Binary Stream

Microsoft Dynamics 365
Business Central

Multi-Entity
Management

CUSTOMER STORY

One remittance, many ledgers.

A manufacturing group of several legal entities, some of them acquired, shares one cash account and one customer master. Customers pay a batch of invoices with a single remittance, unaware that several entities issued them. Multi-Entity Management for Microsoft Dynamics 365 Business Central creates the balancing between entities on each receipt, as it posts.

At a glance.

INDUSTRY

Manufacturing — a group of several legal entities, a number of them acquired companies kept separate.

ERP JOURNEY

Moving its financial ERP to Microsoft Dynamics 365 Business Central. Manufacturing and most of order-to-cash run in a separate operational system.

ENTITY STRUCTURE

All entities in one ledger, sharing one customer master, one vendor master, and one corporate cash account.

TRANSACTION PROFILE

Invoices created in the operational system and fed in by integration. Customers settle invoices from several entities with one remittance, and the entities supply one another.

PRESCRIBED SOLUTION

Multi-Entity Management (MEM) for Business Central.

ENGAGEMENT STAGE

Pre-sale evaluation. Nothing has been measured, and no figure is a result.

Why one payment touches several ledgers.

This manufacturing group is made up of several legal entities, a number of them acquired companies that were kept separate. Manufacturing and most of order-to-cash run in an operational system. The financial ERP is deliberately narrow: general ledger, payables, receivables, and banking.

> **Every entity shares one customer master, one vendor master, and one corporate cash account. Centralization was settled long ago.**

- Cash is collected centrally, in one corporate account.
- Invoices are issued by the entity that did the work.
- The customer neither knows nor cares which entity billed it.
- Acquired companies keep their legal identities while customers consolidate payments.

The cash moves once. The accounting has to move once for every entity the payment touches.

So in practice, customers stop seeing the entities. A large account settles a batch of invoices with one remittance, and the bank deposits it without knowing that several entities issued them. The real question was whether Business Central would balance that reliably.

> **The way this was designed or architected 20 CFOs ago was centralized customer master, vendor master, AR, AP, GL.**
>
> EVALUATION LEAD, MULTI-ENTITY MANUFACTURING GROUP.

Three things finance is fixing after the fact.

Balancing that happens only sometimes.	Customer payments land in the corporate cash account, and one payment can settle invoices from several entities. Matching isn't the problem—the receivables team is comfortable selecting invoices by hand. The failure comes after. Once a cross-entity receipt posts, the balancing entries between the entity holding the cash and each invoicing entity are created some of the time, and not the rest. Whatever is missed becomes a true-up journal at month end.
One receipt, two places to balance.	Two things can differ at once: the entity a receipt is recorded under, and the entity that issued the invoice it pays. When both differ from the entity that owns the bank account, the books need balancing on two legs. • Owns the bank account: Entity A. • Records the receipt: Entity B. • Issued the invoice: Entity C. A process that handles only one leg leaves the other for someone to find at month end.
Trade between entities lives in journal entries.	The entities supply one another. When one can't get material from its outside vendor, another provides it, and the transfer is recorded as inter-entity activity. Today that happens by journal entry, often gathered into one large entry at month end. The evaluation lead called it "probably the second biggest pain point," and asked for something a journal entry can't give: a formal record on each side of the trade.

Keep one ledger. Let the posting do the balancing.

The group never needed convincing to centralize. Every entity lives in one Business Central company, with the entity held on one of the platform's two global dimensions, so it appears on every ledger entry and can filter every report.

Behind each entity value is a full record: legal name, logo for its documents, primary bank account, its own document numbering where wanted, and its own approvers. The chart of accounts is shared, so any account used by more than one entity carries one common number.

Because MEM sits inside Business Central, the group's existing integrations largely carry on as they are, since they were already built to carry the entity. Turning the bank's file into receipt lines remains integration work, owned by the implementation.

Centralized receivables —

solves balancing that happens only sometimes.

A receipt is entered in the cash receipt journal against the central bank account and applied to open invoices, including invoices other entities issued. On posting, a due-to and due-from pair is generated for each applied invoice whose entity differs from the one holding the cash. The clerk's only job is to pick the right invoices—each invoice already knows its entity.

Both legs, one posting —

solves one receipt, two places to balance.

This was tested directly: a receipt recorded under one entity, through a bank account owned by a second, applied to an invoice issued by a third. The posting preview showed two sets of balancing entries—one for the cash and one for the application. The application leg, the one this group's month end depends on, is generated by the posting itself rather than reconstructed afterward.

Trade as documents —

replaces journal entries for inter-entity trade.

Each entity can be represented internally as a vendor and a customer, linked to the entity it stands for. When a purchase order to an internal vendor is received and invoiced, the system offers to create the matching sales order in the counterpart entity, mirrored line for line. Each side of the trade gets its own document. This works from orders, not standalone invoices.

Centralized payables —

one run, one bank account, every entity.

Centralized payment is set per payment journal batch. A suggested payment run draws open invoices from across the entities into one run against the central bank account, and posting generates due-to and due-from entries for every invoice owned by another entity. Purchase invoices with lines charged to several entities balance on posting too, and recurring splits can be saved as allocation templates.

Control —

one company, without sharing everything.

Customers, vendors, items, locations, and bank accounts can each be restricted to the entities allowed to use them, so sharing one company doesn't mean every entity sees every record.

What's changing for finance across the group.

01

Every cross-entity receipt creates its own balancing when it posts, so month end no longer starts with true-up journals.

02

When a receipt is recorded in one entity, banked in another, and pays a third's invoice, both sets of balancing come from the same posting.

03

The receivables team keeps selecting invoices as it does today. Each invoice already knows its entity, so the balancing follows on its own.

04

Trade between entities is recorded as a purchase order on one side and a matching sales order on the other, not a month-end journal entry.

05

One payment run pays invoices across every entity from the corporate account, with the balancing created for each invoice.

06

Every entity shares one company, but each customer, vendor, and bank account is visible only to the entities allowed to use it.

Binary Stream

sales@binarystream.com | +1 604 522 6300

binarystream.com