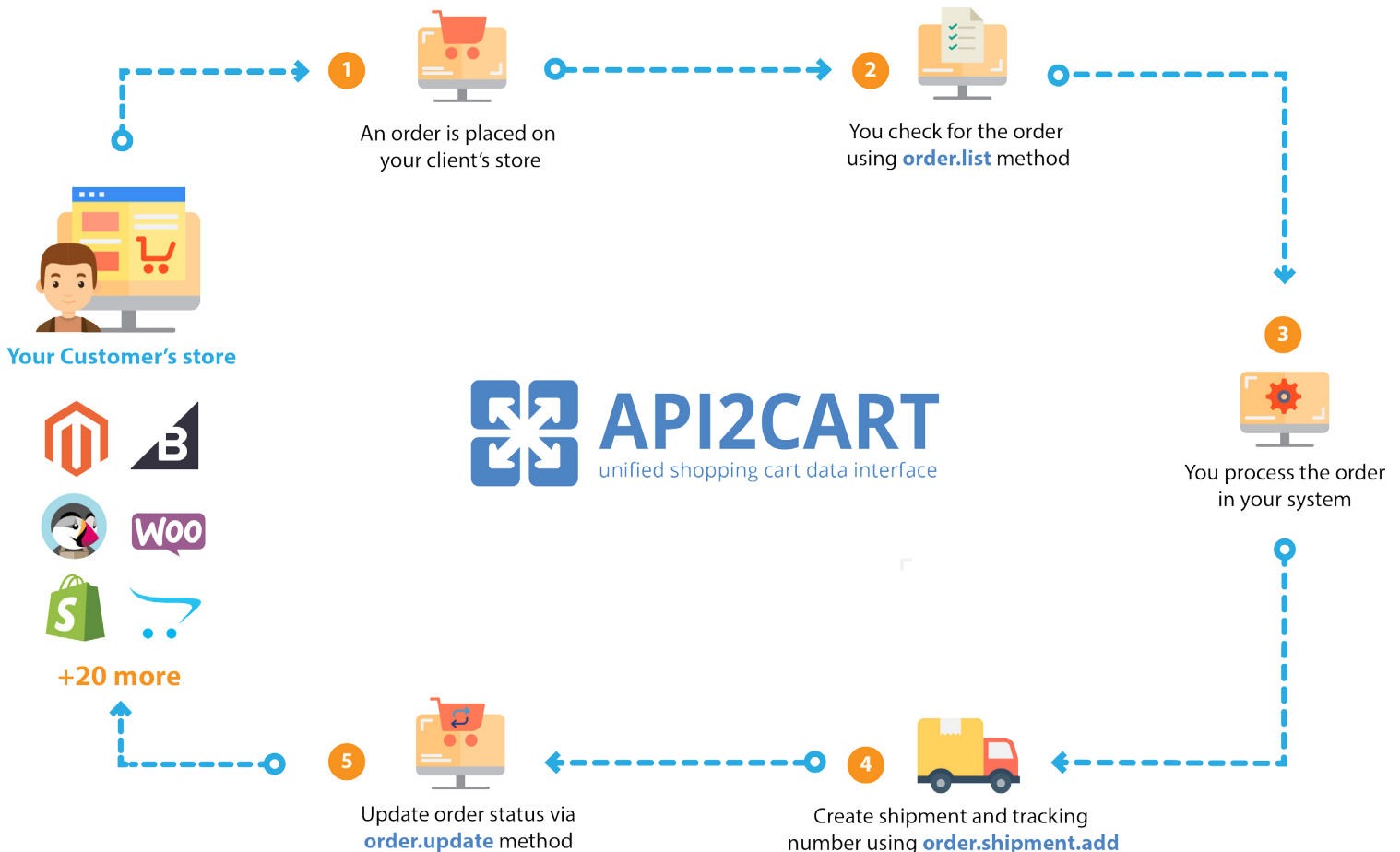# Shopping Cart Integration for Shipping Software

Unified approach

API2Cart-Shipping Management Software Workflow

Below is the outline of API2Cart-Shipping Management Software workflow variant:

Your Customer's store

+20 more

1 An order is placed on your client's store

2 You check for the order using **order.list** method

3 You process the order in your system

4 Create shipment and tracking number using **order.shipment.add**

5 Update order status via **order.update** method

API2CART
unified shopping cart data interface

Why Shipping Management Software Needs Integration with Shopping Platforms?

Shipping is definitely a challenging aspect for any eCommerce business. Besides picking, packing, and transportation, it includes many other operations like order fulfillment, shipment tracking, managing stock levels, updating order statuses. **Shipping software greatly depends on integration with shopping carts**, as carrying out key functions requires ability to access and process order, customer and product information from online stores.

No matter how terrifying this sounds, integrations are unavoidable. What is avoidable, however, is more integrations and the problems they entail.

API2Cart makes integration with multiple shopping carts and marketplaces so much easier, faster and cheaper. Also, API2Cart takes care of further multi-platform maintenance. You integrate with the API once, get connected to more than 35 e-commerce platforms, and don't worry about upgrading your code every time new versions of carts come out.

