposition, and it has to be built before it can be sold.

How a system got there governs how it can move

A system deployed under a live contract and not up for renewal moves on its existing terms or at end of support, not before. A new deployment, an out-of-support replacement or a scheduled refresh can be placed on a new platform directly. The same system can be an immediate opportunity at one airport and a multi-year proposition at the next, so the sequence follows the contract calendar, not the strategy document. Greenfield procurements are the exception and the prize: they can be designed onto a shared in-country platform from the first drawing.

Our position

Localise by layer, design for the group, and let the contract calendar set the sequence. Sovereignty and regulation set the floor and give a national programme its mandate; the local share every vendor duplicates is where the money is, and it appears at group scale. Vendors move for documented requirements and for mandates large enough to matter, so establish

both before asking anyone to change how they deploy. A plan built that way survives its first year, and its first renewal.

Written by Drew Sundin

Founder & Principal Consultant, Going Systems