ICT in Agriculture – Technical Systems for Precision and Resilience

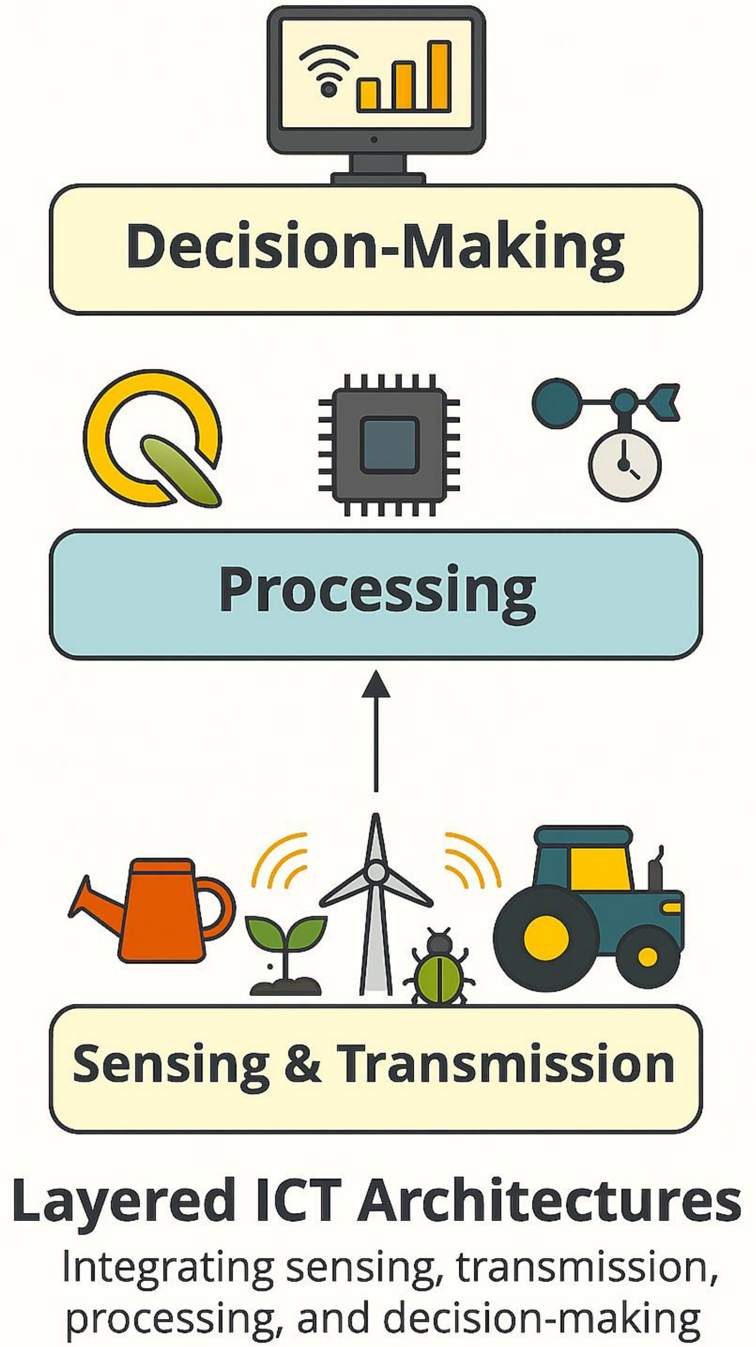
Lesson 05

Overview of ICT Architectures in Agriculture

- Precision agriculture systems rely on layered ICT architectures integrating sensing, transmission, processing, and decision-making.
- Each layer must be optimized for rural constraints: power, terrain, and bandwidth.

Technical Note:

- Layered architectures enable modular upgrades and fault isolation, improving system resilience.
- Rural deployments require adaptive topologies that tolerate intermittent connectivity and variable energy availability.



Peru's Unique Challenges for Smart Agriculture

1. Extreme Landscape

- Andes Mountains & Amazon: The rugged terrain makes it tough to build internet and power lines.
- The Fix: Smart agriculture uses technology that doesn't need much power or a strong signal, like satellite communication and low-power networks.

2. Adapting to the Land

- Many Native Crops: One-size-fits-all technology won't work. Solutions must adapt to the variety of Peruvian crops.
- Ancient Wisdom:
  - Farmers have deep knowledge of their land.
  - Technology acts as a new tool to enhance, not replace, traditional farming methods.

People and Culture in the Mix

1. Economic Barriers

- High Costs: New tech can be expensive for small-scale farmers.
- Our Goal: To succeed, solutions must be affordable and easy to use.

2. Community First

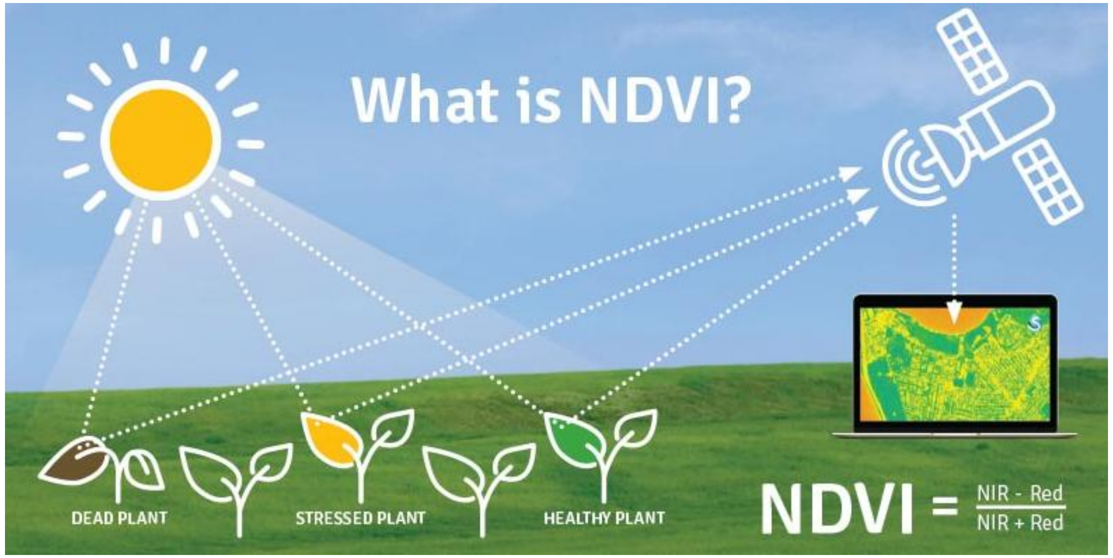
- Learning Curve: Not all farmers are tech-savvy.
- The Approach: Success comes from building trust and offering hands-on training and community support.
- The technology must fit into their lives, not disrupt them.

