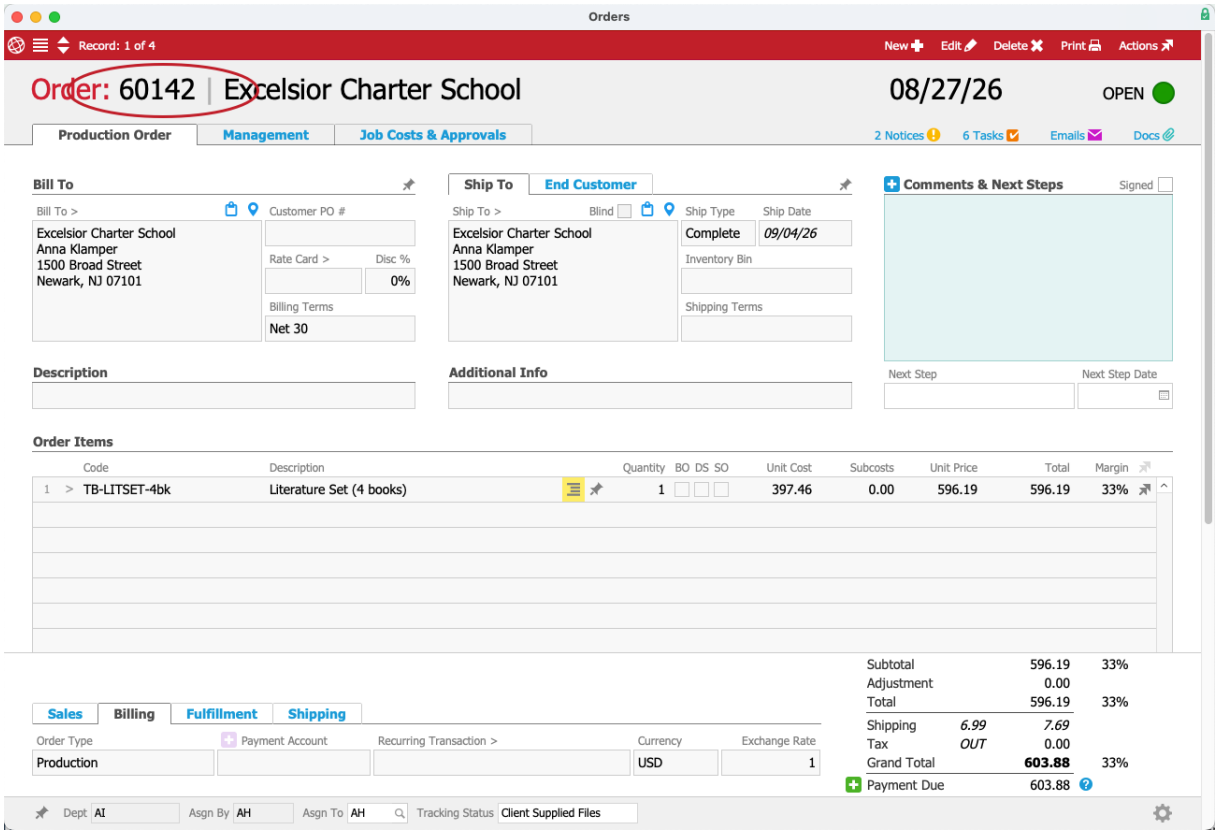
Understanding the aACE ID Numbering Scheme

Last Modified on 11/17/2026 5:20 pm EST

This guide explains the numbering logic for aACE records IDs. It is intended for general users.

Each record or transaction in aACE uses an ID. There is a difference between an ID and a record ID, and IDs and record IDs can be different.:

- ID — Visible in the top left on the detail view. Used by users for searching, sorting, reporting, and other activities within aACE.



- Record ID — Visible in the Log Viewer through Actions > View Logs. Used by programmers and the aACE Software team.

