PROCURANT SUPPLIER API DOCUMENTATION

Purchase Order & PO Change JSON Reference

The field-by-field structure of the order documents Procurant sends to you — the 850 (new Purchase Order) and 860 (Purchase Order Change).

How to read this document

This document describes data you **receive**, so "requirement" means something slightly different than in Documents 3–5: it's about what you can rely on being present, and what you need to act on correctly.

ALWAYS PRESENT Populated on every PO and PO Change — safe to always parse.

BUSINESS-CRITICAL May be blank, but when present it drives a real decision downstream — pricing, compliance, fulfillment, or routing.

CONDITIONAL Populated only in specific situations — e.g. only when cross-docking applies, or only on a PO Change.

RESERVED / LATER Part of the shared schema but filled in at a later lifecycle stage (Order Ack, ASN, Invoice). Expect it to be null on an incoming PO.

Contents

1. What you receive, and when
2. The shared location block pattern
3. Order header fields
4. Line item fields (purchaseOrderLines)
5. Charges, comments & order totals
6. Telling a PO Change from a new PO
7. Sample payload
8. Fields seen in production not yet in this spec

1 What you receive, and when

Procurant sends two document types built from the same JSON schema:

DOCUMENT	EDI EQUIVALENT	WHEN YOU RECEIVE IT
Purchase Order	850	A buyer places a new order with you. `status` is `"New"`.
Purchase Order Change	860	A buyer revises an order already sent to you — quantities, pricing, dates, or line items. `status` is `"Modified"`.

BUSINESS REQUIREMENT

Delivery mechanism (webhook callback vs. a pull endpoint) and the base URL/credentials for receiving these documents are configured per trading partner during onboarding — see Document 1. This reference covers the payload shape only.

WHY ONE SCHEMA FOR TWO DOCUMENTS

Procurant reuses a single, wide order schema for both the initial PO and any later change to it, rather than defining a separate format for each. That keeps your parsing logic simple — one model to build — but means many fields you'll see are irrelevant to a plain new order (Section 3 flags these) and some only become meaningful once a change occurs (Section 6). In addition to this shared-schema pattern, Procurant can also deliver a new order and a later change to it as two separate document feeds — a distinct New Order message and Order Change message — depending on how your integration is configured. Confirm with your Procurant integration contact which pattern applies to your setup.

2 The shared location block pattern

Eight different roles on the order — delivery, ship-from, cross-dock, supplier, buyer, carrier, bill-to, and remit-to — each use the *same* set of suffixed fields. Learn the pattern once:

FIELD	TYPE	REQUIREMENT	DESCRIPTION
Location	string	API REQUIREMENT	Human-readable facility or company name for this role.
Address	string	BUSINESS-CRITICAL	Street address.
Phone	string	OPTIONAL	Contact phone number.
Duns	string	BUSINESS-CRITICAL	DUNS number identifying this trading-partner location inside Procurant.
Code	string	BUSINESS-CRITICAL	Procurant/EDI-assigned location code — used interchangeably with DUNS to identify the location.
City / StateCode / CountryCode / ZipCode	string	OPTIONAL	Postal address components.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
Contact	string	CONDITIONAL	Named contact for this location. Present on `delivery`, `shipFrom`, and `crossDock` blocks only. Required while your trading-partner setup with Procurant is still incomplete; becomes optional once that setup is complete.

Prefixes that use this pattern:

PREFIX	ROLE
`delivery`	Where the order ships to.
`shipFrom`	Your facility the order ships from (also repeated per line item — see Section 4).
`crossDock`	An intermediate cross-dock facility. Only meaningful when `crossDockFlag` is `true`.
`supplier`	You, the supplier, as Procurant has you configured.
`buyer`	The buyer placing the order.
`carrier`	The carrier moving the freight, if known at PO time.
`billTo`	Who should be billed — relevant when you invoice (Document 4).
`remitTo`	Where payment should be sent — relevant when you invoice (Document 4).

BUSINESS REQUIREMENT

The `Duns/Code` pair for each location must match how that location is configured in your own system. A PO can be perfectly well-formed JSON and still fail to match the right facility on your end if these identifiers are out of sync — that's a data setup problem, not a schema problem. Flag mismatches to your Procurant integration contact.

3 Order header fields

Fields below apply to the order as a whole. Location-block fields (delivery, shipFrom, supplier, buyer, etc.) follow the pattern in Section 2 and aren't repeated here.

