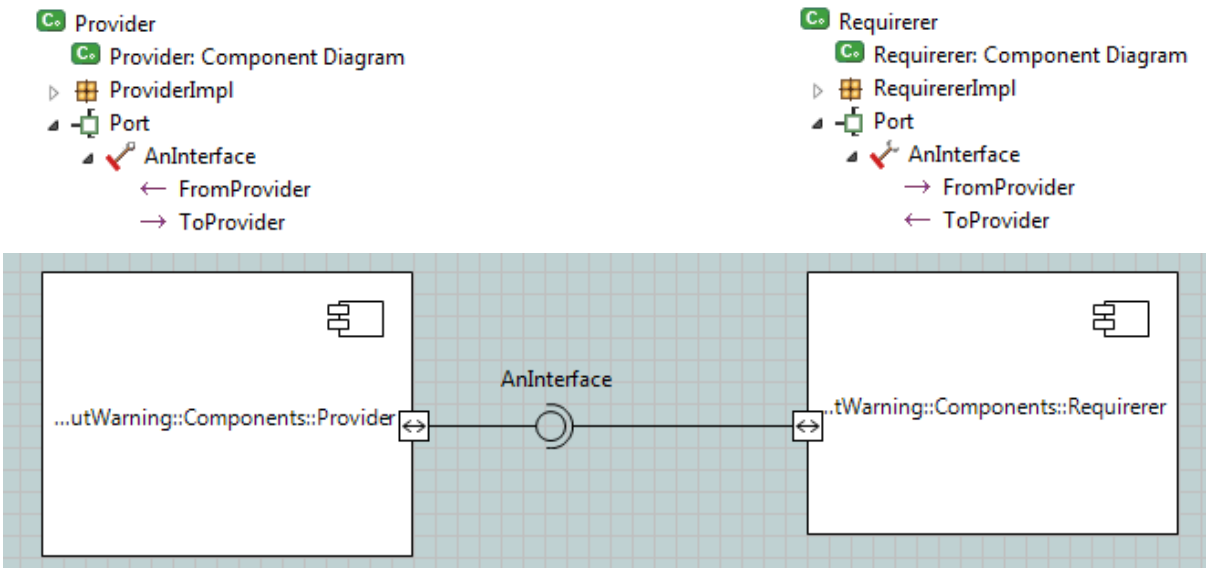
The tool allows to send (and to receive) a signal in wrong direction without warning

Summary

When signal is send through an interface in the wrong direction, instead of warning, behaviour is erroneous, that is a signal can be sent and received at the opposite side.

For example, in the configuration as depcited below, two components are used. A *Provider* and a *Requirer*. Both of the are connected through the interface *AnInterface*, which comprises of two signals, *FromProvider* and *ToProvider*. Clearly, all of the names used are self-illustrative.

If we try to send a signal from the Provider side by use of FromProvider, which should not be possible (except for the case when you try to send the signal from within the component to the same component), instead of warning (or at least "*nothing happens*" behaviour) we get the signal at the other end, i.e. at the Requirer side in the ToProvider.



Here is the list of the testSequence function at the Provider side:

```
send Port::FromProvider();
send Port::ToProvider(); //for this we should have a warning at least
```

And here is the port OAL code at the require side, first for a FromProvider signal:

```
LOG::LogInfo(message:"In the requirerer a FromProvider signal. This is OK");
```

And then for the ToProvider signal:

```
LOG::LogInfo(message:"In the requirerer a ToProvider signal? This should not
happen!");
```

The same is done in the opposite direction, and when we start the verifier against described application, by using test functions on both sides, we get following output on the console display:

```
User invoked function: testSequence
LogInfo:  In the requirerer a FromProvider signal. This is OK
LogInfo:  In the requirerer a ToProvider signal? This should not happen!
User invoked function: testSequence
LogInfo:  In the provider a From provider? This should not happen!
LogInfo:  In the provider ToProvider signal. This is OK.
```

This means that signal travels in wrong direction and that is not acceptable behaviour.

Attached to the SR is the BP project that can be used to reconstruct the erroneous behaviour.

Description of the project

In order to demonstrate erroneous behaviour a BP project named "SignalInWrongDirectionWithoutWaring" is created. This project, exported as an archive is attached to the SR.

The project comprises two components, defined in the Component package. They are named Provider and Requierer. Provider provides AnInterface, while Requierer requires the same interface. Interface is defined within the Interfaces package. It consists of two signals FromProvider, with the direction from provider, and ToProvider with the direction set towards provider.

In order to test desired functionality, in all ports (both components) an OAL code has been placed:

Provider::FromProvider:

```
LOG::LogInfo(message:"In the provider a From provider? This should not happen!");
```

Provider::ToProvider:

```
LOG::LogInfo(message:"In the provider ToProvider signal. This is OK.");
```

Requirerer::FromProvider

```
LOG::LogInfo(message:"In the requirerer a FromProvider signal. This is OK");
```

Requirerer::ToProvider

```
LOG::LogInfo(message:"In the requirerer a ToProvider signal? This should not
happen!");
```

Finally, in the component implementation packages, for both components testSequence functions are defined:

Provider testSequence:

```
send Port::FromProvider();
send Port::ToProvider(); //for this we should have a warning at least
```

Requirerer testSequence:

```
send Port::FromProvider();//for this, I should have a warning at least
send Port::ToProvider();
```

In the run configurations a single configuration is set in which two component references to the mentioned components are connected directly.

The debug configuration is as the figure below depicts: