Feedback of Labor Operations Model

1. Parts, Miscellaneous Items, and Fees

Concern: Dealers need to attach Parts (Oil, Filters) and Miscellaneous items (Hazardous Waste, Tire Tax) to a Labor Operation.

Resolution: `BillOfMaterialReference` (BOM) and the `FinancialSplit`.

- **Parts:** Managed via the `BillOfMaterialReferences` list on the `LaborOperation`. Each entry links to specific `PartIdentifiers`.
- **Fees/Taxes:** The `FinancialSplit` includes `feeReferences` and `taxSplits`. Items like "Hazardous Waste Disposal" are distinct financial line items rather than text notes.
- **Quantities:** `FeeReference` and `DiscountReference` include a `unitCount` attribute to support multipliers (e.g., charging a disposal fee 4 times for a set of tires).

2. Bundles and Menu Pricing

Concern: Labor Operations are often used in Packages or Menu Items.

Resolution: The model handles bundles through a combination of hierarchy and metadata:

- **Hierarchy:** The `parentReferenceId` allows individual Labor Ops to be grouped under a single "Master Package."
- **Logic:** The `PriceTypes` enum includes a `BUNDLE` designation. When set, the system knows to look at the package-level price rather than the sum of individual labor lines.

3. Selling Price vs. Dealer Cost

Concern: The model needs to track what the customer pays versus the dealer's internal cost.

Resolution: The `Price` include comprehensive financial fields:

- `unitCostAmount` and `totalCostAmount`: Tracks the dealer's expense for parts or sublet labor.
- `totalAmount`: Represents the final selling price to the customer or manufacturer.
- `taxIncludedIndicator`: A boolean to clarify if the displayed price is inclusive of taxes.

4. Multi-Payer Splits (Partial Warranty/Goodwill)

Concern: Some systems require splitting a single job across multiple payment types (e.g., Warranty pays for parts, Customer pays for labor).

Resolution: `payType` is part of `FinancialSplit` and `Price`.

- **Granular Control:** You can assign a `capturedFinancialSplit` at the **Line Item level** Bill Of Material (BOM) as well as the **Header level** (Labor Operation).
- **Example:** A single Repair Order line can now have a "Warranty" pay type for the transmission part, but a "Customer Pay" type for the labor hours.

5. OEM vs. Dealer Designations

Concern: Distinction between factory-mandated opcodes and local dealer-created codes.

Resolution: `originType` attribute on `LaborOperation`, utilizing the `PayeeTypes` enum. This allows you to explicitly flag an operation as `OEM`, `DEALER`, or `THIRD_PARTY_PROVIDER`.

How the Multi-Payer Split is Addressed

In a "Partial Warranty" or "Goodwill" scenario, a single job isn't always paid by one person. The model supports this through **delegated financial responsibility**:

- **The Scenario:** A customer has a failed Water Pump. The Manufacturer agrees to pay for the Part (Warranty), but the Customer must pay for the Labor (Customer Pay) because they are past the mileage limit.
- **The Model Solution:**

1. **LaborOperation Level:** The `LaborOperation` references a `FinancialSplit` where `payType = CUSTOMER_PAY`. This covers the 2.0 hours of labor.
2. **Line Item Level:** The `BillOfMaterialReference` for the "Water Pump" part now carries its own `capturedFinancialSplit`. In *this* specific split, `payType = WARRANTY`.
3. **The Result:** The DMS can now generate two distinct accounting entries (and potentially two separate invoice statements) from a single `LaborOperation`.

Concern	Status	How it's handled
Parts & Misc Quantities	**Good**	`unitCount` in `FeeReference/DiscountReference` and the `BillOfMaterialReference` list.
Selling vs. Cost Price	**Good**	`Price` contains `unitCostAmount`, `totalCostAmount`, and `totalAmount`.
Pay Type Logic	**Good**	`payType` is a property of the `FinancialSplit` and `Price`, allowing for tiered pricing.
Package/Bundle Ties	**Good**	Handled via `parentReferenceId` and `PriceTypes.BUNDLE`.
OEM vs. Dealer Codes	**Good**	`originType: PayeeTypes` distinguishes between factory and local opcodes.

