AGENTIC AI + ESYNC OTA

Manufacturing Use Cases

January 2026

Software-Defined Manufacturing with Intelligence, Speed, and Adaptability

Reducing Cycle Time and Increasing Flexibility with eSync

Shrikant Acharya
CTO, Excelfore

Agentic AI + eSync OTA

Three Use cases for Software-Defined Manufacturing with Agentic AI Integration

1. Executive Summary

Automotive manufacturing plants are increasingly constrained by software complexity across dozens of ECUs per vehicle. Traditional methods of using laptops, USB drives, and manual flashing at separate stations introduce inconsistencies, increase costs, and pose safety risks.

Making use of new capabilities driven by Agentic AI, Excelfore eSync production flashing replaces manual steps with cloud-orchestrated AI-optimized automation that's auditable, secure, and built to scale. While there are tremendous benefits to a full in-vehicle integration with eSync, this white-paper focuses on three practical manufacturing use cases deployed with an external (not in the vehicle) eSync system. These use-cases demonstrate how eSync is improving inventory control, line flexibility, cycle time, pre-shipment quality, and labor efficiency.

2. Historical perspective: how we got here

1990s–early 2000s:

- **Islands of automation, vendor-locked tooling, paper travelers**: Lines "ran" but were blind—no common data model, no feedback loop beyond end-of-line scrap/rework.

Mid-2000s–2010s:

- **Lean-over connectivity**: Kaizen squeezed motion and work-in-progress (WIP), but software on products exploded. Electronic control units (ECU) flashing stayed manual/laptop-based, often air-gapped for "safety," so cycle time variability and mis-build risk persisted.

2010s IoT pilot era:

- **Many proof-of-concepts, few plant-wide rollouts**: Operation/Information Technology (OT/IT) split, security fears, and heterogeneous protocols kept stations not fully connected; results trapped in point dashboards, not enterprise systems.

2020s reality:

- Product variants multiplied, compliance tightened, and recalls proved that line software operations are not optimized—updates, validations, and audit trails remain fragmented.

The cost of "if it ain't broke, don't fix it"

Over time, the mantra has turned into a tax. When flashing remains laptop-driven and stations are half-connected at best, small frictions—retries, babysitting, walk-downs—get normalized as "just how our line runs." Cycle time becomes volatile because reprogram durations are serial and unpredictable, and a hiccup at one station ripples upstream and

down. Version drift sneaks in: two identical vehicles leave with different software because each cell followed a slightly different playbook, and there's no VIN-level proof tight enough to satisfy an auditor without a scavenger hunt. Data that should fuel improvement never escapes paper travelers or orphaned CSVs, so lessons learned don't compound. Every new variant or supplier drop triggers bespoke procedures and drawn-out validations. Meanwhile, ad-hoc tools and thumb drives widen the attack surface and create compliance gaps—quiet liabilities until they're not.

Briefly put, "ain't broke" has meant:

- Hidden downtime and labor costs that don't show on the scoreboard
- Cycle volatility that managers fight rather than fix
- Quality escapes and expensive field reflashes
- Slow, one-off change management and audit risk

What changes now

The pivot is to treat software operations as first-class manufacturing. That means a connected, policy-driven, and auditable flow where updates, validations, and evidence move with the product—not with a person's laptop. A closed loop replaces fire-and-forget: the line **collects** telemetry as it works, **analyzes** it in near-real time, **decides** based on policy and model signals, and **acts** autonomously—whether that's choosing the right variant, retrying safely, or quarantining a unit.

Also, now we use AI Agents to reconfigure the eSync Agent automatically to match the requirements of the devices and to account for the edge devices and bus timing requirements of the in-vehicle networking of whichever vehicle it has been plugged into. Continuous optimization becomes the default: stations stabilize cycle times, first-pass yield rises, and each campaign gets easier because the system remembers what succeeded and why. The organization shifts from handcrafted fixes to repeatable playbooks that scale across cells and plants.

