PROCURANT

PROCURANT SUPPLIER API DOCUMENTATION

Order Acknowledgment API Specification

How to confirm what you can actually fulfill from a purchase order — quantities, pricing, and substitutions — before you ship.

How to read this document

API REQUIREMENT **API Requirement** — enforced by the service itself. Miss it and the call is rejected.

BUSINESS REQUIREMENT **Business Requirement** — needed for the transaction to be correct, compliant, or usable downstream, even though the API may technically accept the call without it.

CONDITIONAL **Conditional** — required only when a related field is present, or one of a small group of fields is required (see Section 6).

OPTIONAL **Optional** — safe to omit; include it when you have the data.

Contents

1 What this API does

Call this endpoint to tell Procurant what you can actually fulfill from a purchase order — the quantities, pricing, and any substitutions — before you ship. This is the Order Acknowledgment step in the order flow described in Document 1, the first of the three documents you submit to Procurant.

Buyer issues PO → **You send Order Ack ← you are here** → **You ship & send ASN** → **You send Invoice**

WHY THIS STEP MATTERS

The quantities and prices you confirm here are what downstream systems expect to see on your ASN and Invoice. Acknowledging a PO as-is when you actually intend to ship a different quantity, price, or product is the most common source of mismatches later in the order lifecycle.

2 Quick reference

Description	Submits an Order Acknowledgment for a purchase order.
REST Method	POST
URL pattern	`http://{edi-service-url}/api/message/orderackin`
Test environment	`https://test.procurant.io/edi/api/message/orderackin`
Request header	`Authorization: Bearer <JWT TOKEN>` — see Document 1
Request body	JSON object (see Sections 4–5)

3 The location block pattern

This API uses two different location patterns depending on where the field lives:

- At the **header level**, each party — `buyer`, `delivery`, `supplier`, `shipFrom`, `carrier`, `crossDock`, `remitTo`, `billTo` — is identified by a `Duns`/`Code` pair only; there is no header-level address block.
- At the **line level**, `shipFrom` gets a full address block, since a single order can ship from different locations line by line — see the ship-from block at the start of Section 5.

4 Order header fields

ORDER IDENTIFICATION

FIELD	TYPE	REQUIREMENT	DESCRIPTION
poNumber	string	API REQUIREMENT	The purchase order being acknowledged.
soNumber	string	BUSINESS REQUIREMENT	Your sales order number.
docRevision	integer	OPTIONAL	Revision counter for this acknowledgment. See Section 9.

DATES & ROUTING

FIELD	TYPE	REQUIREMENT	DESCRIPTION
shipDate / deliveryDate	string	API REQUIREMENT	Ship and delivery dates you're confirming for this order. Both fields are mandatory.
deliveryLocation	string	BUSINESS REQUIREMENT	Human-readable name of the delivery location.
routing	string	BUSINESS REQUIREMENT	Shipping/incoterm method: `"FOB Carrier"`, `"Delivered"`, or `"Backhaul"`.

PARTY IDENTIFICATION

FIELD	TYPE	REQUIREMENT	DESCRIPTION
buyerDuns	string	CONDITIONAL	Either this or `buyerCode` is required — see Section 6.
buyerCode	string	CONDITIONAL	Either this or `buyerDuns` is required — see Section 6.
deliveryDuns	string	CONDITIONAL	Either this or `deliveryCode` is required — see Section 6.
deliveryCode	string	CONDITIONAL	Either this or `deliveryDuns` is required — see Section 6.
supplierDuns	string	CONDITIONAL	Either this or `supplierCode` is required — see Section 6.
supplierCode	string	CONDITIONAL	Either this or `supplierDuns` is required — see Section 6.
carrierDuns / carrierCode	string	OPTIONAL	Carrier identification, when known at acknowledgment time.
crossDockDuns / crossDockCode	string	OPTIONAL	Cross-dock facility identification, when the order routes through one.
remitToDuns / remitToCode	string	OPTIONAL	Where payment should eventually be sent. Usually set once during onboarding rather than per order.
billToDuns / billToCode	string	OPTIONAL	Who should eventually be billed. Usually set once during onboarding rather than per order.

RELATED ARRAYS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
charges	array	CONDITIONAL	Header-level allowances/charges. `chargeCode` is required only if a charge is given.
comments	array	OPTIONAL	Free-text notes: `[{"comment": string}]`. Each comment string is limited to 254 characters.