Json Examples of complex concerns: Multiple Payers, Miscellaneous Fees, and Package Bundling.

Example 1: The "Split-Payer" Scenario

Addresses the concern where a single operation has different payers for parts and labor (e.g., a "Goodwill" or "Partial Warranty" repair).

```
{
  "labor_operation_id": "OP-7721",
  "name": "Water Pump Replacement",
  "pay_type": "WARRANTY",
  "origin_type": "OEM",
  "captured_financial_split": {
    "price_name": "Labor Portion (Goodwill)",
    "pay_type": "DEALER",
    "price": {
      "total_amount": "0.00",
      "total_cost_amount": "150.00",
      "price_description": "Dealer covers labor cost for customer satisfaction"
    }
  },
  "bill_of_material_references": [
    {
      "bill_of_material_key": "PART-992",
      "context": "Water Pump Assembly",
      "captured_financial_split": {
        "pay_type": "WARRANTY",
        "price": {
          "total_amount": "245.50",
          "unit_cost_amount": "180.00",
          "price_description": "Manufacturer covers part cost"
        }
      }
    }
  ]
}
```

Example 2: Miscellaneous Fees & Multi-Quantity Taxes

This addresses the concern regarding **Tire Taxes**, **Hazardous Waste**, and **Shop Supplies** where quantities (unit counts) vary.

```
{
  "labor_operation_id": "TIRE-004",
  "name": "Mount & Balance - 4 Tires",
  "captured_financial_split": {
    "price_name": "Retail Tire Service",
    "pay_type": "CUSTOMER_PAY",
    "fee_references": [
      {
        "fee_key": "HAZ-WASTE",
        "unit_count": "1",
        "context": "Shop Supplies / Disposal Fee"
      },
      {
        "fee_key": "STATE-TIRE-TAX",
        "unit_count": "4",
        "context": "State mandated disposal tax per tire"
      }
    ],
    "tax_splits": [
      {
        "tax_code": "SALES-TAX",
        "tax_rate": "0.07",
        "jurisdiction": "County"
      }
    ]
  }
}
```

Example 3: The "Menu/Package" Bundle

This illustrates how a **Package Code** is tied to a Labor Operation, and how the price can be represented as a flat "Bundle" price rather than an itemized sum.

```
{
  "labor_operation_id": "OIL-CHG-SYN",
  "parent_reference_id": "PKG-30K-SERVICE",
  "labor_operation_description": "Synthetic Oil Change included in 30k Mile Package",
  "captured_financial_split": {
    "price_type": "BUNDLE",
    "price_name": "30,000 Mile Menu Item",
    "price": {
      "display_value": "$499.00",
      "total_amount": "499.00",
      "price_description": "Flat rate bundle price for 30k mile service"
    }
  },
  "bill_of_material_references": [
    {
      "bill_of_material_key": "OIL-FILTER-01",
      "context": "OEM Filter"
    },
    {
      "bill_of_material_key": "SYN-OIL-5W30",
      "context": "Synthetic Oil",
      "unit_count": "6"
    }
  ]
}
```

Why these address the customer's concerns:

1. **Multiple Splits:** By allowing a `capturedFinancialSplit` at both the operation and the BOM level, we handle cases where the payer for the part is different from the payer for the labor.
2. **Quantities for Misc:** The `unitCount` in the `feeReferences` handles the "multiple wheel weights" or "4x tire disposal" issue.
3. **Cost vs. Price:** Every `Price` object now explicitly separates `totalCostAmount` (Dealer Cost) from `totalAmount` (Selling Price).
4. **Package Logic:** The `priceType: BUNDLE` and `parentReferenceId` allow the DMS to link specific ops back to a larger Menu Item.