

How do I add a custom amount resolver?

The cart discount implementations use the `\Aero\Cart\Traits\DeterminesAmount` trait, which allows you to set a custom amount resolver for a discount type

For example you can set a limit for percentage discounts:

```
\Aero\Cart\PercentageDiscount::resolveAmount(function (\Aero\Cart\CartDiscount $discount,
float $percentage, $items) {
    // Limit in pence, e.g. 100 = £1
    $limit = $discount->model->additional('percentage_discount_limit');

    if (! $limit) {
        return $percentage;
    }

    $itemsTotal = $items->sum(function (\Aero\Cart\CartItem $item) use ($percentage) {
        return $item->total()->inc()->value;
    });

    $discountTotal = $itemsTotal * ($percentage / 100);

    return $discountTotal > $limit
        ? ($limit / $itemsTotal) * 100
        : $percentage;
});
```

By default if multiple amounts are resolved for a discount by resolvers (e.g. if you had another custom resolver for percentage discounts) the largest amount is resolved. This can be changed to instead prioritise the smallest amount:

```
\Aero\Cart\PercentageDiscount::giveMinResolvedAmount();
```

In fact, you can even take full control over which amount would be resolved. For example, you could resolve the largest amount for customers but the lowest amount for guests:

```
\Aero\Cart\PercentageDiscount::determineResolvedAmount(function(array $amounts) {  
    $customer = \Aero\Cart\Facades\Cart::customer();  
  
    return $customer ? max($amounts) : min($amounts);  
});
```

Revision #1

Created 2026-09-18 11:12:48 UTC by Ethan Payne

Updated 2026-09-18 11:38:28 UTC by Ethan Payne