Sensor Modalities and Deployment Strategies

- Agricultural sensors include soil moisture probes, temperature loggers, pH meters, and NDVI imaging units.
- Deployment must consider crop type, terrain gradient, and microclimatic zones for optimal data granularity.

Technical Note:

- Sensor placement affects data fidelity and predictive accuracy in agronomic models.
- Calibration routines and sampling intervals must be tailored to crop cycles and environmental volatility.



Normalized Difference Vegetation Index

Transmission Protocols for Rural Connectivity

- There is a set of them who offer low-power, long-range transmission suited to rural zones.
- Protocol selection depends on
  - terrain profile,
  - node density, and
  - interference tolerance.

Technical Note:

- Link budget modeling ensures reliable transmission under foliage, elevation, and weather constraints.
- Protocol stack selection must balance throughput, latency, and energy consumption.

Let’s visit some of them

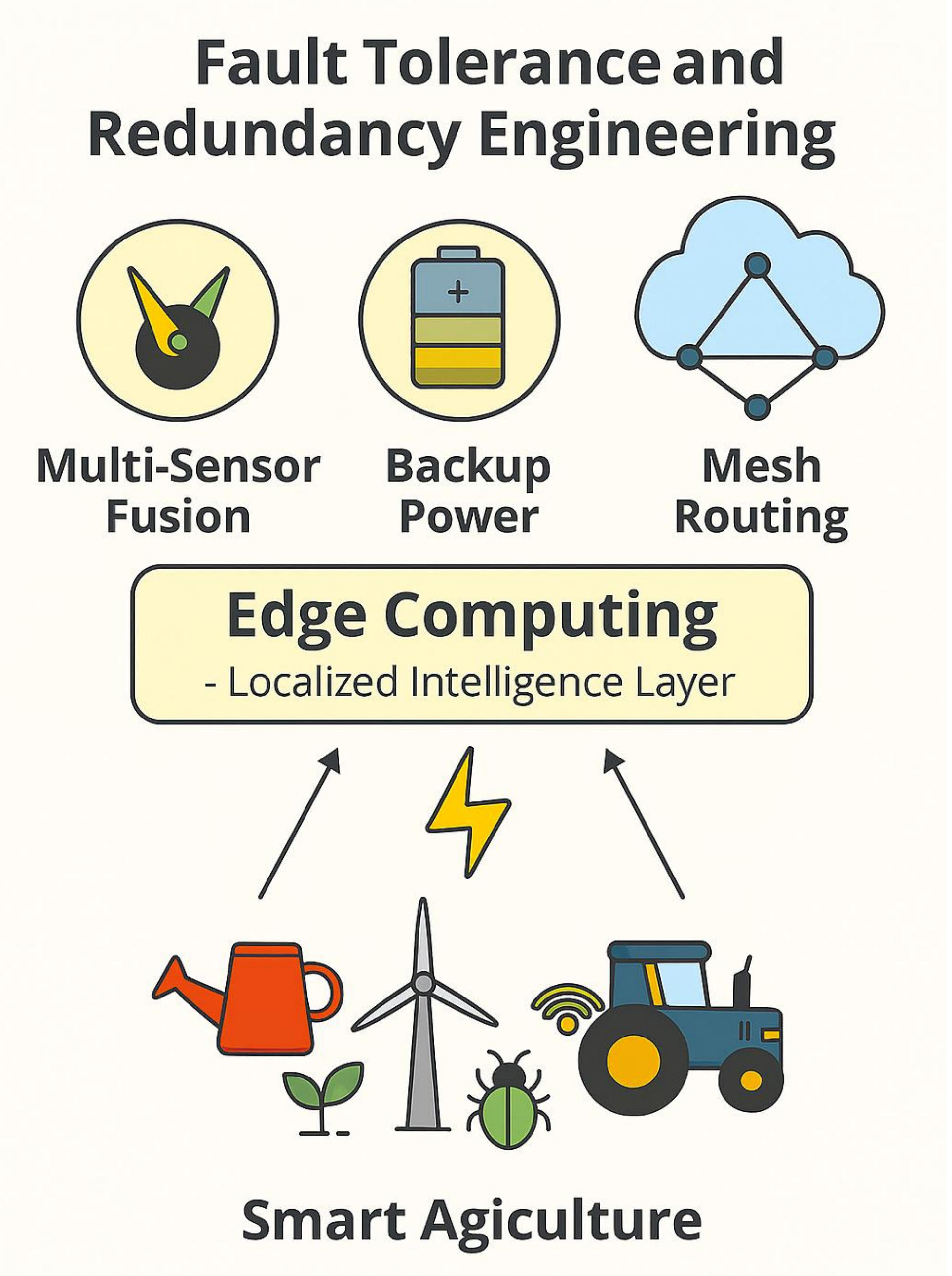
Protocol Name	Description	Ideal Use Scenario
LoRaWAN	Long-range, low-power protocol for sparse deployments	Wide-area sensor networks in large rural farms
Zigbee	Short-range mesh for dense sensor networks	Greenhouse automation and intra-field coordination
TVWS	Uses unused TV spectrum for long-distance links	Connectivity backbone in hilly or forested zones
NB-IoT	Cellular-based, low-power, deep coverage	Soil and climate sensors with periodic updates
SigFox	Ultra-narrowband, minimal payload, long range	Remote telemetry with low data requirements
Wi-Fi HaLow	Sub-GHz Wi-Fi with extended range and low power	Farm-wide sensor arrays with moderate data needs
BLE	Short-range, energy-efficient for wearables	Livestock tracking and portable diagnostics
RFID	Tag-based tracking with proximity readers	Asset and inventory management
LTE-M	Mobile-friendly cellular protocol with voice/data	Telemetry from mobile agricultural machinery
Satellite IoT	Global coverage for ultra-remote deployments	Climate monitoring in isolated farming regions

