DISTRIBUTED INTELLIGENCE

Put intelligence where the building needs it

How rules, analytics, inference and decision-making are distributed between the building edge and the cloud — what decides where a workload belongs, how models are governed once deployed, and why intelligence expanding does not mean control authority expanding with it.

Traditional IoT architecture assumes a simple path: device to cloud, cloud to application, application back to device. That works when the objective is monitoring, reporting or centralized analysis. Buildings are less forgiving. They contain physical systems that operate continuously — temperature moves, occupancy changes, equipment cycles, valves actuate, fans start and stop, network connections fail — and a room becomes uncomfortable while software is still waiting on a round trip to a distant system.

Keeping everything local is not the answer either. The cloud is better at portfolio analysis, long-horizon trend detection, central administration, large-scale processing, software and model lifecycle, cross-building comparison, and any workload that gains nothing from running beside the equipment.

So the architectural question is not edge or cloud. It is which intelligence belongs where, and what governs it once it is there.

SCOPE

This explainer owns the edge/cloud intelligence architecture: workload placement, local processing and inference, model governance, and how intelligence expands without rebuilding the building integration. Physical-authority mechanics belong to Protected Control & Resilience; resource isolation and software lifecycle to Managed Edge; normalization and the data permission model to Connectivity & Integrations. Capability availability varies by release and configuration — see Current availability at the end.

FIRST, A DISTINCTION

Distributed intelligence does not mean AI everywhere.

Intelligence in a building is a spectrum, and most of the useful work at the low end is not machine learning at all.

Deterministic local logic	Schedules · thresholds · state machines · equipment coordination · basic automation.

Local analytics	Trend evaluation · state comparison · derived conditions · simple fault logic · sensor correlation.
Model-based inference	Classification · prediction · anomaly detection · computer vision · more complex sensor fusion.
Distributed intelligence	All of the above, plus cloud analytics and centralized models, operating as one system.

Edge AI is one component of this, not a synonym for it. A great deal of what makes a building operate well is deterministic logic executing reliably in the right place — and an architecture that treats every problem as a model problem will run inference to answer questions a threshold already answers.

CONTEXT COMES FIRST

More compute does not fix bad building data.

Register 40217 = 724 → RTU-3 · Supply Air Temperature · 72.4 °F · serving Zone 3

A model working from the first has a number. A model working from the second has building context. Add zone occupancy, room temperature, HVAC operating mode, electrical demand, outside conditions and equipment state, and the same model reasons about an operating condition instead of an isolated signal.

A cloud platform can have enormous compute and still not know which rooftop unit serves which room, which meter corresponds to which equipment, whether a space is occupied, whether a controller is in alarm, whether a setpoint is under operator control, or whether a commanded action actually produced a physical response. None of that is inferable from raw telemetry; it has to be established through building integration, which is what Connectivity & Integrations describes.

The value of the edge is therefore not that it contains a processor. It is that the processor is attached to a connected and contextualized physical environment.

WORKLOAD PLACEMENT

Six questions decide where a workload runs.

Placement is the central decision in this architecture, and it is answerable rather than philosophical. For any given workload:

Model and memory size	Does it fit inside the deployed hardware envelope, alongside everything else the Gateway is already doing?
Input data volume and rate	Is the input stream cheap to move upstream, or expensive enough that processing it locally is the point?
Response-time requirement	Does anything physical depend on the answer arriving quickly, or is it advisory?
Concurrency	How many instances run at once, across how many zones or devices?

Thermal and power envelope	Can the deployed hardware sustain the load, not just peak to it?
Behaviour when disconnected	Must this keep working when the WAN is down, or is pausing acceptable?

Those questions have answers before a line of code is deployed, and they generally decide the matter without appeal to architecture preference.

What the edge is good at

Low response time, because conditions are evaluated beside the equipment producing them. Continuity, because supported local functions carry on when the WAN does not. Local building context, because protocol services and contextualized equipment state are already there. Data reduction, because raw observation becomes derived condition before anything moves upstream. Privacy, because an operational state can be derived locally instead of raw evidence being shared. And physical verification, because whether an authorized action produced the expected equipment response is observable locally and nowhere else.

