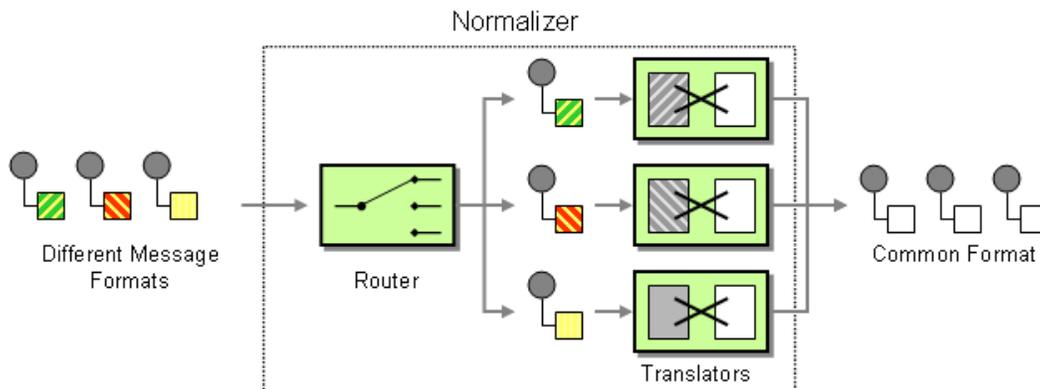


NORMALIZER

In a business-to-business (B2B) integration scenario it is quite common for an enterprise to receive messages from different business partners. These messages may have the same meaning, but follow different formats, depending on the partners' internal systems and preferences. For example, we built a solution for a pay-per-view provider that has to accept and process viewership information from over 1700 (!) affiliates, most of which did not conform to a standard format.

How do you process messages that are semantically equivalent, but arrive in a different format?



Use a *Normalizer* to route each message type through a custom *Message Translator* so that the resulting messages match a common format.

The *Normalizer* uses a *Message Router* to route the incoming message to the correct *Message Translator*. This requires the *Message Router* to detect the type of the incoming message. Many messaging systems equip each message with a type specifier field in the Message Header to make this type of task simple. However, in many B2B scenarios messages do not arrive as messages compliant with the enterprise's internal messaging system, but in diverse formats such as comma separated files or XML document without associated schema. While it is certainly best practice to equip any incoming data format with a type specifier we know all too well that the world is far from perfect. As a result, we need to think of more general ways to identify the format of the incoming message. One common way for schema-less XML documents is to use the name of the root element to assume the correct type. If multiple data formats use the same root element, you can use XPATH expressions to determine the existence of specific sub-nodes. Comma-separated files can require a little more creativity. Sometimes you can determine the type based on the number of fields and the type of the data (e.g. numeric vs. string). If the data arrives as files, the easiest way may be to use the file name or the file folder structure as a surrogate *Datatype Channel*. Each business partner can name the file with a unique naming convention. The *Message Router* can then use the file name to route the message to the appropriate *Message Translator*.

Source: <http://www.enterpriseintegrationpatterns.com/patterns/messaging/Normalizer.html>