Fault Tolerance and Redundancy Engineering

- Agricultural ICT systems must anticipate sensor failure, power loss, and transmission dropouts.
- Redundancy strategies include multi-sensor fusion, backup power, and mesh routing protocols.

Technical Note:

- Fault-tolerant designs reduce downtime and protect yield-critical operations
- Redundancy increases system complexity but is essential for high-reliability deployments in volatile rural environments.

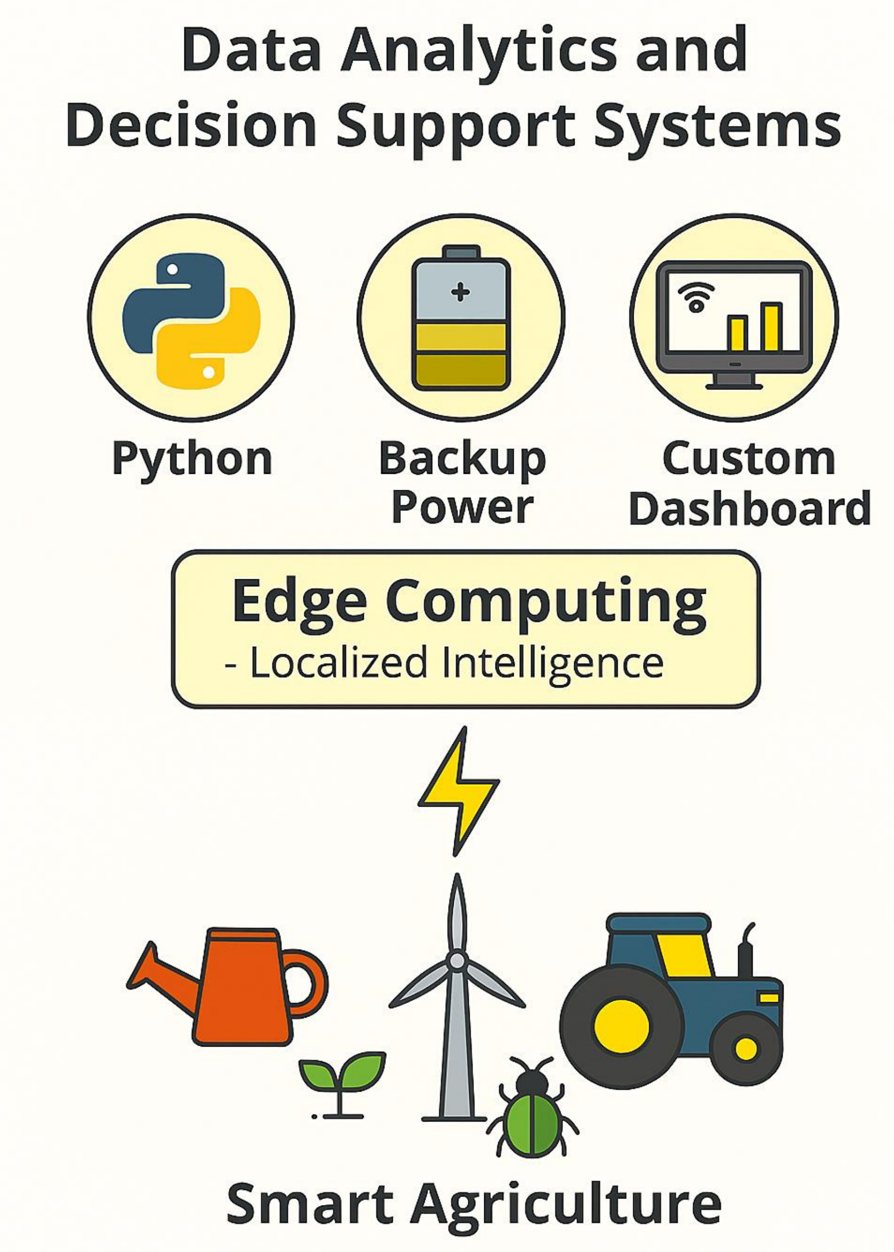


Data Analytics and Decision Support Systems

- Collected data feeds into models for irrigation scheduling, pest detection, and yield forecasting.
- Tools include Python (Pandas, Scikit-learn), QGIS, and custom dashboards.

Technical Note:

- Analytics pipelines must handle noisy, incomplete data from heterogeneous sources.
- Decision support systems must translate outputs into actionable insights for low-literacy users.