STATUS / SUBSTATUS ARE SET PER LINE, NOT AT THE HEADER

The previous version of this document listed header-level `status/subStatus` fields that should always be `"Acknowledged"`. Current production acknowledgments leave these blank at the header level and set `status = "Acknowledged"` on each line item instead (Section 5). Confirm with your Procurant integration contact whether your environment expects the header-level fields as well.

5 Line item fields

Each entry in `purchaseOrderLines` describes what you're acknowledging for one item on the order. A line can carry its own ship-from address block (below) when it ships from a different location than the header default.

LINE-LEVEL SHIP-FROM BLOCK

FIELD	TYPE	REQUIREMENT	DESCRIPTION
shipFromDuns	string	CONDITIONAL	DUNS number for the shipping location on this line. Required (with `shipFromCode` as the alternative) when the line ships from a specific location.
shipFromCode	string	CONDITIONAL	Procurant/EDI-assigned code for the shipping location. Interchangeable with `shipFromDuns`.
shipFromLocation	string	BUSINESS REQUIREMENT	Human-readable name of the shipping location.
shipFromAddress / shipFromPhone / shipFromCity / shipFromStateCode / shipFromCountryCode / shipFromZipCode	string	OPTIONAL	Postal address and phone detail for the shipping location.

STATUS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
status	string	API REQUIREMENT	Should always be `"Acknowledged"` for this endpoint. Set on every line — see the callout in Section 4.

PRODUCT IDENTIFIERS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
buyerProductCode	string	CONDITIONAL	Either this or `supplierProductCode` is required — see Section 6.
supplierProductCode	string	CONDITIONAL	Either this or `buyerProductCode` is required — see Section 6.
buyerProductGtin / supplierProductGtin	string	OPTIONAL	Standard product identifiers, when available.
buyerProductUPC / supplierProductUPC	string	OPTIONAL	Standard product identifiers, when available.
buyerProductPackType	string	BUSINESS REQUIREMENT	Pack/case type you're acknowledging.

QUANTITY & PRICING

FIELD	TYPE	REQUIREMENT	DESCRIPTION
quantity	integer	API REQUIREMENT	The quantity you're confirming you can fulfill — the core number of the acknowledgment.
unitPrice	number	API REQUIREMENT	Price per unit you're confirming.
previousUnitPrice	number	CONDITIONAL	The PO's original unit price, when your confirmed `unitPrice` differs from it — lets the buyer see the change at a glance.
finalPrice	number	OPTIONAL	Computed/reconciled price, when applicable. Not currently supported on this endpoint — Procurant does not process a submitted value, so it's safe to omit.

PHYSICAL & LOGISTICS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
grossWeight	number	OPTIONAL	Case/line weight, when relevant to pricing or freight. Only a header-level total gross weight is considered downstream; per-line detail here is for your own reference.

SUBSTITUTION & TRACEABILITY

FIELD	TYPE	REQUIREMENT	DESCRIPTION
substituted	boolean	CONDITIONAL	Set to `true` when you're fulfilling this line with a different product than ordered.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
substitutedProductCode	string	CONDITIONAL	Required when `substituted` is `true` — identifies what you're actually shipping. Substitution relies on your product catalog having multiple supplier codes mapped to the single buyer code being substituted — confirm that mapping exists with your Procurant catalog/integration contact.
countryOfOrigin	string	BUSINESS REQUIREMENT	Country-of-origin labeling detail, carried through from the product master.
harvestDate / lotNumber	string	CONDITIONAL	Include when lot/traceability detail is already known at acknowledgment time.
bioengineered	string ("Yes"/"No")	BUSINESS REQUIREMENT	Bioengineered-food disclosure, carried through from the product master.

RELATED ARRAYS

FIELD	TYPE	REQUIREMENT	DESCRIPTION
charges	array	CONDITIONAL	Line-level allowances/charges. Same shape as header `charges`.
comments	array	OPTIONAL	Free-text notes tied to this line. Each comment string is limited to 254 characters.

charges (header and line level)

FIELD	TYPE	REQUIREMENT	DESCRIPTION
chargeCode	string	CONDITIONAL	Only mandatory if charge information is given.
chargeName	string	BUSINESS REQUIREMENT	Human-readable charge description.
chargeAmount	number	BUSINESS REQUIREMENT	Amount of the charge (negative for allowances/discounts).
chargeTypeCode / prorateByCode / applyByCode	string	OPTIONAL	Classification and proration rules for the charge.
thirdPartyCode	string	OPTIONAL	Third party responsible for the charge, when applicable.