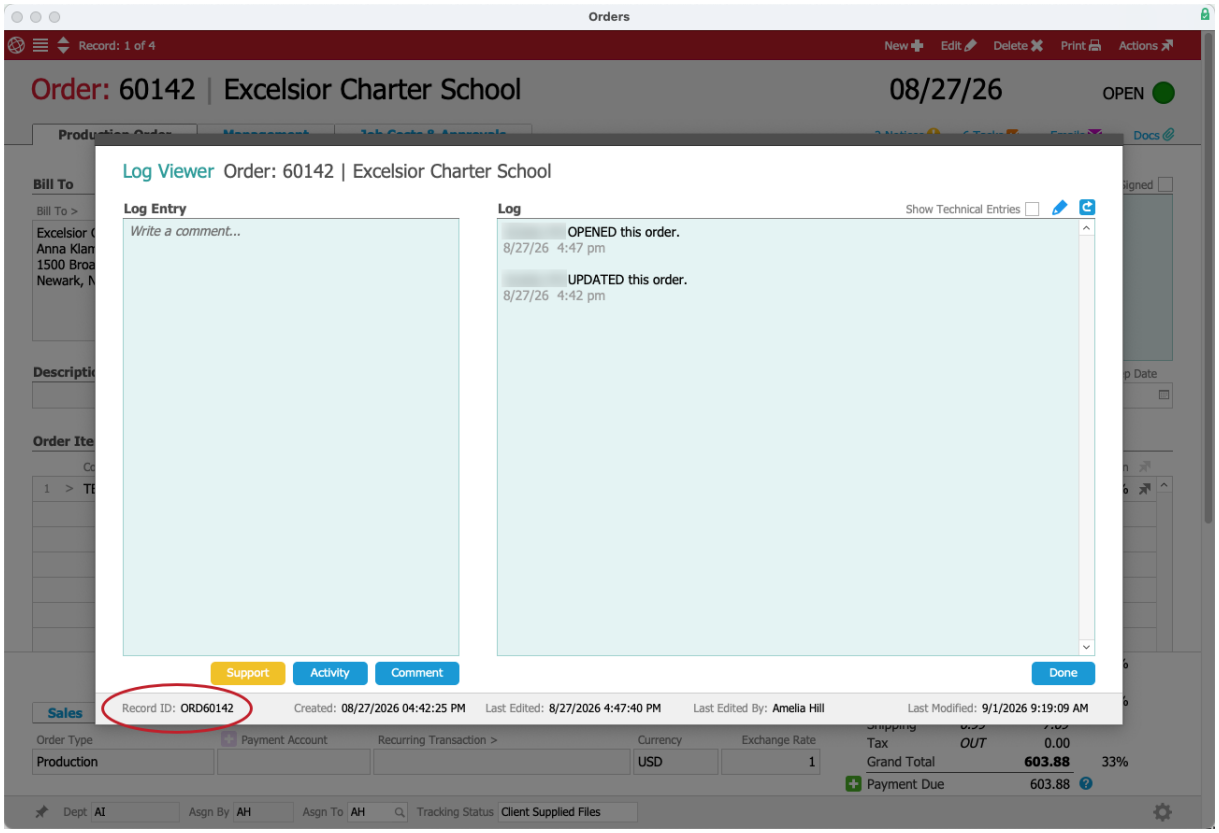


Table of Contents

- aACE Default Logic
- aACE Record Hierarchy

aACE Default Logic

For most modules, aACE assigns IDs and record IDs when new records or transactions are created beginning at #60000 and usually incrementing by 1 with each new record. These numbers only indicate the order that the records were created in. For example, a new purchase order ID is #60485 and the next new purchase order will be #60486.

However, the default logic changes when there are parent-child record relationships or orders belonging to a project. In these examples, the IDs change to inherit the parent ID with a child indicator to show hierarchy, but the record ID continues to increment by 1.