IDENTIFICATION & STATUS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
poNumber	string	API REQUIREMENT	The purchase order number. Your primary key for this order.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
poType	string	API REQUIREMENT	Order category, e.g. `BUYER_PO`. Currently observed as a fixed value in production — treat any other value as informational.
status	string	BUSINESS REQUIREMENT	`"New"` for a first-time PO, `"Modified"` for a PO Change. Drive your intake logic off this field — see Section 6.
poStatus / poSubStatus	string	OPTIONAL	Procurant workflow status, distinct from `status`. Informational.
referenceNumber	string	OPTIONAL	Free-form reference number, when supplied by the buyer.
version / docRevision	integer	OPTIONAL	Internal revision counters. `docRevision` increments with each PO Change to the same order.
createdBy	string	OPTIONAL	Name of the user or system that created the record on the buyer/Procurant side.
offline	boolean	OPTIONAL	Internal flag indicating the order originated outside the standard flow.

DATES & SHIPPING

FIELD	TYPE	REQUIREMENT	DESCRIPTION
createDate / createTime	string	API REQUIREMENT	When Procurant created this record.
updateDate	string	API REQUIREMENT	Last modification timestamp. Changes whenever the PO or a PO Change is saved.
shipDate	string	BUSINESS REQUIREMENT	Requested ship date for the order (line items may override — Section 4).
deliveryDate	string	API REQUIREMENT	Requested delivery date. Always present — mandatory on every PO and PO Change.
deliveryDateModFlag	integer (0/1)	CONDITIONAL	`1` if `deliveryDate` changed on this PO Change versus the prior version.
routing	string	BUSINESS REQUIREMENT	Shipping/incoterm method: `"FOB Carrier"`, `"Delivered"`, or `"Backhaul"`. Map this to your own routing logic.
soNumber	string	CONDITIONAL	Your sales order number, once you've assigned one back through an Order Acknowledgment.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
loadNumber / demandNumber	string	OPTIONAL	Populated later in fulfillment planning; typically blank on a new PO.
billOfLading	string	OPTIONAL	Bill of lading number. Populated once shipping documentation exists — expect null on a new PO.

CROSS-DOCK & FLAGS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
crossDockFlag	boolean	API REQUIREMENT	`true` if this order routes through an intermediate cross-dock facility (see the `crossDock` location block).
addItemBR	boolean	OPTIONAL	Internal business-rule flag related to adding items. Safe to ignore unless instructed otherwise.
totalLineWeight	number	OPTIONAL	Sum of line-level weights, provided for convenience.

CONTACTS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
buyerUserCode / FName / LName / Phone / Email	string	OPTIONAL	The specific buyer-side employee associated with the order — useful for escalations.
supplierUserCode / FName / LName / Phone / Email	string	OPTIONAL	The specific supplier-side (your) contact on file for the order, if one is set.

RESERVED FOR DOWNSTREAM DOCUMENTS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
invoiceNumber / invoiceStatus / invoiceNotes / invoiceCreatedDate	—	OPTIONAL	Populated once *you* submit an Invoice (Document 4). Always null on an incoming PO or PO Change.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
paymentTermName / paymentNetDays / paymentNetDaysFrom / paymentDueDate / paymentPercentDiscount / paymentDiscountAmount / paymentDiscountDays / paymentDiscountDueDate / discountAmountDue	—	OPTIONAL	Payment-term fields, relevant when invoicing. See Document 4 for how they're used.
stmtTitle / stmtContent	string	OPTIONAL	Legal/invoice statement text (e.g. a PACA trust notice). Populated at invoicing — see Document 4.
buyerInvoicedTotalAmt / supplierInvoiceTotalAmt / buyerTotalAmt	number	OPTIONAL	Invoice-stage total amounts. Null until invoicing occurs.
loadStatusCode	string	OPTIONAL	Internal fulfillment/load status, populated later.

4 Line item fields (purchaseOrderLines)

Each entry in the `purchaseOrderLines` array describes one ordered item. A line item can also carry its own `shipFrom*` location block (Section 2) when it ships from a different facility than the header default.

IDENTIFICATION

FIELD	TYPE	REQUIREMENT	DESCRIPTION
lineNumber	string	API REQUIREMENT	Sequence number for the line. Use this as your key when matching an Order Acknowledgment line back to this PO line.
soNumber / status / demandNumber	string	OPTIONAL	Line-level overrides of the header values, when they differ.