FIELD	TYPE	REQUIREMENT	DESCRIPTION
includedInLC	boolean	OPTIONAL	Whether the charge is already included in the line's contract price.

6 Conditional field rules

These pairs are each “one of the two is required” — the API rejects the request if neither is present:

PROVIDE AT LEAST ONE OF...	IDENTIFIES...
`buyerProductCode` or `supplierProductCode`	The product on each line.
`buyerDuns` or `buyerCode`	The buyer.
`deliveryDuns` or `deliveryCode`	The delivery location.
`supplierDuns` or `supplierCode`	You, the supplier.

> **API REQUIREMENT**
>
> `chargeCode` is required only when a `charges` entry is present. `substitutedProductCode` is required only when `substituted` is `true`.

Unlike the PO, ASN, and Invoice APIs, `carrierDuns`/`carrierCode`, `crossDockDuns`/`crossDockCode`, `remitToDuns`/`remitToCode`, and `billToDuns`/`billToCode` are not required pairs on this endpoint — production acknowledgments routinely leave them blank.

7 Business rules & compliance

> **ACKNOWLEDGE WHAT YOU'LL ACTUALLY SHIP**
>
> `quantity`, `unitPrice`, and `substituted`/`substitutedProductCode` on this document set the baseline your ASN and Invoice are compared against. Confirming the PO's original values when you plan to ship something different creates a mismatch downstream instead of catching it here. Note that the underlying PO can still be shipped while it's in `"Modified"` status — a pending PO Change doesn't block you from shipping against the order.

> **BIOENGINEERED DISCLOSURE**

> As with the ASN and Invoice (Documents 3 & 4), `bioengineered` reflects a federal food-labeling disclosure. Carry it through from your product master rather than defaulting it.

> **LOCATION IDENTIFIERS**
>
> `Duns/Code` values must match your trading-partner setup in Procurant. A mismatch here can misroute the acknowledgment even though the JSON is valid.

8 Sample request

SAMPLE ORDER ACKNOWLEDGMENT REQUEST BODY (TRIMMED TO ONE LINE)


```json
{
  "poNumber": "46759358",
  "soNumber": "1216312",
  "routing": "FOB Carrier",
  "shipDate": "2026-09-08T00:00:00.000+0000",
  "deliveryDate": "2026-09-11T00:00:00.000+0000",
  "deliveryLocation": "UNFI Centralia",
  "deliveryDuns": "0079439130052",
  "buyerDuns": "0041130079153",
  "supplierDuns": "055635940",
  "supplierCode": "0556359400000",
  "billToDuns": "008118456",
  "purchaseOrderLines": [
    {
      "status": "Acknowledged",
      "buyerProductCode": "6449235",
      "buyerProductUPC": "007457483325",
      "buyerProductPackType": "CA",
      "supplierProductCode": "GGN903",
      "quantity": 32,
      "unitPrice": 23.85,
      "previousUnitPrice": 0,
      "bioengineered": "No",
      "countryOfOrigin": "United States",
      "substituted": false
    }
  ],
  "comments": []
}
```


9 Fields seen in production, not yet in this spec

Current production acknowledgment records include several fields not covered in the previous version of this specification — consistent with what Documents 2–4 found for the other document types. Procurant's own document-monitoring (DM) screens will show a generic payload during post-processing that reflects these same production fields, so don't be alarmed if it doesn't match this specification field-for-field.

FIELD	NOTE
docRevision / version	Revision counters, consistent with the other document types.
crossDockFlag / deliveryDateModFlag / addItemBR	Internal business-rule/change flags, consistent with the other document types.
totalBuyerAmount / totalSupplierAmount / totalLineWeight / totLoadFreight / buyerTotFreight / supplierTotFeight / buyerTotHeaderChg / supplierTotHeaderChg / buyerTotLineChg / supplierTotLineChg	Roll-up totals, consistent with the other document types.
receivingBRRuleEnabled / suppressSupplierTrigger / sendCommissionQty / sendGrowingRegionPMA / sendNonSSCC / sendShipFromDuns / sendShipToDuns	Account-level configuration flags rather than per-acknowledgment data.
previousQuantity / quantityModFlag / priceModFlag / shipDateModFlag / addItemModFlag	Line-level change-tracking fields alongside `previousUnitPrice`.
receivedQuantity / shippedQuantity / invoicedQuantity / variancePrice / balanceItem	Line-level fields that appear populated later in the order lifecycle, once shipping and invoicing have happened.
itemWeight / netWeight	Additional weight detail alongside `grossWeight`.

RECOMMENDED NEXT STEP

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