In practice, this looks like:

- Policy-gated updates and tests with VIN-level proof by design
- Always-on connectivity that turns edge data into decisions
- Agentic AI based automation that reduces variance while preserving security
- Rollouts that accelerate instead of resetting for every variant

3. The Problems We're Solving

Excelfore is tackling several critical challenges associated with production line flashing, including:

- **Fragmented tools** across stations lead to inconsistent versions and rework
- **Long, serial flash cycles** inflate cycle time and starve downstream stations
- **Rigid processes** inhibit rapid response to late BOM changes and order-mix shifts
- **Poor traceability** makes root-cause analysis and compliance painful

In order to address these issues, **eSync** establishes a unified, standards-based OTA pipeline across manufacturing and field operations:

- **Server (Cloud): Agentic AI** campaign orchestration, policy, and audit

- **Client (Gateway/OBD):** secure link and execution coordination
- **eSync OTA Agents (per-edge environment), as needed:** protocol-specific flashing, validation, and reporting

4. Using Agentic AI to Optimize the Flashing Processes

	Manufacturing Challenge	Approach	AI Optimization
1	The same base ECU may be used with different software configurations in different vehicle models. Pre-programming limits flexibility and drives high inventory costs.	**OTA gang programming**: A rack can detect the devices, and OTA campaigns can install the latest software for the configurations needed for specific vehicle models.	Use AI Agent to reconfigure the eSync Agent automatically to match the requirements of the devices that have been placed in the rack.
2	On the production line or in post-production inventory, software may require updates. eSync might not be pre-installed in the vehicle. An external device that is easily portable, could be used to update each vehicle when it is plugged in.	**OBD dongle for programming**: A single OBD dongle reads the VIN and self-configures each time, to serve as the OTA client for any vehicle in an open eSync campaign.	Use AI Agent to reconfigure the eSync Agent to automatically account for the edge devices and bus timing requirements of the in-vehicle networking of whichever vehicle it has been plugged into.

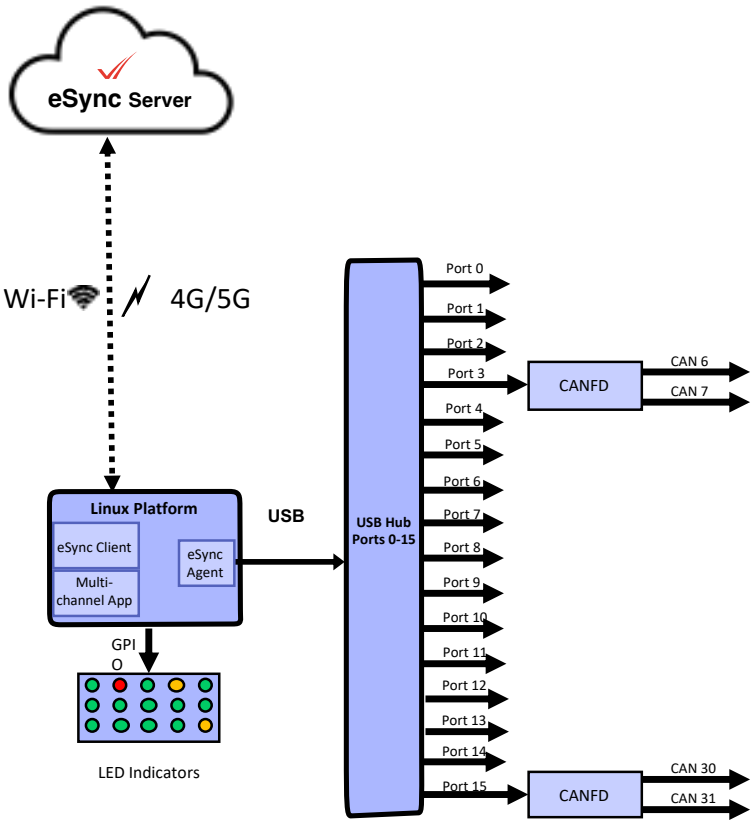
5. Three Manufacturing Use Cases: Integrated into the Flow for automotive OEMs and Tier-1s

Below, we illustrate three use cases that directly integrate into the production workflow—from inbound inventory to end-of-line and pre-shipment.

a. **<u>Pre Production Line Flashing: Build-Time ECU Personalization and Gang Flashing to Minimize Inventory Complexity</u>**

Deployment Use Case: European Tier-1

Challenge: Manufacturing plants often maintain multiple pre-flashed ECU variants for each trim or market, tying up valuable capital and driving up inventory and operational costs. Serial flashing of multiple ECUs per vehicle extends cycle time and increases work-in-process.