What the cloud is good at

Portfolio analytics comparing buildings, equipment classes, climates and operating patterns across sites. Long-horizon analysis over weeks, months or years of operating history. Model and software lifecycle — distributing approved versions to the fleet. Cross-site learning that is invisible from inside one building. Compute-intensive work beyond any practical local envelope. And central administration of users, portfolios, applications and policy.

The cloud sees breadth; the edge sees immediacy. Most genuinely useful workflows use both, and the interesting architecture is in how they hand off rather than in which one wins.

HYBRID IN PRACTICE

One system, work happening where it makes sense.

Take a comfort and HVAC workflow. At the edge, the platform already holds room temperature, occupancy state, equipment mode, setpoint, fan state and whether the unit is responding. A local rule or model flags a condition — room occupied, temperature rising, cooling requested, equipment running, and yet the space is not approaching setpoint.

Upstream, the same equipment is compared against other rooms, against its own operating history, against similar units across the portfolio, against maintenance records and longer-term energy performance. What comes back is an alert, a diagnostic recommendation, an updated model, an operating policy, or an authorized request for action. It remains one system; the work simply occurs where each part belongs.

A condition often has more than one source

Few useful operating conditions are represented by a single sensor. Occupancy may be informed by room sensors, people counting, access events, schedule context or environmental change. Equipment

performance combines commanded state, actual state, temperature, power, runtime and ambient conditions.

Occupancy high · CO_2 rising · ventilation inactive	Investigate the ventilation condition.
Power elevated · space unoccupied	Possible unnecessary operation.
HVAC running · room not approaching setpoint	Possible performance issue.
Water flow · environmental change · no expected use	Possible abnormal water condition.

None of those four examples requires AI, and that is the point. Once a building has a shared contextual model, signals are evaluated together instead of staying isolated inside separate applications — and that is worth more, more often, than any individual model.

WHAT ACTUALLY RUNS

A deliberately narrow production set.

Abstraction is easy in this subject, so it is worth being specific about the boundary without reproducing a second capability inventory. Selected intelligence capabilities are already operating in qualified deployments today. AI & Connected Intelligence contains the current capability inventory, maturity and scope notes; this explainer focuses on where those workloads run and how they are governed.

The current set is intentionally narrow, and the reason is a placement argument rather than a product one. The useful workloads are those where building context materially improves the result, where the required inputs are available at the edge, and where the output has a defined operating consumer - rather than demonstrations of what a model can be made to say about a building.

> **Accuracy claims are tied to validated sensing configurations and site admission checks — not to universal marketing promises. A capability that depends on a sensing arrangement the site does not have is not a capability that site has.**

MODEL GOVERNANCE

Models are governed artifacts, not files.

The platform model treats deployed models as governed software artifacts rather than arbitrary files. As the Edge AI capability set expands, that governance architecture encompasses model registration and approval, placement against available hardware, workload isolation, capability abstraction and post-deployment monitoring. Specific enforcement mechanisms vary by hardware and software release. A model becomes eligible to run only through the controls supported by the deployed configuration, rather than simply appearing on a device.

Registry admission	Owner, version, license, signature, runtime requirements, permitted uses, risk class, residency and approval status are checked before download — so an arbitrary or unapproved model does not reach a gateway.

Hardware placement	CPU, memory, accelerator, GPU memory, storage, network, thermal state, power budget and utilization determine placement. A model is not placed on hardware that cannot safely run it.
Isolated execution	Models run inside their own identities, namespaces, storage boundaries and resource allocations, which contains noisy neighbours and separates tenant, partner and building workloads.
Abstraction layer	Applications call stable building capabilities rather than model-specific providers or endpoints.
Monitoring and rollback	Latency, error rate, loading, drift and resource use are monitored; a model version can be suspended or rolled back after deployment.

Why the abstraction layer matters commercially

Applications do not hard-code model endpoints. They consume AI capabilities through a model abstraction layer that routes each request to an approved local model or another policy-permitted provider. The practical consequence is that a customer can change model strategy — a local model, an open-source model, one they supply themselves, or a cloud provider where policy permits — without rewriting the applications built on top. Model choice and local-only data policy become configuration rather than a rebuild.