PRODUCT IDENTIFIERS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
buyerProductCode	string	API REQUIREMENT	The buyer's product code. Map this to your own SKU — it's the primary product identifier on an inbound PO.
buyerProductDescription	string	BUSINESS REQUIREMENT	Human-readable product description, useful for validating the code mapping.
buyerProductUPC / buyerProductGTIN	string	CONDITIONAL	Standard product identifiers — sent only when your account's business-rule configuration is set up to include them.
buyerProductPackType	string	BUSINESS REQUIREMENT	Pack/case type ordered (e.g. `CA`). Confirm this matches what you can ship.
supplierProductCode / Description / UPC / GTIN	string	BUSINESS-CRITICAL	Your own product identifiers, populated once your product-code mapping is set up in Procurant. Typically blank until that mapping exists.
tpUpc	string	BUSINESS-CRITICAL	Trading-partner UPC variant, when supplied. Newly observed in production — see Section 8.

QUANTITY & PRICING

FIELD	TYPE	REQUIREMENT	DESCRIPTION
quantity	integer	API REQUIREMENT	Quantity ordered.
unitPrice	number	API REQUIREMENT	Agreed unit price.
finalPrice / variancePrice	number	OPTIONAL	Computed once the order is priced/reconciled downstream.
receivedQuantity / invoicedQuantity / shippedQuantity	integer	OPTIONAL	Reserved for downstream documents — zero on a fresh PO.

PHYSICAL & LOGISTICS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
grossWeight / netWeight / itemWeight / weightUOM	number / string	BUSINESS REQUIREMENT	Weight specs used for load and freight planning.
length / width / height / dimensionUom	number / string	OPTIONAL	Case dimensions, when provided.
hi / ti	integer	OPTIONAL	Pallet stacking pattern (high × tie).
loadFreight	number	OPTIONAL	Freight amount allocated to this line.
priceByUOM	string	OPTIONAL	Pricing unit convention, e.g. `PACKTYPE`.

TRACEABILITY & COMPLIANCE

FIELD	TYPE	REQUIREMENT	DESCRIPTION
bioengineered	string ("Yes"/"No")	BUSINESS REQUIREMENT	Bioengineered-food disclosure for this product. Carry this through accurately — it originates from the buyer's product master and reflects a regulatory disclosure. Whether it's populated depends on your product-catalog setup.
countryOfOrigin	string	BUSINESS REQUIREMENT	Country of origin. Confirm against what you actually ship; mismatches are a compliance issue, not a validation error.
harvestDate / lotNumber	string	CONDITIONAL	Populated by you at shipping (ASN) time for perishable/traceable items — see Document 3.
balanceItem	boolean	OPTIONAL	`true` when this line represents the remaining balance of a previously split/partial line.

CHANGE TRACKING (PO CHANGE ONLY)

FIELD	TYPE	REQUIREMENT	DESCRIPTION
addItemModFlag	boolean	CONDITIONAL	`true` if this line was newly added by a PO Change rather than present on the original PO.
originalQty	integer	CONDITIONAL	The quantity originally ordered, for comparison against the current `quantity`.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
previousQuantity / previousUnitPrice	number	CONDITIONAL	The prior value before this change, so you can compute exactly what moved.
quantityModFlag / priceModFlag / shipDateModFlag	integer (0/1)	CONDITIONAL	`1` flags that quantity, price, or ship date changed on this line versus the prior version. All `0` on a brand-new PO.

RELATED ARRAYS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
charges	array	BUSINESS-CRITICAL	Line-level allowances/charges. See Section 5 for the shape — treat the detail as business-critical for accurate reconciliation whenever a charge is present.
comments	array	OPTIONAL	Free-text comments tied to this line. Each entry is `{"comment": string}`.
traceInfos	array	OPTIONAL	Not populated on an incoming PO — this is the structure you send back on the ASN (Document 3).

5 Charges, comments & order totals

charges (header and line level)

The same shape appears at both the order header and inside each line item.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
chargeCode	string	CONDITIONAL	Charge type code. Required only if a charge is given.
chargeName	string	BUSINESS-CRITICAL	Human-readable charge description.
chargeAmount	number	BUSINESS-CRITICAL	Amount of the charge (negative for allowances/discounts).
chargeTypeCode	string	RESERVED / LATER	Classification of the charge.
prorateByCode / applyByCode	string	RESERVED / LATER	How the charge should be prorated or applied.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
thirdPartyFlag / thirdPartyCode	boolean / string	RESERVED / LATER	Whether a third party (not buyer or supplier) is responsible for the charge.
includedInLC	boolean	RESERVED / LATER	Whether the charge is already included in the line's list/contract price.

comments (header and line level)

An array of `{"comment": string}` objects carrying free-text notes.

orderSummary

A header-level roll-up of the order's totals, provided for convenience and reconciliation.

