

Increase Service Gross Profit with Smarter Parts Pricing



Your Monday Morning Outcome

By following this guide, you will have your pricing matrices built, price codes configured, and default assignments applied – so every part transaction prices itself correctly, at the margin you set. When costs change, your margins don't. The formula adjusts the selling price automatically. Set it up once. Let it work.

Before You Start

Pre-Implementation Checklist

✔ Permissions confirmed

Verify the following permissions are enabled for your role before proceeding. Navigate to App Grid → Settings → Core Settings → Roles to confirm. If you cannot access Roles, contact your system administrator. Pricing Permissions: Price Code Override (override price code at document level on SO/RO); Selling Price Override (override selling price at part line level on RO and SO); Update Cost Price (update part cost when selling on RO or SO); Operation Price Edit (edit opcode-level prices on RO job lines – controls static opcode pricing, not preferred but must be visible). Setup Module Access: Price Breaks & Formulas View + Edit; Customized Price Setup View + Edit; Price Code View + Edit; Default Parts Pricing Create; Parts Settings Create. Report Access: Parts Reports View All; Parts Performance Report View.

✔ Existing pricing reviewed

Before building new price codes, review what is already configured in ARC. Navigate to App Grid → Settings → Parts Settings → Price Codes to see active price codes. Knowing what exists prevents duplication and gives you a baseline to compare against after configuration.

✔ Source codes identified

Know your source codes before building formulas. Each source can carry a different pricing formula. Navigate to App Grid → Settings → Parts Settings → Source Code and list your active sources. You will assign formulas by source in Steps 1 and 2.

✔ Baseline captured

Pull the Price Override Report, Gross Profit by Source report, and Parts Performance Report now, before making any changes. These are your before numbers. Navigate to App Grid → Parts → Reports. You need a before to measure what changes after configuration.



Build Price Breaks and Formulas

Time: 20 minutes | **Navigation:** App Grid > Settings > Parts Settings > Price Breaks and Formulas

Setting	Configuration
Formula Name	Create a descriptive name (e.g., “Standard Retail Matrix”). This name becomes selectable in Customized Price Setup. Avoid generic names – use names that immediately identify the pricing strategy.
Base Variable	Select the calculation base. Cost is the most common choice – the formula recalculates the selling price each time cost changes, so your margin holds automatically. Options: Cost, List, Trade, Comp.
Price Breaks (Tiers)	Add one or more tiers: enter a Start Amount, End Amount, and Formula per range. Example: \$0.01-\$50.00 → Cost + 30%; \$50.01-\$200.00 → Cost + 20%; \$200.01+ → Cost + 15%. No negative values. Two decimal precision.
Test Before Saving	Use the built-in price calculator before saving. Enter a sample cost and verify the selling price matches your target margin. Resolve errors here, before the formula is applied to transactions.

Create Customized Prices

Time: 20 minutes | **Navigation:** App Grid > Settings > Parts Settings > Customized Price Setup

Setting	Configuration
Price Name	Assign a clear, descriptive name (e.g., “Retail Pricing,” “Wholesale Pricing”). This name appears as a selectable option in Price Codes when assigning a Regular Price per source. Make it immediately recognizable.
Source Code Assignment	Assign your pricing formula to each relevant source code. All sources follow the default formula unless individually configured as exceptions. Only separate a source if it genuinely needs a different pricing structure. Duplicating sources unnecessarily creates maintenance overhead.
Price Formula per Source	Link the formula you built in Step 1 to each source. If all your sources use the same markup structure, assign the same formula to all. If aftermarket sources need a different margin than OEM sources, create separate formulas and assign accordingly.

