Farm Controller Layer (Local / Federated)

See overview: System Architecture Overview

The **Farm Controller Layer** is the on-site AOFS controller that provides **local supervision, configuration, and federation**. It sits between the Field Controller (authoritative safety layer) and HQ / Federated Controllers.

This layer is **offline-first**, **federation-capable**, and **authoritative for non-critical decisions**.

1. Responsibilities

The Farm Controller Layer **must**:

- Aggregate telemetry from Field Controllers:
  1. Soil moisture sensors
  2. Tank levels
  3. Flow meters
  4. Power usage
  5. Optical monitoring data
- Provide a **local operator interface** for monitoring and configuration.
- Validate all operator requests against Field Controller safety rules.
- Manage irrigation schedules and configuration updates locally.
- Enable **push/pull federation** with other Farm Controllers or HQ Controllers.
- Log all local actions, operator inputs, and synchronization events for auditability.

—

2. Offline Operation

1. **Full local autonomy:**

- Must operate independently of network connectivity.
- Irrigation schedules, sensor monitoring, and fail-safe enforcement continue uninterrupted.

2. **Local operator interface:**

- Operators can view data and update configurations.
- Only non-critical adjustments are applied locally; safety-critical overrides are blocked by Field Controller logic.

3. **Local data storage:**

- Telemetry and logs are persisted locally.
- Data is queued for synchronization when connectivity is restored.

—

3. Federation Model

Farm Controllers support a **Git-like push/pull model**:

- **Pull:**
  1. Farm Controller can pull configuration, software, or firmware updates from HQ or peer controllers.
  2. Pulled changes are validated against local rules and logged.
- **Push:**
  1. Farm Controller can push logs, irrigation events, and audit data to HQ or peer controllers.
  2. Push is queued if connectivity is unavailable.
- **Peer-to-peer:**
  1. Farm Controllers can synchronize directly with other farms for configuration and data exchange.
  2. Conflicts are resolved using deterministic rules (see section 4).

—

4. Conflict Resolution

When multiple controllers modify configurations or schedules:

```
1. **Timestamp precedence:**
   * The most recent authorized change takes effect.
```

```
2. **Operator approval:**
   * In case of conflicting schedule changes, local farm operators must
approve before applying.
```

```
3. **Field Controller enforcement:**
   * Field Controller always vetoes any configuration that violates **safety
constraints**.
```

```
4. **Logging:**
   * All conflict events must be logged with timestamps, operator ID, and
resolution outcome.
```

—

5. Authority Rules

1. **Safety authority:**

- Field Controller is authoritative for safety-critical decisions.
- Farm Controller cannot override irrigation cut-offs, pump safety, or flood prevention rules.

2. **Configuration authority:**

- Farm Controller is authoritative for **non-critical configurations** and local schedules.
- HQ or peer controllers may propose updates, but local rules take precedence.

3. **Audit and compliance:**

- All configuration changes, operator interactions, and sync operations must be logged.
- Logs must be retained locally and transmitted to upstream controllers when possible.

—

6. Human Interface

* The Farm Controller **must provide a full local UI**:

1. Monitoring dashboards for all irrigation zones
2. Configuration of schedules and thresholds
3. Alerts and notifications for operators

* Operator actions are validated against Field Controller safety rules. * The interface supports **manual requests**, but these cannot bypass critical safety decisions.

—

7. Implementation Notes

* Hardware: Industrial-grade single-board computers (e.g., NanoPi, Raspberry Pi). * Communication protocols: LAN, WiFi, or cellular for synchronization. * Data format: Structured, versioned, and compatible with HQ controller ingestion. * Security: All communications must be encrypted; operator authentication is required. * Scalability: Supports multiple Field Controllers per farm; multiple zones; multi-farm federation.

—

8. Compliance Notes

* Any AOFS-compliant Farm Controller **must implement offline operation, federation, logging, and safety validation**. * Failure to preserve Field Controller authority or provide audit logs **invalidates compliance**. * Synchronization conflicts must follow the deterministic resolution rules defined above.

From:
http://wiki.irrigation.afriticgroup.com/ - **Afritic Open Farming Standard**

Permanent link:
http://wiki.irrigation.afriticgroup.com/doku.php?id=architecture:farm_controller:start&rev=1769027381

Last update: **2026/01/21 20:29**