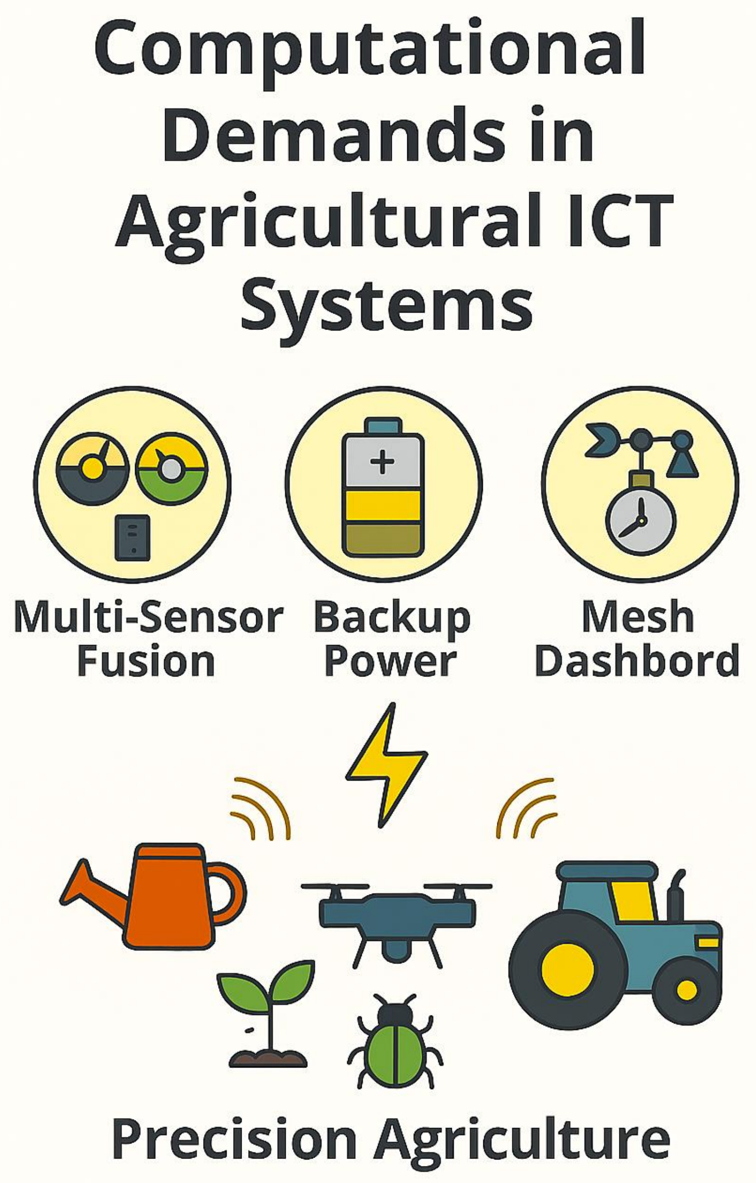


Computational Demands in Agricultural ICT Systems

- Precision agriculture generates continuous data streams from sensors, drones, and weather stations.
- Processing needs vary by task: real-time alerts, historical trend analysis, and predictive modeling.

Technical Note:

- Real-time tasks like irrigation control require low-latency processing at the edge.
- Long-term analytics benefit from cloud scalability and integration with external datasets.

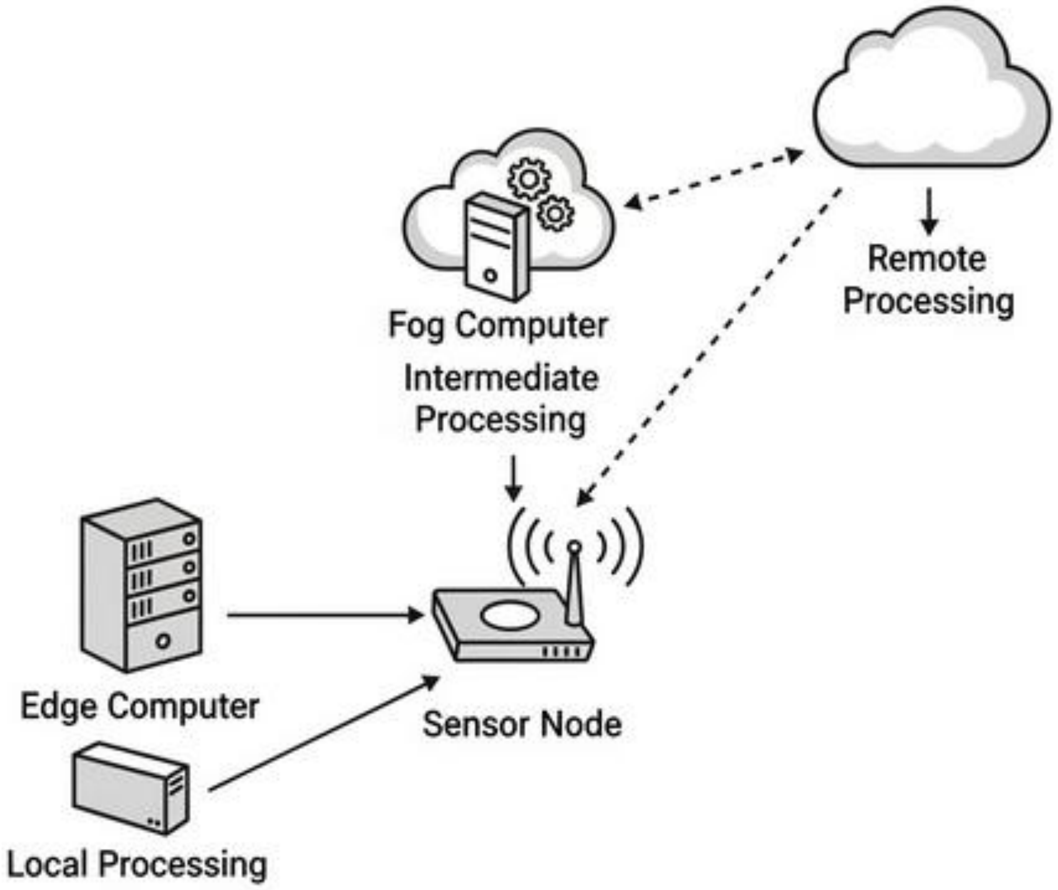


Edge vs. Cloud Processing Models

- Edge computing reduces latency and bandwidth usage by processing data locally.
- Cloud models offer scalability and advanced analytics but depend on stable uplinks. Hybrid models optimize trade-offs.

