FAQ - Palletforce Cross-Dock & Barcode

Answers to common questions about Palletforce cross-docking in Qargo: barcode generation, mobile scanning, supported scan codes, and known limitations.

Updated 3 July 2026

Q: How does Qargo now support Palletforce cross-dock operations, and what data is provided at dispatch? What happens if I have a multi-depot pallet network setup - will the barcode still scan?

A: Qargo provides comprehensive support for Palletforce cross-docking by enabling barcode generation early in the workflow and facilitating mobile scanning updates.

- **Barcode Generation at Dispatch:** Pallet barcodes are now generated directly in Qargo immediately after an order is imported or dispatched. This ensures that every pallet has a corresponding barcode assigned to its goods trace *before* the cross-docking process begins.
- **Mobile Scanning & Transmission:** Users can use the Qargo mobile application to scan these Palletforce labels during the physical cross-dock process. This scan data is then sent to Palletforce to update consignment tracking and status.

‼️ Palletforce may occasionally change pallet barcodes after consignment creation. If a label is scanned in Qargo but the barcode has changed externally, the scan may fail, as the barcode registered in Qargo will not match the label.

Scanning workflow

Cross-docking allows drivers and depot staff to scan Palletforce barcodes using the Qargo mobile app. Each scan transmits a status update directly to Palletforce.

The standard scanning sequence for a cross-docked pallet is:

1. **Collection from shipper** — scan triggers a *SCOT* (Collection from Shipper) status update to Palletforce.
2. **Receipt at inbound depot** — scan triggers a *SRID* (Scan on Receipt at Inbound Depot) status update.
3. **Arrival at delivery depot** — scan triggers an *ARRD* (Arrival at Delivery Depot) status update.
4. **Delivery to consignee** — scan triggers a *DELV* (Delivery) status update.

Supported scan codes

The following Palletforce scan codes are supported via the Qargo mobile app:

Code	Description
SRID	Scan on receipt at inbound depot
SCOT	Collection confirmation from shipper
ARRD	Arrival at delivery depot
DELV	Delivery confirmation

Barcode changes after generation

If the delivery depot or service level on an order is changed after the cross-docking barcode has been generated, the existing barcode will no longer reflect the correct routing.

- The barcode must be regenerated before the pallet is scanned into the Palletforce network.
- Scanning an outdated barcode may cause routing errors at the inbound depot.

External barcode changes

If Palletforce assigns a different tracking number to the consignment after export (for example, due to a depot reassignment), the barcode on the physical label will differ from the one in Palletforce's system.

- The label must be reprinted after the correct tracking number is returned from Palletforce.

Handling Multi-Depot barcode scans

- **What does this feature do?**
  - It makes cross-dock barcode resolution ***depot-aware***. When several orders share the same barcode, a scan is attributed to the order whose consignment actually transits the depot where the scan is performed.
- **Why does it matter?**
  - In multi-depot pallet network setups, the same physical consignment can exist as two orders that share the same barcode. Previously the scan was attributed to the most recently created order, which could be the wrong depot - sending incorrect status updates back to the network.
- **How does it work?**
  - The pallet shown on screen, the stop that is completed, and the order the scan attaches to all reflect the correct depot. Outbound *Palletforce* status updates sync to the integration instance of the scanning depot. Unique-barcode scanning is unchanged - the depot tiebreaker only runs when a barcode is genuinely ambiguous.

Use Cases

- A carrier runs a multi-depot pallet network where a consignment is dispatched from one depot to another. The same consignment exists as two orders sharing one barcode and tracking ID. When a driver scans at the sending depot, the scan now attaches to that depot's order and the correct status flows back to the network, rather than being attributed to the receiving depot's order.

Terminology

Term	Definition
Cross-dock scanning	The process of scanning a pallet barcode at a depot to record its movement and complete the relevant stop.
Barcode resolution	The step that matches a scanned barcode to a single goods trace (and therefore to one order).
Depot-aware resolution	Resolution that uses the depot where the scan is performed to choose the correct order when a barcode is shared.
Goods trace	The internal record that links a barcode to a specific order and consignment.
Integration instance	The connection between a depot and the pallet network used to sync status updates.

How Depot-Aware Resolution Works

1. **SCAN** a pallet barcode at the depot during cross-dock handling.
2. **VISUALISE** the pallet details on screen - these belong to the order that actually moves through the scanning depot.
3. **COMPLETE** the scan - the resulting status update completes the correct stop on the correct order.

4. The outbound *Palletforce* depot event (collection or delivery) is attached to the integration instance of the scanning depot.

🧰 When a barcode is unique within the tenant - the vast majority of scans - behaviour is identical to before. **The depot tiebreaker only runs when more than one order shares the same barcode.**

Troubleshooting

Issue	Guidance
A scan still attaches to the wrong order	Check whether both consignments sharing the barcode have a stop at the scanning depot. In that case the system falls back to the previous most-recent-match behaviour and logs a warning.
A bulk or offline scan resolves differently	Bulk and offline scans, and older mobile builds, rely on depot-aware barcode resolution rather than the round-tripped pallet identifier. Accuracy is the same, but resolution happens on the backend.
Incident report lookups are not depot-specific	Bulk barcode lookups used by incident reports are not depot-scoped in this version.