

# Understanding a business process

Take the following business process: a user has to check fuel prices using web resource 1 (external company website) weekly and update a file with the new values. Another user will then utilize web resource 2 (internal company website) to obtain information about distances traveled by vehicles in company service and correlates this information with the new costs of fuel. He then uses web resource 3 (external company website) to pay for the deliveries.

In this example, we could use three business process components:

- The first, a weekly one, would read data from resource 1 to check and update the fuel price file.
- The second would download information about distances from resource 2 and reference the values obtained by the previous sub-process to filter and further refine that data. Once done, it would save the data.
- The third component would read the information produced by process 2 use it to input data into resource 3.

This business process could, of course, be expressed as the sum of two business process components instead of three, for example by grouping sub-processes 1 and 2 together.

And, of course, it could be also broken up, for example sub-process 2 might be broken up into two further pieces, one that downloads information from resource 2 and another that reads both resource 1 and 2 information and processes it.

This technique of splitting a problem into easily definable, simple components is a great tool in solving any business process, no matter how complex.

It also helps deal with a fundamental aspect of reality, time.

To see exactly how, let's make a change to the business process described above: after the file containing fuel prices was updated a user would need to confirm the validity of the values by opening the file and signing it.

This means that before we utilize web resource 2 to obtain information we first need to wait for the fuel cost file to be signed by a supervisor.

You can easily see how, If we implemented this using 3 components, the change would be small, as we would need to only add one check in subprocess number two: is the file signed. If it is, proceed. If it is not, exit and try again later. This also means that the robot is free to process other tasks.