Technical Note:

- Edge nodes must support real-time analytics and local decision logic under constrained resources.
- Cloud integration enables historical trend analysis and remote system diagnostics.

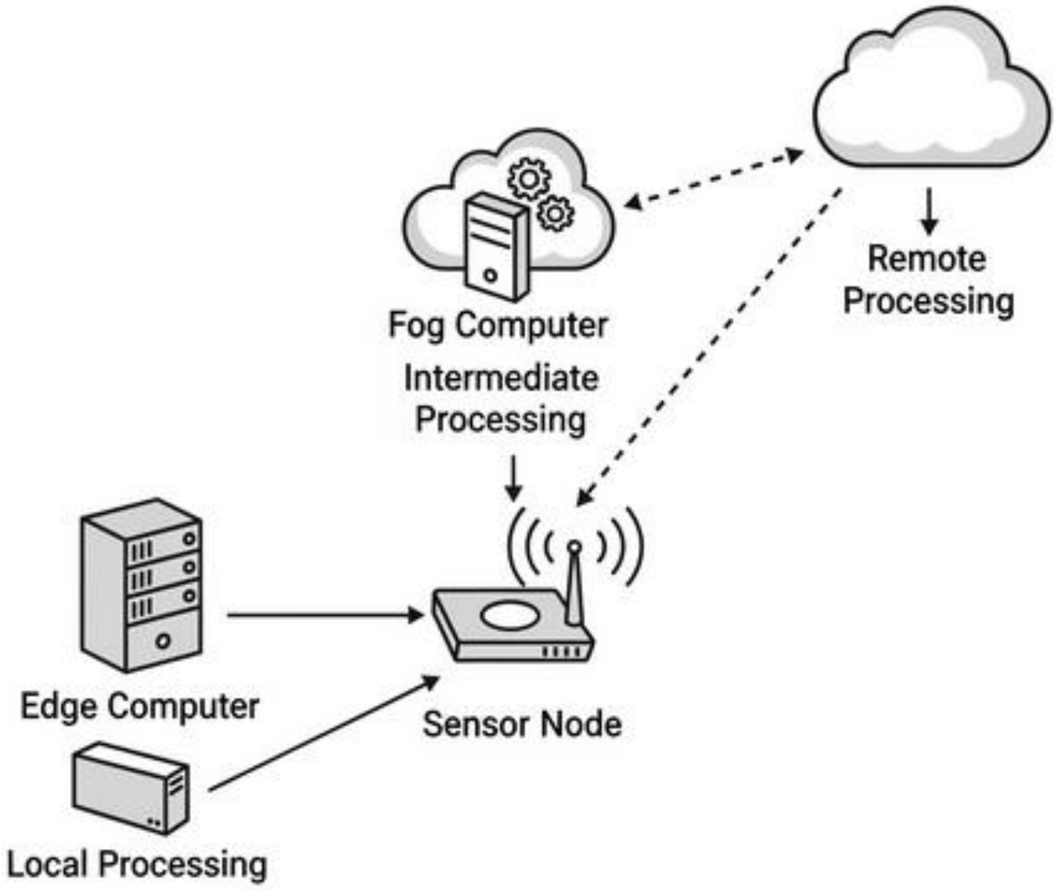


Edge vs. Fog Processing Models

- Edge computing operates directly at the data source, enabling immediate response and minimal latency.
- Fog computing introduces an intermediate layer that aggregates data from multiple edge nodes, offering regional coordination and preprocessing.

Technical Note:

- Edge nodes are ideal for isolated, real-time control tasks with limited compute.
- Fog nodes extend system intelligence by managing distributed loads and routing across multiple edge clusters.

