

Formula Builder: Build & Explain

Run your billing platform by describing what you want.



Describe the calculation to build it. Ask about any formula to understand it. Both in plain language.

The Problem

Formulas drive the numbers that matter — derived fees, prorations, tier and threshold logic, eligibility conditions, the values written by automated processes. They are also where billing gets quietly wrong. A formula has to navigate the right relationships between entities, use the right functions, and aggregate the right records. Built by hand, that work needs someone fluent in both the business logic and the expression syntax. Once it exists, the formula becomes a string almost no one can read, so no one is sure what it computes or whether it is still correct. BillingPlatform AI closes both ends of that gap.

Build — a formula from a description

Describe the calculation in plain language — for example, “a 1.5% late fee on the unpaid balance of any invoice more than 30 days past due.” BillingPlatform AI builds the formula expression, navigating the real relationships between your entities — invoice to billing profile to account — rather than guessing at field names. It works against a semantic model of your data, so it builds against the relationships and fields that actually exist. The expression is syntax-validated before it can be saved.

Explain — plain language from any formula

Point at any existing formula field — one you built, one you inherited, one that has gone unquestioned for years — and BillingPlatform AI describes, in clear language, what it does:



What the formula calculates and the value it returns



The relationships it navigates to reach that data



Which fields and which entities it draws from



The conditions and aggregations applied along the way

An opaque expression becomes a sentence anyone on the team can read — turning audit questions, handoffs, and “why is this number what it is?” into a conversation instead of a forensic exercise.

Validated before it is relied on

Every generated formula is checked for valid syntax before it can be saved — an expression that will not run cannot be persisted. Syntax validity is not the same as business correctness, so BillingPlatform AI can also execute the formula against real records before you depend on it, showing the actual result on actual data. You confirm the number is right before it reaches a customer account — not after a reconciliation surfaces it.

Trust & governance

- **Permission-faithful** — every action runs as a real user under your existing role-based permissions, and every change is captured in the native audit trail — attributable and reversible. No elevated AI account, no separate access model to govern.
- **Validated before save** — generated formulas pass the same validation as hand-built ones. What will not run cannot be persisted.
- **Governed your way** — layer custom validations and approval workflows on top wherever your process requires explicit sign-off. You decide how much governance each change carries.

Specifications

- ✓ **Where formulas run**
Computed entity fields (evaluated per record at read time) and field values within automated workflow actions
- ✓ **Function families**
Numeric, string, date and time, conditional logic, aggregation, and subselects — the full Oracle SQL function set
- ✓ **Cross-entity logic**
Navigates lookup relationships across entities; supports aggregation over related records and existence checks
- ✓ **Validation**
Automatic syntax validation before save; optional execution against live data to confirm the result before it is relied on
- ✓ **Foundation**
Built on the Model Context Protocol — open standard, no model lock-in, no parallel data plane
- ✓ **Tenant-aware**
Custom entities, fields, and relationships discovered via native metadata — no per-tenant integration build

Who it's for

Finance and RevOps teams who own the calculation logic but should not need an expression specialist for every change. Controllers and auditors who need to know what a formula computes without decoding it. Implementation and operations teams inheriting calculations they did not write.

See it on your own workflows. Talk to your BillingPlatform account team to request a demo — or point your AI assistant at your sandbox tenant and try it yourself.