It also means a model is accountable after it ships. Drift and error rate are not discovered when someone complains about comfort; they are monitored, and a version that degrades can be rolled back the way any other deployed software would be. Managed Edge covers the distribution and lifecycle mechanics.

AUTHORITY

Inference is not control, and local is not autonomous.

An inference workload concludes that a setpoint should drop. An optimization routine concludes that equipment should start earlier. An anomaly model concludes that something is wrong. A vision model concludes that a space is occupied. Those are outputs. None of them is a physical command.

> **Building state → rule, analytic or model → proposed action → authority and limits → control service → field equipment**

The probabilistic intelligence layer is separated from the deterministic path that acts on equipment, which is what lets intelligence become more sophisticated without every model acquiring access to the systems it observes.

Location and authority are also separate questions, and conflating them is a common error. A workload running locally may be observation-only, analytical, advisory, or permitted to request bounded action. A cloud workload may hold broad analytical visibility and no equipment authority at all. Distributed intelligence determines where computation happens. It does not determine who is allowed to act.

Established building authority is unaffected by any of it. Equipment safeties remain in place. A manual operator action, a native equipment protection or a supervisory command is not treated as one more model prediction competing for priority — control policy determines which source is authoritative and what an intelligent workload may request. Protected Control & Resilience covers precedence, override and recovery in full.

PRIORITY

Building operations come before compute.

A building edge may eventually host protocol services, local automation, energy optimization, equipment monitoring, analytics, inference and partner applications. Those workloads are not of equal operational importance, and connectivity, control services and required operating functions stay protected from less critical compute activity.

> **Compute is available to intelligence inside an operating envelope — not at the expense of the building.**

Managed Edge owns the resource-management and workload-isolation mechanics. The consequence owned here is that a building gains intelligence without building operations becoming a best-effort workload competing with an inference job.

DATA MOVEMENT

Not all data needs to move, and some of it should not.

A cloud-first architecture starts by asking how to get the data upstream. A distributed architecture asks what the upstream system actually needs. A local workload turns many raw samples into an equipment state, an anomaly, an occupancy condition, a trend or an operational event, and the upstream system consumes that derived information rather than every underlying observation.

That is a bandwidth argument, and it is the weaker of the two. The stronger one is privacy. Occupancy sensing, people counting and vision workloads produce data about people, and the difference between sending raw radar returns or video frames to a cloud service and sending an occupied-or-vacant state is not a difference in efficiency — it is a difference in what leaves the building at all. Deriving the operational state locally means the raw evidence never has to travel, which is a materially stronger position than promising to handle it carefully once it arrives.

Connectivity & Integrations defines the integration boundary and the purpose-binding model that governs what a consumer receives. Placement determines what exists to be shared in the first place.

WHEN CONNECTIVITY CHANGES

Lose the functions that need the cloud, not every function.

A cloud-dependent workload cannot continue normally when the cloud is unreachable. A building-local workload often can. The architecture is built so that different functions degrade differently rather than the whole building being treated as either online or offline.

Local schedule	Continues while its required information and authority remain available.
Locally deployed inference	Continues while its inputs, runtime and authorization remain available.
Portfolio optimization	Pauses, since it depends on data from multiple buildings.
Model update	Waits until communications return.

Note the second row carefully: a local model keeps running while its authorization holds, not indefinitely. Autonomy is bounded by policy rather than by the presence of a processor, and Protected Control & Resilience describes how that authority contracts as isolation continues.

GROWTH

The physical integration should outlive the compute generation.

A building may begin with a Universal Gateway performing schedules, thresholds, local automation, device coordination, equipment-state evaluation and local data processing. An Edge AI configuration extends the same architecture with hardware-accelerated inference and model serving.

What matters is that the building does not have to be rediscovered because compute improved. What equipment exists, which space it serves, which points represent which capabilities, which applications may access them and what physical authority exists — all of that persists while the compute envelope changes above it. That is a fundamentally different upgrade model from installing a separate AI appliance beside the BAS every time a new workload appears.