FIELD	DESCRIPTION
totalAmount	Total order value.
totalFreight / totalCharges	Freight and charge totals across all lines.
totalQty	Total quantity across all lines.
totalVolume / totalWeight / totalGrossWeight	Aggregate logistics figures.
totalLinePallets	Computed pallet count for the order.

> **BUSINESS REQUIREMENT**
>
> Reconcile `orderSummary` against your own extended totals when you process the PO. A mismatch usually means a pricing or quantity discrepancy worth flagging to the buyer before you acknowledge the order.

6 Telling a PO Change from a new PO

SIGNAL	WHAT IT TELLS YOU
`status`	`"New"` on the first PO; `"Modified"` on any later change.
`docRevision`	Increments with each successive change to the same `poNumber`.
`deliveryDateModFlag` (header)	`1` when the delivery date changed on this revision.

SIGNAL	WHAT IT TELLS YOU
`quantityModFlag` / `priceModFlag` / `shipDateModFlag` (per line)	`1` when that specific value changed on that line.
`addItemModFlag` (per line)	`true` when the line was added by this change and didn't exist on the original PO.
`previousQuantity` / `previousUnitPrice` (per line)	The prior value, so you can diff without keeping your own history.

BUSINESS REQUIREMENT

Use `poNumber` together with `docRevision` to recognize that an incoming message updates an order you already have, rather than processing it as a brand-new order. Re-acknowledging or double-fulfilling a changed order is a common integration bug.

7 Sample payload

Trimmed to one line item for readability; a real PO typically contains one entry per ordered product.

SAMPLE PURCHASE ORDER (850), STATUS = NEW


```
{
  "poNumber": "448253",
  "poType": "BUYER_PO",
  "createDate": "2026-08-30T19:00:55.337+00:00",
  "updateDate": "2026-08-30T19:00:55.769+00:00",
  "status": "New",
  "routing": "Backhaul",
  "shipDate": "2026-09-05T00:00:00.000+00:00",
  "deliveryDate": "2026-09-07T00:00:00.000+00:00",
  "deliveryLocation": "WSLC 5229 Salt Lake City Produce",
  "deliveryDuns": "0091372095229",
  "deliveryCity": "North Salt Lake",
  "deliveryStateCode": "UT",
  "shipFromLocation": "Watermelon Warehouse",
  "shipFromDuns": "0784047230001",
  "supplierName": "Hagerman Canyon Farms",
  "supplierDuns": "078404723",
  "buyerName": "Albertsons Companies Inc",
  "buyerDuns": "009137209",
  "crossDockFlag": false,
  "purchaseOrderLines": [
    {
      "lineNumber": "1",
      "buyerProductCode": "84020006",
      "buyerProductDescription": "APPLES FUJI 5LB BAG",
      "buyerProductPackType": "CA",
      "quantity": 110,
      "unitPrice": 10,
      "previousQuantity": 110,
```

```json
      "quantityModFlag": 0,
      "priceModFlag": 0,
      "addItemModFlag": false,
      "bioengineered": "YES",
      "countryOfOrigin": "USA"
    }
  ],
  "orderSummary": {
    "totalAmount": 12190.5,
    "totalQty": 229,
    "totalWeight": 39750.01,
    "totalLinePallets": 20
  }
}
```

8 Fields seen in production, not yet in this spec

Comparing this specification against current production PO and PO Change payloads surfaced a few fields the previous version of this document didn't cover. They're included in Sections 3–4 above; this table calls them out explicitly so nothing gets missed during review.

FIELD	WHERE	NOTE
poType	Header	Not previously documented. Currently observed as a fixed value (`BUYER_PO`) in production.
billOfLading	Header	Not previously documented. Null on a new PO; likely populated once shipping documentation exists.
tpUpc	Line item	Not previously documented. Appears alongside the standard UPC/GTIN identifiers.
supplierInvoiceTotalAmt	Header	Production data uses this spelling. If your integration was built against a field named `supplierInvoicedTotalAmt`, confirm the correct name with your Procurant integration contact.

The change-tracking fields (`previousQuantity`, `previousUnitPrice`, `quantityModFlag`, `priceModFlag`, `shipDateModFlag`, `addItemModFlag`, `originalQty`) surfaced the same way, but are already fully documented above in Sections 3–4 and Section 6, so they aren't repeated in this table.

RECOMMENDED NEXT STEP

These fields were identified by comparing this specification against current production sample payloads, not against Procurant's internal API contract. Confirm with your Procurant engineering/integration contact before building logic that depends on them.