City-Based Shipping Not Working

Troubleshooting · https://wpfactory.com/docs/city-location-based-shipping-not-working/

Problem description

If city-based rules in Conditional Shipping for WooCommerce: Restrict Shipping Options by Anything do not apply, first check the city names. Go to WPFactory > Conditional Shipping, open the Cities section, and put each city on its own line with no extra spaces. Then test with a real address and exclude checkout from caching and minification.

Common symptoms:

- Customers from excluded cities still see the restricted shipping method.
- The shipping method disappears even for cities you meant to allow.
- Behavior varies depending on the browser or how the address was entered.

Cause: the Cities condition relies on reading the customer's checkout city and on the checkout JavaScript re-evaluating the shipping methods when the city field changes. A mismatch in the city entered, or a caching or JavaScript conflict, can prevent the condition from evaluating correctly.

How city matching works

- Enter cities **one per line** in the Cities condition, not comma-separated.
- Matching is **case-insensitive**, so `London` and `london` are treated the same.
- The customer's city comes from the checkout city field. When the customer changes the city, the plugin re-checks the shipping methods through the WooCommerce `update_checkout` JavaScript event.
- Before an address is entered, the plugin falls back to the store base city, so what you see with an empty address reflects the base city rather than the customer's.

Solution, step by step

1. **Check the city names.** Go to **WPFactory > Conditional Shipping** (also reachable at **WooCommerce > Settings > Conditional Shipping**), open the Cities section, and confirm each city is on its own line with no extra spaces. Matching ignores case, so capitalization does not need to match, but spelling does.
2. **Test with a fresh checkout and a real address.** Clear the browser cache or use a private window, then fill in the shipping address manually so a real city is submitted rather than the store base city fallback.
3. **Confirm the checkout updates after the city changes.** The plugin re-evaluates shipping methods on the `update_checkout` event when the city field changes. Make sure no optimization or broken script prevents that event from firing.
4. **Exclude the checkout from caching and minification.** City conditions run at checkout, so exclude the cart and checkout pages from caching and minification to avoid interference.
5. **Test with a default theme and minimal plugins.** Temporarily switch to a default theme, disable other plugins, and test the city restriction to rule out a theme or plugin conflict.
6. **Enable Debug if it still fails.** Turn on **Debug** in the General section, then reproduce the checkout and read the log under **WooCommerce > Status > Logs** with source `conditional-shipping-for-woocommerce`. Debug logs the Cities condition, so it shows which city was read and how the methods were evaluated. Share the log with support if needed.

Prerequisites

- Admin access, and the ability to test the checkout.
- No heavy caching or minification on the checkout page.

Additional notes and prevention

- Keep a list of the accepted city spellings so your entries stay consistent as you add cities.
- Because matching is case-insensitive and per line, one clean line per city is enough. There is no separate ZIP or postcode condition, so location targeting is done by city.



CONDITIONAL SHIPPING

See every feature and the pricing on the Conditional Shipping plugin page.

View plugin →

Need a hand? Our team answers every ticket on the Conditional Shipping support forum.

Related articles

- Membership-Based Shipping Methods in WooCommerce
- Enable Debug Logging for Conditional Shipping
- Plugin and Theme Conflicts with Conditional Shipping
- Common Shipping Condition Setup Mistakes
- Fix Conditional Shipping After Site Migration