WatchNode Architecture

After the event arrives: verify, dispatch, handle, close — all on the record

① Events arrive ② SOP dispatch ③ Audited close

① Event sources — Sources

It raises no events; it handles what arrives

XMap events — Primary source
A map alarm arrives; clicking it focuses the map and that camera.

Camera analytics — AI

Smart Gateway — Optional

Positioning boundary
- Standard event handling: use XMap
- Layer on only for custom dashboards
- A demo UI today, deployed per case

Sits on top of existing sources; it does not replace XMap.

Alarms in → XMap · analytics

Event type → Fire · intrusion · other

Site context → Map · camera view

② WatchNode — SOC Console

1 **Alarm verification**
Clicking an alarm focuses the map and the camera, so the operator confirms it is real before acting.

2 **SOP dispatch**
One click brings up the checklist for that event type; fire and intrusion follow different procedures.

3 **Audited close**
All four stages are recorded, so who did what and when can be answered afterwards.

How it is deployed Project-based · tailored to the site · not off the shelf

Queue & dispatch → Assigned to an operator

SOP checklist → Follow the procedure

Close record → Auditable

③ SOC console — Operator

The operator does it all on one page

Single-page console
Verification, SOP dispatch and one-click alerting together.

Industry dashboards
Security, traffic, retail and industrial safety, each modular.

Close records and audit
Full lifecycle on the record, with reports you can hand over.

A demo UI today, to be rolled out case by case.

When is WatchNode worth layering on?

Enough events to need a process
Seeing them is not enough — someone must be assigned, follow the steps and close it.

Different events, different steps
Fire and intrusion are not handled the same way; the SOPs have to differ.

An audit trail you can produce
Who did what, and when, answerable after the fact.

Where XMap ends

XMap
Standard event handling — enough for most

WatchNode
Custom dashboards and cross-system dispatch