How Does eSync with Agentic AI Make a Difference?

In this scenario, an automatic flash process is executed as follows:

1. Receive a blank or base-image ECUs from suppliers
2. Based on the vehicle sub-model for the targeted ECU, the eSync Client automatically requests the correct variant of the associated firmware/software from the server based on the vehicle's configuration and options
3. An AI Agent auto-configures to varying sub-model specifications
4. Gang programming: Agents attached to multiple channels in a hub (e.g., 16–32 CAN/CAN-FD or mixed DoIP + UDS) flash ECUs in parallel
5. An eSync Agent automatically executes the flash, runs a functional test script, validates each ECU and posts results and artifact hashes to the audit log

Impact:

- Shrinks the SKU matrix; reduces work in progress and finished-goods buffers
- Eliminates mis-builds from incorrect pre-flashed stock
- Enables last-minute software variant changes without scrapping inventory
- Improves inventory management and traceability with audit logs

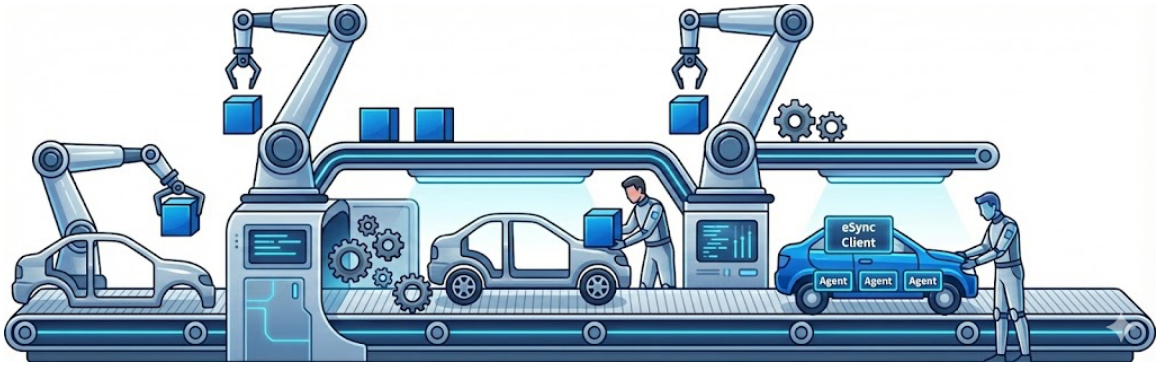
With this eSync process, build-time ECU personalization occurs at the first software insertion point, replacing pre-flashed inventory with on-the-line, configuration-accurate programming.

ROI for the European Tier-1			Key Values
	Original	**with Excelfore**	
Av time spent per issue (in weeks)	4	0.5	
% overhead to retrieve and analyze	15%	5%	• Simplified Work Flow • Improved quality and SKU audit • Substantial Cost Reduction
Resource Cost per issue	€16,800	€2,100	
Ave Issue per year	4		
Engineer Rate/hr	€105		
Total Cost per project + 15% Overhead	€79,800	€8,820	
Annual Cost	€319,200	€35,280	
% Savings	89%		

b. <u>Production line Flashing: Simplifying Production-Line to Improve Safety, Reduce Labor Time and Cycle Time</u>

Deployment Use Case: Japanese OEM

Challenge: Manual laptop-based flashing consumes skilled labor and introduces safety risks (runners in/around moving lines). Single-station dependency creates bottlenecks and increases MTTR (Mean time to Repair) when equipment or staffing issues arise.



How Does eSync with Agentic AI Make a Difference:

1. A single, policy-driven workflow is triggered from a plug-and-safe gateway/OBD device replacing per-station laptops and USB media
2. Agentic AI Drives Autoconfiguration for different build packages on the production Line
3. Specifies authorized stations (e.g., Stations 11, 14, and 22) where the same or subsequent campaigns can be executed under policy (e.g. dependency)
4. If a station is down, the campaign automatically shifts to the next enabled point, with no rework of manifests
5. Automated logging eliminates handwritten and manual reconciliations

Impact:

- Reduces overtime and the need for engineers moving on the line
- Lowers rework from human error; improves ergonomics and safety
- Smooths variance and maintains target cycle time
- Frees skilled staff to focus on root-cause and continuous improvement
- Absorbs disruptions without starving downstream stations

This eSync process spans initial programming stations, rework cells, and pre-ship campaigns. Software can be securely installed at any authorized station that meets the policy requirements—eliminating the need for a fixed location.