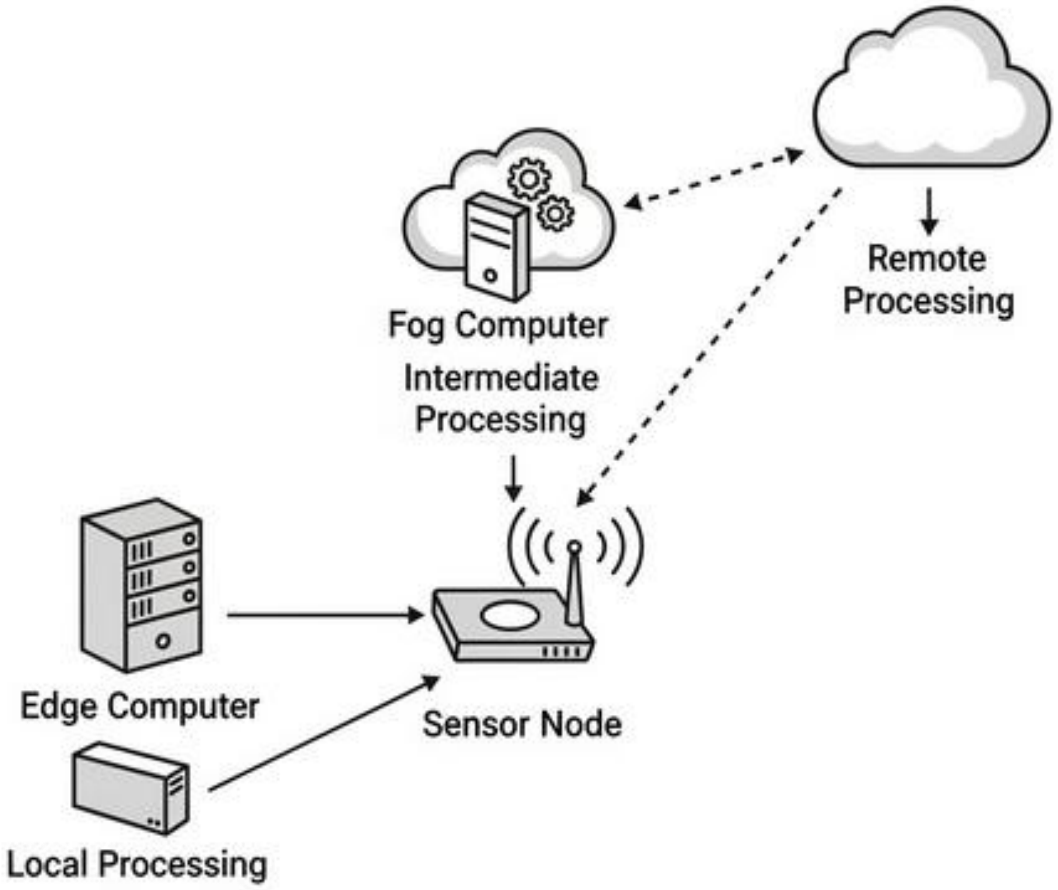


Fog vs. Cloud Processing Models

- Fog computing provides localized analytics and buffering, reducing dependence on continuous uplinks.
- Cloud computing centralizes storage and advanced processing, supporting large-scale modeling and cross-regional insights.

Technical Note:

- Fog systems mitigate uplink instability and enable near-real-time decisions.
- Cloud platforms excel in historical data mining, model training, and integration with external datasets.

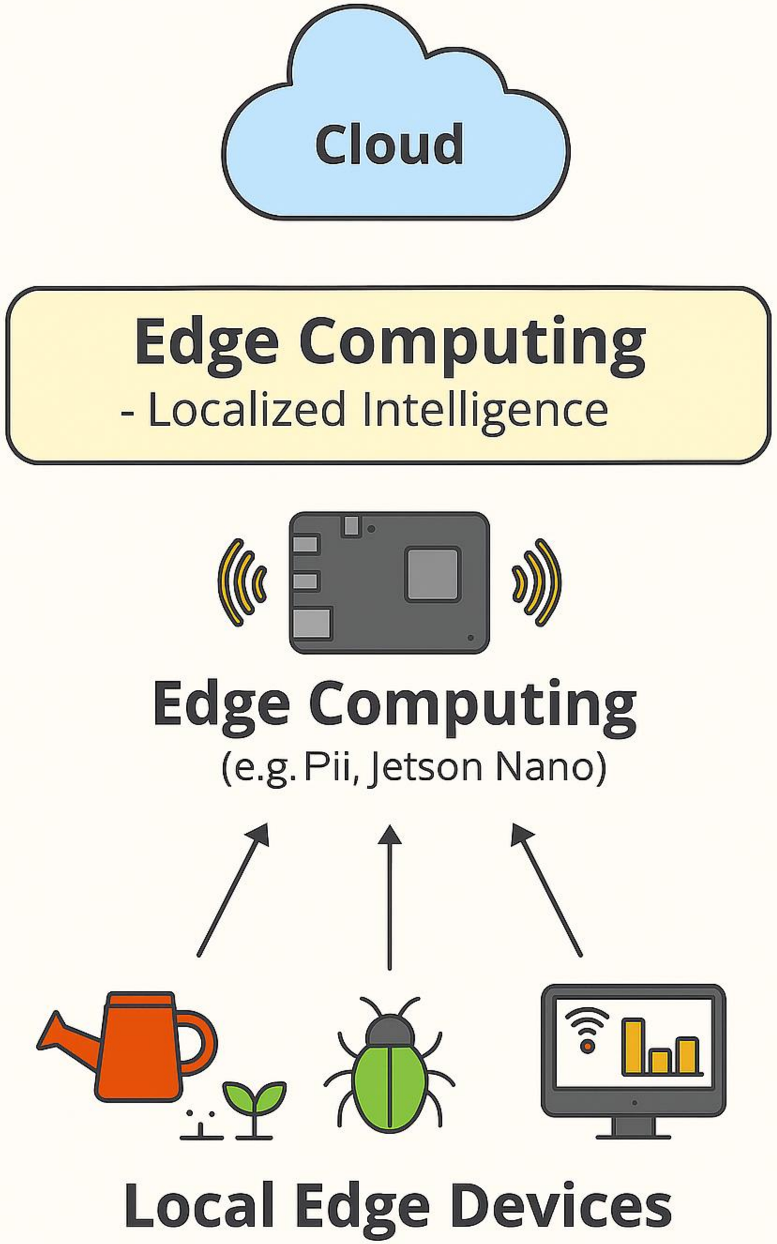


Edge Computing – Localized Intelligence

- Edge nodes (e.g., Raspberry Pi, Jetson Nano) process data near the source.
- Ideal for latency-sensitive tasks: irrigation triggers, pest alerts, and offline dashboards.

Technical Note:

- Edge computing reduces bandwidth usage and ensures autonomy during uplink failures.
- Resource constraints demand efficient code and lightweight models for embedded deployment.