How Shopping Cart Integration Builds a Critical Link in the Chain

For you, as a shipping software provider, merchants' e-shop is a valuable asset because that is where your system gets information on orders and products to run their functions.

Quality of shipping software depends on its ability to automate retail operations, which in its turn, depends on its level and quality of integration with shopping carts. Info from stores powers the majority of features and processes that every shipping management system covers.

In particular, integration with customer stores via API2Cart enables the software to do the following:

- import orders from multiple sales channels
- update tracking info and order statuses
- create shipping labels
- automate, schedule, and customize shipping tasks to run only for specific stores
- organize the data in detailed reports

Quality of shipping software depends on its level and quality of integration with shopping carts. Integration means tying your application with a third party, so should something go wrong and an information loss or distortion occurs, and the whole lot of integrated processes will go on the blink. The consequence of this will be, that first your customers, and then you, will suffer losses.

The best way to integrate safely is to invest into a well-coined connection via API2Cart Unified Shopping Cart and Marketplace API. Having dealt with hundreds of integrations, our team have learned the pain points that various web and mobile applications face. With these in mind, **we crafted our unified API that lets your product seamlessly communicate with clients' stores, be they based on Shopify, Magento, WooCommerce, or any other shopping cart or marketplace that we support**. Plus, API2Cart's API makes it equally easy to integrate shopping carts with both web and mobile applications.

API2Cart Methods for Shipping Management Software

API2Cart has rich capabilities for shipping management software. We provide the whole scope of functionality to get and manage all needed info on products, orders, shipments, customers, stocklevels, etc. Let's dig deeper into the methods which help to perform the integration.

Inventory control

Access to accurate inventory data is vital for avoiding errors in shipping operations. Double check product availability by using product.info method, which returns items' quantity in stock. Once the order is placed your system can automatically update stock level of product using product.update method.

Import orders

There are two ways to learn about new orders on stores.
- ➤ Webhook for order.add event

First is setting the webhooks for order.add event to get notified every time the new order is placed. Using webhooks is much more beneficial way in comparison to polling an API, as it decreases the server load, saves api requests, and provides more up-to-date info.
- ➤ Order.list method

Another option for tracking new orders is executing order.list method once in a while to get list of products to be imported to your system. Filter orders by create time with *created_from* and *created_to* parameters.

Product data uploads

Use product.list method to get list of products along with all related info and images.
Get details on product variants via product.variant.info method.

Add shipment statuses

Automatically insert shipment tracking url or update tracking info into the ecommerce store by using order.shipment.tracking.add to seamlessly notify end customers.
Create shipments and add tracking numbers to orders with help of order.shipment.update and order.shipment.add methods.
Update order statuses via order.update method.

Manage customer info

Retrieve all customers details you need via customer.info and customer.list methods.

Key Challenges of In-house Integration (Multiply by the Number of Integrations)

The integration steps mentioned above imply software vendors looking to integrate with shopping carts and other related systems will have to deal with the following challenges:

1. Time.
On average, an integration takes 4-12 weeks to develop. In this time the responsible developer will learn the platform logic and architecture, put effort into reducing the friction between the product and the system, and very likely find themselves off base a couple of times.

2. Expenses.
Be ready to pay a set sum of a few thousands of dollars plus maybe extra if the need for additional development occurs. After the integration is complete and iterated to success, it will need developer work and attention because new versions will appear and this will mean upgrades to make. Plus, your clients might request modifications and custom enhancements.

3. Human resources.
Each integration is both tons of work to do and a separate thorn in the side that never goes away after the work is done, that someone has to deal with. For better and more reliable results, you will want the someone to be dependable and expert, and that is not easy to find and hard to allocate if they are a part of your product team.

How API2Cart Helps

Having dealt with hundreds of integrations, our team have learnt the pain points that various web and mobile applications face. With these in mind, we crafted our unified API that lets your product communicate with clients' stores, be they based on Shopify, Magento, WooCommerce, or any other shopping cart or marketplace that we support.

Speaking the language of benefits, API2Cart offers you to do the following:

➤ **Enhance your functionality faster**
Our unified API eliminates the need to develop separate integration for each platform. Integrate once, get the data required from the platforms needed.

➤ **Expand your market share dramatically**
Win more clients by having access to data from stores built on the shopping cart they use. Each integration means another thousands of users need you.

➤ **Reduce cost, save time**
You connect to dozens of platforms via one integration. It's months of expensive development work avoided.

➤ **Save resources**
Concentrate on what matters for your business. We take care of platform updates.

Speaking the language of tech, API2Cart lets you do the following:

1. Integrate your product with more than 30 shopping carts via one API
2. Add stores to join the 50k+ stores successfully connected by other clients of ours
3. Avoid the need to maintain each connection separately
4. Synchronize data from stores with your product real time via webhooks enabled
5. Escape mapping headaches

**Let us know if you have any questions.
We'll be happy to discuss them in any form you like.**

CONTACT US   **SCHEDULE A CALL**