TOOL NAME
Ship From Store Route Economics Calculator

INPUTS
market
vehicle_type
miles_to_hub_or_spoke
avg_routing_time_per_stop_minutes
default_service_time_minutes
max_driver_time_minutes
avg_speed_mph
default_avg_cubic_inches_per_package

STORES_UPLOAD_TABLE
route_id
anchor_id
stop_type # "Anchor" or "Satellite"
store_name
address
city
state
zip
packages
avg_cubic_inches_per_package
pickup_window_start_time
pickup_window_end_time
service_time_minutes

RATING_TABLE
vehicle_type
base_fee
per_mile_rate
per_stop_rate

RATING_VALUES
26' Box Truck,175,2.20,50
Cargo Van,95,1.50,20

CORE_LOGIC
```
import math

# Rating lookup
if vehicle_type == "Cargo Van":
    base_fee = 95
```

```
    cost_per_mile = 1.50
    stop_fee = 20
    vehicle_cubic_capacity = 650000
else:
    base_fee = 175
    cost_per_mile = 2.20
    stop_fee = 50
    vehicle_cubic_capacity = 2700000

results = []

for anchor_id in unique_anchor_ids:

    stops = [s for s in stores if s["anchor_id"] == anchor_id]
    anchor_stops = [s for s in stops if s["stop_type"] == "Anchor"]
    satellite_stops = [s for s in stops if s["stop_type"] == "Satellite"]

    anchor_packages = sum(s["packages"] for s in anchor_stops)
    satellite_packages = sum(s["packages"] for s in satellite_stops)
    total_packages = anchor_packages + satellite_packages
    total_stops = len(stops)

    total_cube = 0
    for s in stops:
        cube_per_pkg = s["avg_cubic_inches_per_package"] if
s["avg_cubic_inches_per_package"] else default_avg_cubic_inches_per_package
        total_cube += s["packages"] * cube_per_pkg

    vehicles_by_cube = math.ceil(total_cube / vehicle_cubic_capacity)

    latest_start = max(s["pickup_window_start_time"] for s in stops)
    earliest_end = min(s["pickup_window_end_time"] for s in stops)
    pickup_overlap_minutes = max(0, (earliest_end - latest_start).total_seconds() / 60)

    service_minutes = 0
    for s in stops:
        service_minutes += s["service_time_minutes"] if s["service_time_minutes"] else
default_service_time_minutes

    routing_minutes = total_stops * avg_routing_time_per_stop_minutes
    pickup_minutes_required = service_minutes + routing_minutes

    hub_travel_minutes = (miles_to_hub_or_spoke / avg_speed_mph) * 60
    total_minutes_required = pickup_minutes_required + hub_travel_minutes
```

```
    drivers_by_time = math.ceil(total_minutes_required / max_driver_time_minutes)
    drivers_required = max(vehicles_by_cube, drivers_by_time)

    window_feasible = pickup_overlap_minutes >= pickup_minutes_required

    total_route_miles = miles_to_hub_or_spoke
    distance_cost = cost_per_mile * total_route_miles
    stop_cost = stop_fee * len(satellite_stops)

    anchor_route_cost = base_fee + distance_cost
    anchor_cpp = anchor_route_cost / anchor_packages if anchor_packages > 0 else 0

    blended_cost = anchor_route_cost + stop_cost
    blended_cpp = blended_cost / total_packages if total_packages > 0 else 0

    results.append({
        "anchor_id": anchor_id,
        "anchor_cpp": anchor_cpp,
        "blended_cpp": blended_cpp,
        "drivers_required": drivers_required,
        "vehicles_required_by_cube": vehicles_by_cube,
        "pickup_overlap_minutes": pickup_overlap_minutes,
        "pickup_minutes_required": pickup_minutes_required,
        "window_feasible": window_feasible,
        "total_packages": total_packages,
        "total_stops": total_stops
    })

OUTPUT_TEMPLATE
Ship From Store Route Economics

Market: {market}
Vehicle: {vehicle_type}
Anchor ID: {anchor_id}

Anchor CPP: ${anchor_cpp}
Blended CPP: ${blended_cpp}

Drivers Required: {drivers_required}
Vehicles Required by Cube: {vehicles_required_by_cube}

Pickup Window Overlap (mins): {pickup_overlap_minutes}
Pickup Minutes Required: {pickup_minutes_required}
```

Window Feasible: {window_feasible}

Total Packages: {total_packages}
Total Stops: {total_stops}

What this calculator answers

1. Total Route Cost

This tool calculates the **true all-in cost** to run a pickup route using a Cargo Van or 26’ Box Truck.

It includes:

- One base equipment fee per route
- Mileage cost for the full route to the hub
- Incremental stop fees for any additional pickup locations

It does **not** double count miles, base fees, or drivers when stops overlap.

The output labeled **Total Route Cost** represents the real cost to operate that route on a given day.

---

2. Discount from Overlapping Addresses

This calculator shows **how much cost per package drops when multiple pickup locations are combined onto the same route**.

Instead of applying an artificial discount, it models:

- Shared base cost across all packages
- Shared mileage across all packages
- Only incremental cost for added stops

The difference between:

- **Anchor CPP** (cost per package if run alone)
- **Blended CPP** (cost per package with overlapping pickups)

…is the **effective discount created by route density**.

This mirrors how transportation actually works in the real world.

How customers should use this tool

Step 1: Upload your stores

Provide a list of pickup locations including:

- Store address
- Daily package volume
- Average package size in cubic inches
- Pickup window start time
- Pickup window end time

These can be existing customer stores or candidate nearby locations.

Step 2: Select equipment

Choose whether the route runs on:

- Cargo Van
- 26’ Box Truck

The calculator automatically applies the correct:

- Base fee

- Per mile rate
- Per stop rate
- Vehicle capacity constraints

Step 3: Review feasibility

The calculator checks:

- Whether pickup windows overlap enough to service all stops
- Whether package volume exceeds vehicle capacity
- Whether total route time fits within a driver's shift

If any constraint fails, the route is flagged as not feasible.

Step 4: Compare economics

You'll see:

- Total Route Cost
- Anchor Only Cost per Package
- Blended Cost per Package with overlapping pickups
- Effective savings created by route density

This lets you quantify **how much cheaper each package becomes when routes are combined**.

What this is designed for

This calculator is designed to:

- Evaluate ship-from-store and local pickup density
- Identify which stores should act as anchors
- Show when adding nearby pickups lowers cost instead of raising it
- Support pricing, network design, and rollout decisions

No integrations are required.
No routing optimization is assumed.
All results are grounded in real vehicle, driver, and time constraints.