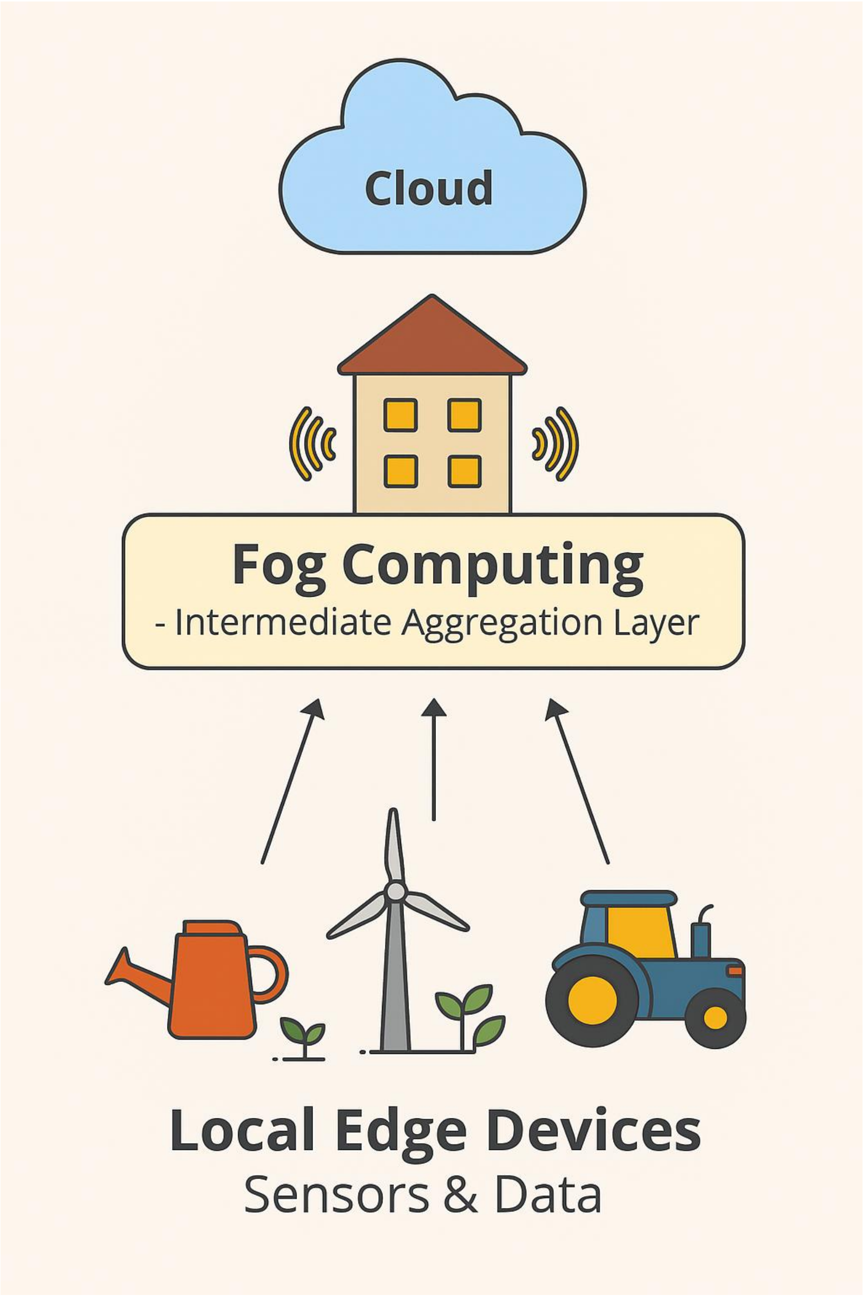


Fog Computing – Intermediate Aggregation Layer

- Fog nodes aggregate data from multiple edge devices, perform preprocessing, and manage routing.
- Useful in distributed farms or cooperatives with semi-centralized infrastructure.

Technical Note:

- Fog computing balances latency and scalability, enabling regional analytics and load distribution.
- It supports modular expansion without full reliance on cloud infrastructure.



Cloud Computing – Scalable Analytics and Storage

- Cloud platforms (e.g., AWS, Azure, GCP) handle large-scale data storage, ML model training, and remote access.
- Suitable for trend analysis, policy dashboards, and multi-region coordination.

Technical Note:

- Cloud systems offer elastic resources but depend on stable connectivity and cost management.
- Integration with edge/fog layers ensures continuity during network disruptions.

Comparative Scenarios – Edge vs. Fog vs. Cloud

Scenario	Optimal Layer	Justification
Real-time irrigation	Edge	Low latency, local autonomy
Regional crop health mapping	Fog	Aggregation, preprocessing
National yield forecasting	Cloud	Scalability, external data fusion
SMS alert system	Edge/Fog	Local trigger, regional dispatch
Drone image analysis	Cloud	High compute, storage needs

Technical Note:

- Layer selection must reflect task criticality, connectivity profile, and compute availability.
- Hybrid architectures allow dynamic task allocation across layers for optimal performance.

Resource Planning and Deployment Models

- Define compute, memory, and power budgets per layer.

Technical Note:

- Resource planning ensures system stability and cost-efficiency across deployment phases.
- Monitoring tools must track usage, detect bottlenecks, and trigger reallocation when thresholds are breached.

Edge nodes	**≤2GB RAM, ≤10W power.**
Fog servers	multi-core CPUs, UPS backup.
Cloud	pay-as-you-go with autoscaling

Economic Viability of Smart Agriculture Systems

- Smart agriculture in rural zones must balance technical ambition with economic feasibility.
- Initial investments include sensors, transmission modules, edge devices, and energy systems.
- Long-term viability depends on minimizing operational costs and maximizing yield impact.