Nor does every building need the same hardware. The six placement questions decide the configuration as much as they decide the location, and current product specifications govern the processor, accelerator, memory and supported workload envelope for any given deployment.

More compute is not the strategy

A general-purpose edge computer runs sophisticated software. That alone does not make it a building-intelligence platform — it still needs device connectivity, building protocols, semantic context, equipment relationships, deployment lifecycle, operating authority, and a path from digital output to physical response. The value of more capable edge hardware is not more TOPS or larger models. It is that capable computation sits inside infrastructure that already reaches and understands the physical building.

HOW THIS DIFFERS

Four architectures, each good at what it was built for.

	Cloud-first IoT / AI	Traditional building controller	General edge compute	LEXI distributed intelligence
Primary strength	Central analytics and scale	Local deterministic building control	Flexible local compute	Building-aware edge and cloud intelligence
Building protocols	Usually another layer	Core strength	Integration-specific	Part of the shared building edge
Building context	Must be supplied upstream	BAS and station-oriented	Usually application-created	Shared normalized building context
Local operation	Often cloud-dependent	Strong	Software-dependent	Rules, automation and local intelligence
AI inference	Primarily centralized	Usually outside the architecture	Strong compute potential	Local or cloud by workload
Model governance	Platform-dependent	Not applicable	Left to the operator	Registry, placement, monitoring, rollback
Portfolio analysis	Strong	Usually secondary	Platform-dependent	Cloud layer over common building context
Physical authority	Often outside the control architecture	Embedded in controller logic	Not inherently building-aware	Separated from governed control
Upgrade path	More cloud compute	Controller replacement	Add compute and integrations	Expand compute over existing context

The distinction is the combination, not any single row.

WHAT THIS MEANS

For the people who deploy, own and build on it.

Building owners	Infrastructure that delivers value now and supports additional intelligence later, without every future AI workload having to be defined on day one.
Systems integrators	Field engineering becomes reusable — new analytics and AI applications operate over an existing building model instead of requiring another integration project.
AI and technology partners	Focus on the model rather than rebuilding protocols, gateway hardware, sensor connectivity, semantic normalization, remote management and control boundaries — and ship into a runtime where a model version can be monitored and rolled back like any other software.

Telecom and managed-service providers	Extend intelligence from centralized infrastructure into the building over the same managed edge used for connectivity and local execution.

CURRENT AVAILABILITY

What runs today.

> **The current building-edge architecture supports local execution of building functions including schedules, thresholds, automation rules, device coordination, equipment-state evaluation, local data processing and selected optimization. Universal Gateway configurations provide the connectivity and local execution foundation; Edge AI configurations extend the same architecture with hardware-accelerated local inference and model serving.The current intelligence capability inventory, maturity and scope notes are maintained in AI & Connected Intelligence rather than duplicated here. Which workloads execute locally depends on hardware configuration, available compute, software release, application support, connected equipment and deployment requirements. Broader distributed workload orchestration is delivered progressively as the Managed Edge and Edge AI platform develop. The architectural claim is not that every AI workload already runs locally - it is that the building is structured so the right workloads can move closer to the physical environment as the compute envelope expands.**

THE CORE IDEA

The cloud should not do what the building does better itself.

And the building should not do what the cloud does better centrally. Distributed intelligence is the architecture between those two positions — and the result is not an intelligent gateway or an intelligent cloud, but a platform in which intelligence can be placed where it creates the most operational value.

> **Connect the physical building. Give its data meaning. Run immediate and contextual functions locally. Use the cloud for scale and breadth. Add capable edge inference where the workload benefits from being inside the building. Keep intelligence separate from unrestricted physical authority.**

Local-processing, inference and workload capabilities vary by Gateway family, Edge AI configuration, software release, application and deployment. Physical actions remain subject to configured permissions, equipment limits and operating policies.

Related · Universal Gateway · Managed Edge · Connectivity & Integrations · Protected Control & Resilience · AI & Connected Intelligence · White-Label LEXI Cloud