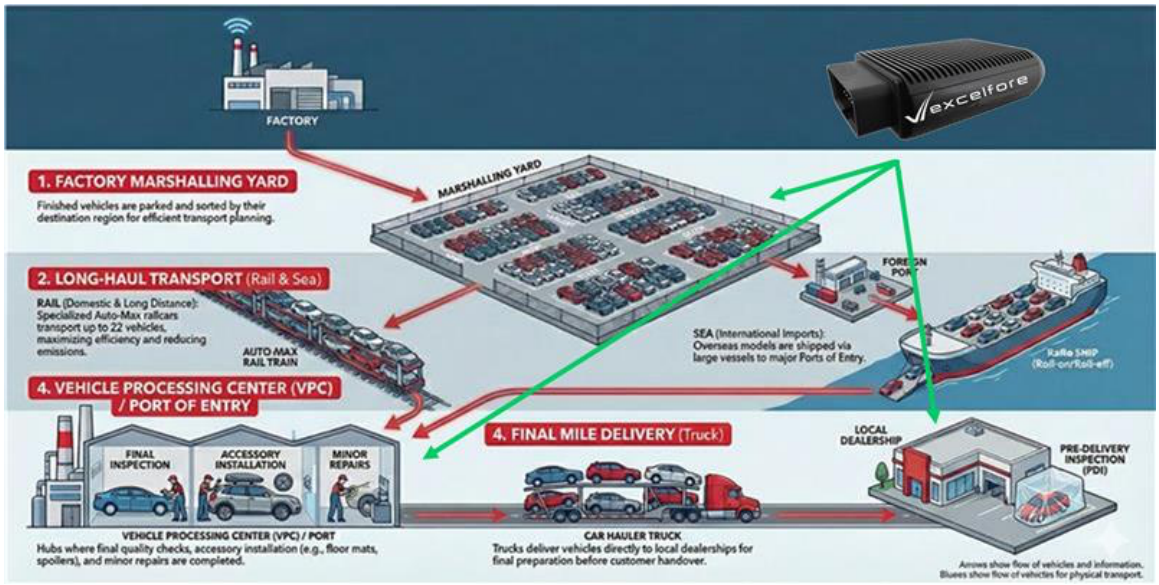
ROI for the Japanese OEM			Key Values
	Original	**with Excelfore**	
Number of Personnel	25	2	• Enhanced Technician Safety • Improved quality and SKU audit • Substantial Cost Reduction
Vehicles produced annually	82,800	82,800	
Personnel Cost	$777,778	$62,222	
Gross Cost Savings†		92%	

† excludes payments to Excelfore

c. Post Production Line Flashing: Pre-Shipment Fleet Update & Conformance Sweep

Deployment Use Case: US OEM

Challenge: Software may evolve while vehicles transit to the dealership or remain in post-production holding (points 1 and 4 in the image below), causing quality issues when units ship with inconsistent software versions.



How Does eSync with Agentic AI Make a Difference:

1. Manufacturing / Distribution Control
    - Ruggedized OBD Dongle
    - Built-in security for OTA
    - eSync OTA workflows
2. AI Agent drives autoconfiguration for seamless ECU integration and scale
    - One dongle serves many different vehicles
3. Run campaigns against specified model type to update all target ECUs to the current approved version
4. Generate digitally signed conformity reports per secure-ID (or VIN) for shipping docs and downstream service systems

Impact:

- Ensures that vehicles have consistent, latest-approved software when they arrive at the dealership; Avoids dealer reflash programs and reduces warranty exposure
- Lowers yard-handling time via automated wake/flash/verify cycles

With this eSync process, a critical pre-shipment software update is performed prior to dealer deliveries, which is integrated with MES and yard management systems.