Technical Note:

- Economic modeling must include lifecycle costs and local labor dynamics to ensure realistic projections.
- ROI should be benchmarked against traditional practices to validate technology adoption.

Key Factors:

CAPEX	**Hardware, installation, training**
OPEX	Maintenance, energy, data plans
ROI	Yield increase, water savings, reduced input waste
Risk	Seasonal variability, equipment failure, market volatility

Cost Modeling and Sustainability Planning

CAPEX

includes sensors, transmission modules, and edge devices.

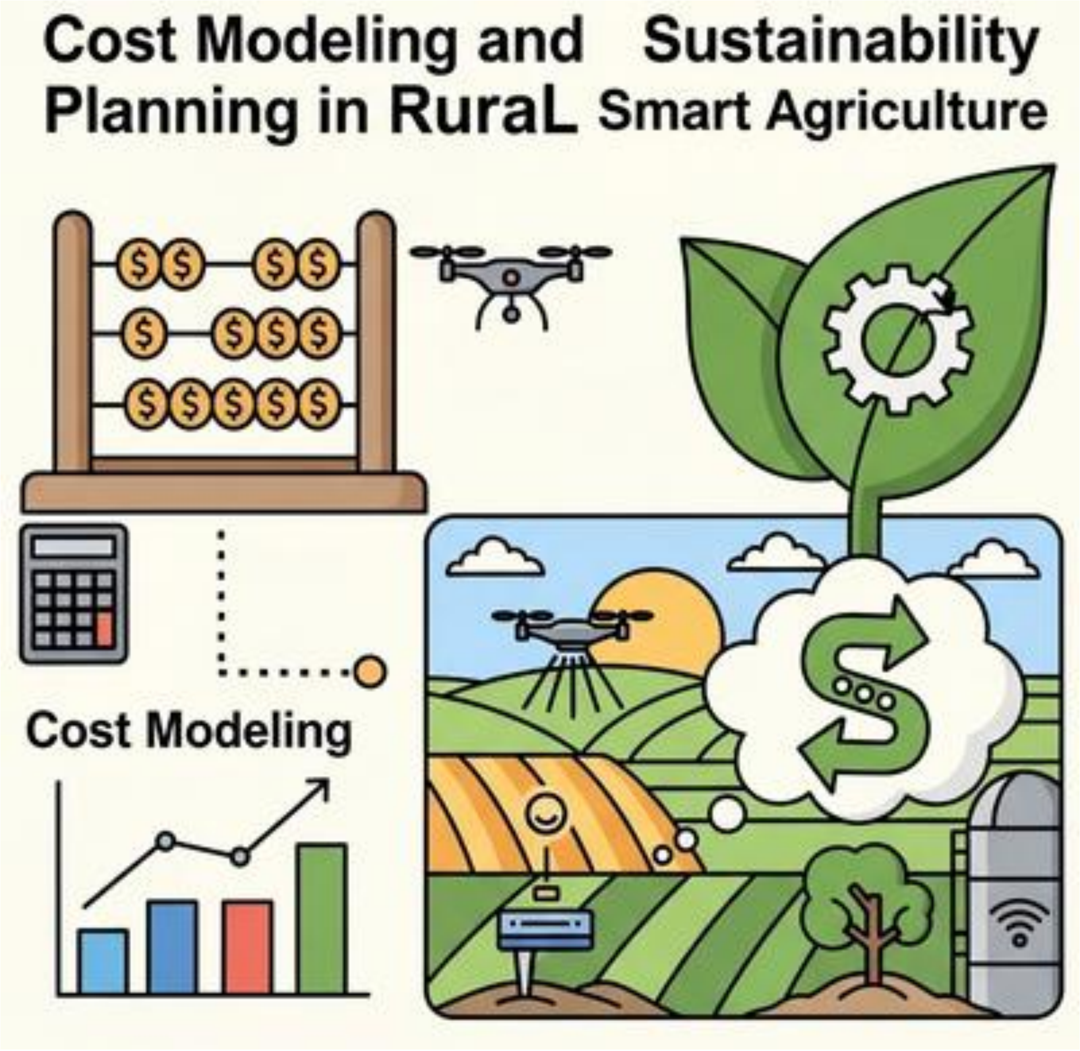
OPEX

covers maintenance, energy, and data plans.

Sustainability depends on local capacity, training, and institutional support.

Technical Note:

- Cost models must account for lifecycle expenses and scalability potential.
- Community ownership and training programs enhance long-term viability and reduce system abandonment.



Financing Models and Stakeholder Ecosystems

- Rural smart agriculture requires blended financing strategies to overcome capital barriers.
- Models include public subsidies, cooperative ownership, microfinance, and CSR-backed deployments. Stakeholder alignment is critical for sustainability.

Key Models:

- Government grants for SDG-aligned infrastructure
- Farmer cooperatives pooling resources for shared ICT kits
- NGO-led deployments with training and maintenance support
- Private sector partnerships offering bundled services

Technical Note:

- Financing must be modular and scalable, allowing phased deployment and cost recovery.
- Stakeholder mapping ensures accountability and aligns incentives across technical, economic, and social dimensions.

Financing Models and Stakeholder Ecosystems in Rural Smart Agriculture

SDG Mapping and Impact Metrics

- Each system component maps to SDG targets:
  - sensors (SDG 2),
  - transmission (SDG 9),
  - analytics (SDG 13).
- Impact metrics include
  - yield increase,
  - water savings, and
  - farmer income uplift.

Technical Note:

- SDG traceability enables institutional reporting and funding alignment.
- Technical metrics must be translated into development indicators for policy and stakeholder engagement.