Set Up Price Codes

Time: 20 minutes | **Navigation:** App Grid > Settings > Parts Settings > Price Codes

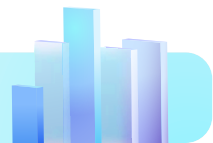
Setting	Configuration
Price Code Identifier	Assign a clear, descriptive name (e.g., “Retail Pricing,” “Wholesale Pricing”). This name appears as a selectable option in Price Codes when assigning a Regular Price per source. Make it immediately recognizable.
Regular Pricing Formula	Assign the Customized Price you built in Step 2. This is what ARC uses to calculate the selling price for parts sold under this price code. The formula runs against current cost every time – no manual updates required when costs change.
Customer and Sale Type Defaults	A customer can have a default price code for Sales Orders and a separate default for Customer Pay ROs. Set these under Customer Management → Customer Detail → Parts section (SO) and Service section (RO). When a part hits an RO, ARC checks: pay type → customer price code → default RO pricing.
Minimum GP% per Price Code	If using Price Code Level GP Validation, set the minimum gross profit percentage for each price code independently. Example: Retail at 25%, Wholesale at 10%, Employee at 5%. Individual values override the global default and cannot be overwritten by it once set.

Apply Default Price Codes

Time: 15 minutes | **Navigation:** App Grid > Settings > Parts Settings > Price Codes

Sale / Pay Type	Configuration
Sales Order – Retail	Assign the price code that applies when parts are sold over the counter to retail customers.
Sales Order – Wholesale	Assign the price code for wholesale account sales. Wholesale margins differ from retail – a separate price code ensures the correct markup applies automatically without advisor input.
Sales Order – Internal	Assign the price code for internal parts sales (shop use, demos, loaner vehicles). Typically at or near cost per dealer policy. Keeping this separate protects your retail GP reporting from being diluted by internal transactions.
Repair Order – Customer Pay	Assign the price code for customer-pay RO jobs. This is your most common RO pay type. The price code maps directly to the formula you built, so parts price themselves when added to a job line.

Sale / Pay Type	Configuration
Repair Order – Warranty	Assign the price code for warranty RO jobs. Warranty pricing often differs from customer pay. Note: if a selling price was manually overridden on a part line, it will not change when the job type switches to warranty – this is expected system behavior.
Repair Order – Internal	Assign the price code for internal RO jobs (shop vehicles, demos, loaner fleet). Price at cost or near-cost per your dealer policy to protect retail GP reporting from being diluted by internal work.



Measure Your Results

Report	Navigation	What to Look For
Price Override Report	App Grid → Parts → Reports → Price Override	Every manual price change logged by user, date, original price, and new price. This is your direct proof the pricing structure is holding. A high override rate means permissions need tightening or the formula needs adjustment.
Gross Profit by Source	App Grid → Parts → Reports → Gross Profit	Margin performance across every source code. Compare against your pre-implementation baseline. Any source showing lower GP than expected points to a formula assignment that needs review.

Report	Navigation	What to Look For
Parts Performance Report	App Grid → Parts → Reports → Parts Performance	Comprehensive parts department performance – sales, gross profit, and fill rates. Your holistic view of whether the pricing structure is delivering results.
Detailed Sales Report	App Grid → Parts → Reports → Detailed Sales	Line-level pricing detail – see exactly what each part sold for, at what margin, on which transaction. Use this to spot individual lines where pricing deviated from the matrix.
Employee Productivity Report	App Grid → Parts → Reports → Employee Productivity	Sales per employee and productivity metrics. Identifies which team members are generating price overrides and at what frequency.
In & Out Sale Report	App Grid → Parts → Reports → In & Out Sale	All non-stock (in-and-out) sales and their margins. Non-inventory parts often have different margin profiles – this report catches any pricing gaps there.



Service-Side Reports

Report	Navigation	What to Look For
Service Summary Report	App Grid → Service → Reports → Service Summary Report	Overview of service department performance including parts dollars per RO. Shows the impact of parts pricing on overall RO revenue.
Repair Order Detail Report	App Grid → Service → Reports → Repair Order Detail Report	Comprehensive RO breakdown including labor and part prices per job line. Use this to audit individual ROs for pricing accuracy.
Service Advisor Coupon Report	App Grid → Service → Reports → Service Advisor Coupon	Discount and coupon usage by advisor. Shows the GP impact of discounts applied at the advisor level.