API Integration

Customer ID Documentation

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-10-13

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Customer ID

API Integration

The extension does not add a dedicated Customer ID REST route or GraphQL field. It extends Magento's existing data instead:

- Customer records use the EAV custom attribute `customer_id`.
- Orders expose the copied value as the OrderInterface extension attribute `customerid`.

Access remains subject to Magento's normal API authentication, ACLs and website scope.

Read a customer

Magento's standard customer endpoint can include the value under `custom_attributes`:

```
GET /rest/V1/customers/123
Authorization: Bearer <integration-token>
```

Relevant response fragment:

```
{
  "id": 123,
  "email": "purchasing@northstar-industrial.example",
  "custom_attributes": [
    {
      "attribute_code": "customer_id",
      "value": "B2B-01042"
    }
  ]
}
```

Customer search uses Magento's standard customer service contracts with the same attribute code. Writing `customer_id` through `custom_attributes` is restricted: only admin and integration API tokens may set it — customer and guest tokens get rejected on account create and customer/address repository saves. See Protected customer attributes for the shared mechanism. Test the exact payload against your Magento release and integration permissions before building a synchronization job.

Read an order

For a registered-customer order, the copied business number can appear in order extension attributes:

```
{
```

```
  "extension_attributes": {
    "customerid": "B2B-01042"
  }
}
```

The name is `customerid`, without an underscore, because Magento already uses `customer_id` for the internal customer entity relation. Guest orders normally have no copied Customer ID. The field names reference compares all three identifiers used by the extension.

Integration ownership

Choose one owner for customer-number creation:

- If Magento owns it, let automatic assignment run and send the resulting `customer_id` to the ERP or CRM.
- If the ERP owns it, leave automatic assignment off during import and write the ERP value to `customer_id`.

Duplicate values are rejected — across all websites when **ID range is global** is enabled, otherwise within the customer's website. Imports should handle the validation error instead of silently generating a replacement.

There is no module-specific GraphQL schema. A GraphQL integration needs a separately implemented field or another supported integration layer.