ROI for the US OEM			Key Values
	Original	**with Excelfore**	
Number of Personnel	38	2.5	• Faster Time-to-Delivery • Improved quality and Audit • Substantial Cost Reduction
Number of Vehicles produced annually	124,200	124,200	
Personnel Cost	$4,200,000	$280,000	
Cost Savings†		93%	
† excludes payments to Excelfore			

6. Metrics and KPIs to Track

Following are examples of high-level metrics that are tracked and made available:

- **Cycle-Time Contribution:** Average flashing & validation time per vehicle and variance
- **First-Pass Yield (FPY):** Per-ECU and per-vehicle pass rates after flash & test
- **Station Utilization:** % time policy conditions met and campaigns executed
- **Inventory Turns:** Reduction in ECU variant SKUs and carrying costs
- **Labor Hours per Unit (LH/U):** Before/after for line and yard
- **Escape Rate:** Post-shipment software deviations detected

7. Takeaways

Reducing cycle time, streamlining inventory, and improving first-pass yield isn't about adding more laptops or personnel—it's about orchestrating software through a cloud-to-edge pipeline that understands the vehicle, the station, and the policy in real time, while minimizing human errors.

All the use cases described above lend themselves to Agentic AI optimization. Excelfore is leading with Agentic AI integration in gateways for production use-cases.

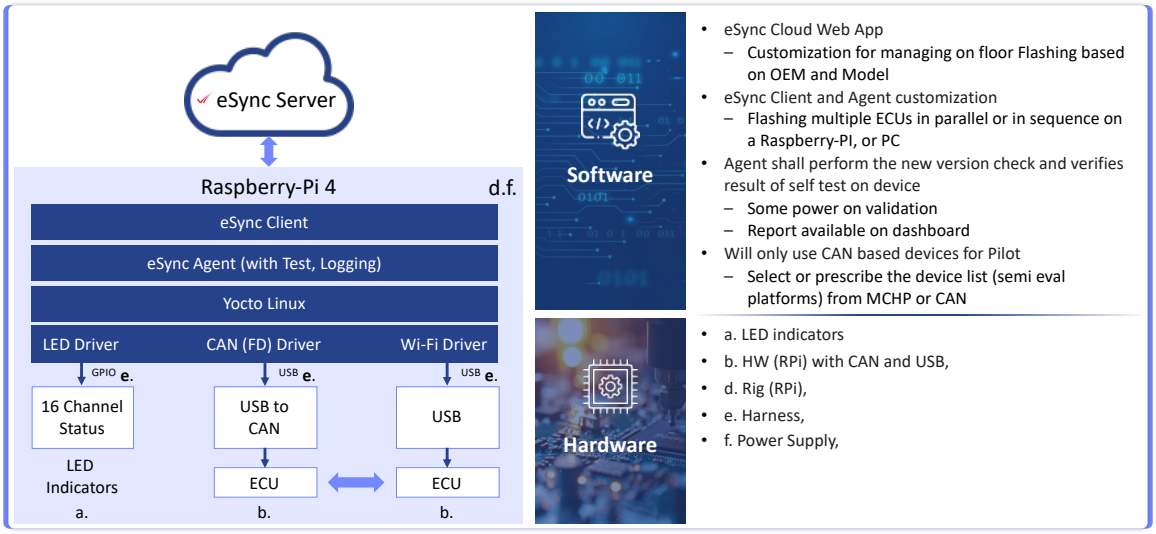
The use-cases specified above have been integrated with the OEM's MES/PLM systems with lightweight APIs/signals for station-ready deployment. These systems are securely deployed with end-to-end signing, per-ECU agent isolation, and immutable audit trails.

With eSync, manufacturers convert flashing from a labor activity to a controlled, parallelized, auditable workflow—on the line and in the yard. Automation is essential for reducing errors, and eSync delivers it in abundance.

8. Getting Started

Excelfore offers a simple way to get started with its offerings on AWS, Google and Azure marketplaces. After signing up on the marketplace, the following process for rapid integration for a pilot can be achieved within days.

1. **Pilot:** Use Raspberry Pi for performing one multi-ECU flashing demonstration with limited feature set
2. **Policy Tuning:** Calibrate state conditions and link priorities to your line cadence
3. **Scale to Yard:** Add pre-shipment batch campaigns and conformance reporting
4. **Rollout Playbook:** Replicate across lines and plants with shared manifests and dashboards



eSync software organization in a gateway (e.g. Raspberry Pi)

Resources:

- eSync Agent Software Developer's Kit
- Request a demo

Contact Us:

- sales@excelfore.com
- www.excelfore.com