



# FedEx Web Services Coding Best Practices Guidelines

To efficiently integrate with FedEx Web Services, developers should follow these best practice guidelines:

1. Use current standard modern namespace aware XML parsers that comply with W3C standards. Do not try to use custom built XML parser. Be aware FedEx changes namespace prefixes.
2. The "Content-Type" in the HTTP POST should be 'text/xml'.
3. Avoid sending empty elements. Example:  
<Streetlines></Streetlines> or <StreetLines/>
4. Test transaction reply for missing elements before using data. When developing, determine how to react if a non-required reply element is not returned. For example, it is possible to ship a package if rating is not functional.
5. Only send the data necessary to process the request. Example: For a U.S. domestic shipment, avoid sending Commercial Invoice and commodity data that may only be required for international shipments.
6. Be sure the code allows for handling of a reply with Highest Severity other than successful. Warnings and notes are not indications of a failure; however, the error or warning message should be logged and examined for informative content.

The notifications are:

- **Success:** Your request was processed successfully.
- **Note:** Your request was processed successfully. The code/message contains additional information about how your request was fulfilled; you do not need to take any special action.
- **Warning:** Your request was processed successfully. The code/message explains what was done to fulfill the request; however, some assumptions may have been made to fulfill the request and you should determine if the reply matches intent. A warning message may also be a notification of a future change.
- **Error:** Unable to process request because either the request data was invalid or violated a business rule. Correct the request and try again.
- **Failure:** Request could not be handled at this time because of an internal FedEx error.



7. Be careful when repeating unsuccessful transactions. If a reply is returned with Highest Severity of Failure, resend the transaction to account for an intermittent environment problem. Limit the number of times a request is resent. If a reply is returned with Highest Severity of Error, this normally means there is a problem with the request and the request needs to be corrected before being resent.
8. Log and retain the XML request and reply.
9. The optional element CustomerTransactionID is available for the customer to provide a unique ID for logging purposes. This element makes it easier for the customer or FedEx to find an individual transaction for troubleshooting.
10. It is preferred that a developer instead of the end user contact FedEx for support. A support process between the product end user and responsible developer should be in place. Provide an account and meter number when calling technical support.

11. When processing a web services transaction, it is recommended to provide the account and meter number used to register for credentials to obtain a successful response. Mismatch of credential/account and meter number combinations will result in an Authentication Failure when attempting to process the transaction.
12. Only use UTF-8 character encoding. Unless needed, limit usage to 7 bit ASCII characters. Test usage of non 7 bit ASCII characters for desired behavior.
13. Use the fully qualified URL that is defined in the WSDL when setting the endpoint for your transactions. Failure to do so may lead to inconsistent results. Example: In the RateService WSDL the endpoint may be defined as:
 

```
<s1:address location="https://wsbeta.fedex.com:443/web-services/rate"/>
```

 In this case, use <https://wsbeta.fedex.com:443/web-services/rate> for the test endpoint or <https://ws.fedex.com:443/web-services/rate> for the production endpoint. Do not use <https://wsbeta.fedex.com:443/web-services/> or <https://ws.fedex.com:443/web-services/>.
3. For a special service to be included on a shipment, both the special service type and its detail must be included. Some special services do not have a detail. There may be no indication the special services are not included
4. The timestamp for a rate or shipment should be the time the package is expected to be tendered to FedEx or a FedEx agent. This is not necessarily the time at which the ship or rate transaction is performed. For example, a shipment generated late Friday night for a package that will not be picked up by FedEx until Monday should have a Monday timestamp. Use the correct timestamp if it is known, or the delivery estimation and rates may not be correct.
5. Perform the close for FedEx Ground and FedEx SmartPost® at the end of the shipping day before pickup. Perform the FedEx Ground close prior to the FedEx SmartPost close.

## Tracking

6. Limit the number of tracking numbers in a single track request to 30.
7. Limit the number of times a package is tracked to what is necessary for business needs.
8. For batch tracking, remove any packages that have returned a track status of “delivered” from the batch.

## Programming

1. In Visual Studio, updating the Track or Address Validation WSDL web reference will cause the application to fail when being compiled. The reference document (e.g., reference.vb or reference.cs) will need to be manually updated to resolve the problem. A typical error would be “CS0029: Cannot implicitly convert type 'fedexreq.ParsedElement' to 'fedexreq.ParsedElement[]'.” For more detail, see the Technical FAQs at [fedex.com/us/developer/faqs](http://fedex.com/us/developer/faqs).
2. If the ‘Specified’ property is not set to “true,” .NET does not pass the value of the corresponding field in the serialized SOAP request. Always set the “Specified” property where appropriate. Example:
  - request.RequestedShipment.Recipient.Address.ResidentialSpecified = true
  - request.RequestedShipment.Recipient.Address.Residential = true
3. Validate that required fields such as recipient postal code and package weight have data before sending the transaction. Validate that the data is appropriate for the field in question (e.g. for US postal codes, verify that the field is all numeric and is in the form of a 5-digit or a ZIP+4 postal code format). This will minimize transaction errors.
4. Do not allow new meter number generation without end user intervention.

## Rating/Shipping

1. Do not assume a particular service will be available for all scenarios. For example, STANDARD\_OVERNIGHT (among others) is not available between all postal codes.
2. If a specific service is being requested for rating, include the ServiceType in the request. This will decrease the size of the reply and reduce transaction response time.

## Address Validation

1. FedEx provides Address Validation as a suggestion and not a final determination. The end user needs to make a final determination of whether an address is usable from the data provided and their business needs. A process needs to be in place to handle addresses that cannot be validated so that orders can still be processed.
2. To ensure a better shipping experience, do not make the shipping process dependent on optional services such as Address Validation. For example, if Address Validation is unavailable at the time of order entry or shipping, a contingency should be in place in order to complete the shipment.

## Additional Resources

For more about FedEx Web Services, go to [fedex.com/us/developer/index.html](http://fedex.com/us/developer/index.html) to learn how to integrate FedEx into a website or application you’re developing. Learn how integration with existing business systems can make shipping around the world faster and easier. This business site will provide you with online developer resources along with feature and service information.

## Customer Support

If you have questions or need assistance, please contact the FedEx Technical Support Center at **1.877.FDX.Assist 1.877.339.2774**, Monday through Friday, 7 a.m. to 9 p.m. CST; and Saturday, 9 a.m. to 3 p.m. CST.