Top

aACE Record Hierarchy

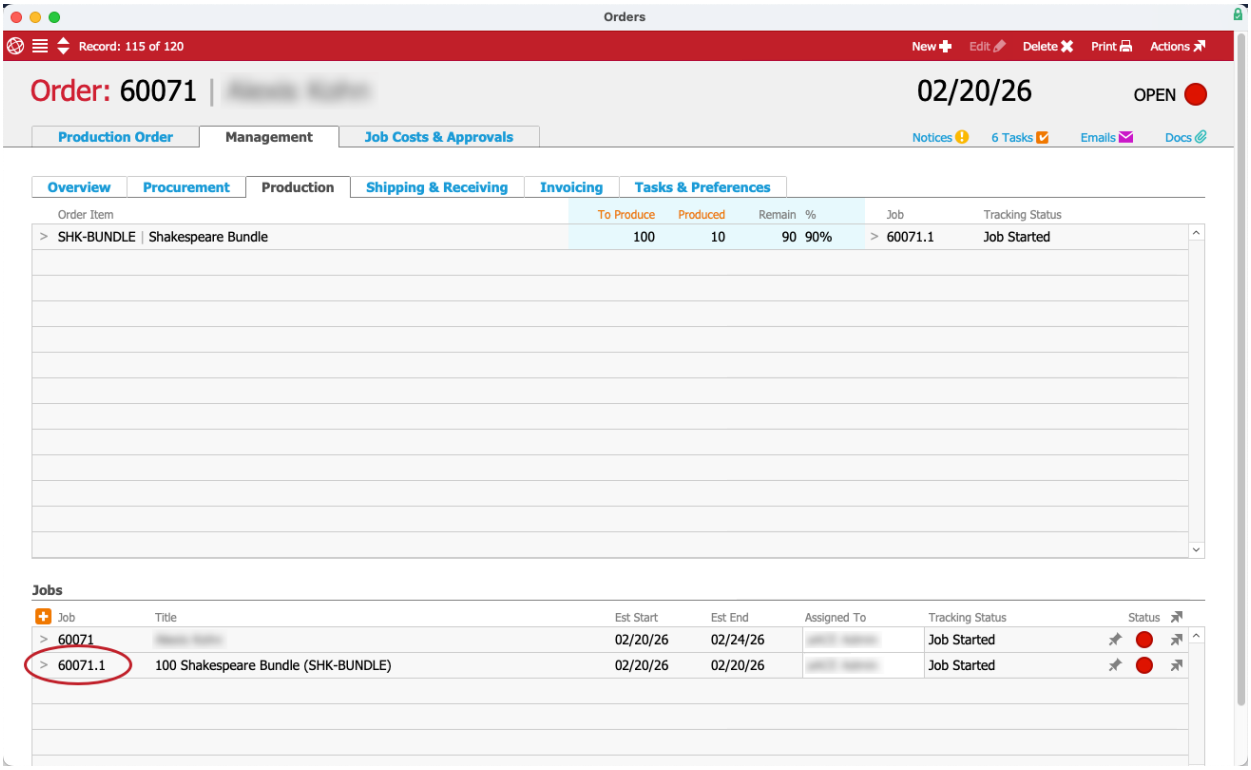
aACE uses hierarchical numbering in specific modules such as Jobs and Projects.

aACE cannot use hierarchical IDs for modules where voiding a record means creating a reversal records because there could be two records with the same ID. For example, if the invoice IDs were Invoice #60098-1 and Invoice #60098-2, and the user voids Invoice #60098-1, then the reversal would be Invoice #60098-3. Then, if the ID for Invoice #60098-1 is used again, there are two records in aACE with the same record ID: the new record and the previously voided record. The order would be missing Invoice #60098-1 and there would be duplicate record IDs, causing both customer and internal team confusion.

Invoices, Receipts, Purchases, and Disbursements do not use hierarchical IDs to prevent this duplicate ID situation.

Record Hierarchy in Jobs

For records with a parent-child relationship, the ID numbering includes suffixes. The child records inherit the ID from the parent record, but then include additional suffixes. For example, suppose you have a production order with the ID of order #60071. For an order job belonging to the order, the order job inherits the same ID: job #60071. However, when any new child production jobs for the order job are generated, the child job IDs are job #60071.1, job #60071.2, and so on.



In examples where a child production job has child records (e.g. part of the bill of materials needed for assembly requires an assembly item with its own bill of materials), an additional suffix is added after the parent production job ID. For example, if the parent production job ID is job #60071.1, then its first child production job ID is job #60071.1.1, then second is job #60071.1.2, and so on.

Record Hierarchy for Projects

aACE uses record hierarchy for orders within a project. When an order is created as a child of a project, the order inherits the Project ID with a suffix. For example, if the project ID is project #60150, then child orders would be order #60150.1, order #60150.2, and so on.

An order may be created as a standalone order with its own unique ID and then attached to a project using the Update Project admin actions option. The order might have its ID updated so that it is treated as a child of the project.