

Erste Schritte mit **expecco**

Inhaltsverzeichnis

Vorwort.....	2
Erzeugen einer Testsuite.....	2
Bibliotheken und Bausteine.....	3
Erstellen eines zusammengesetzten Bausteins	5
Erstellen eines komplexeren Bausteins.....	10
Hinzufügen von Ausgangspins zu einem Baustein.....	13
Hinzufügen von Eingangspins zu einem Baustein.....	16
Wiederverwendung von Bausteinen.....	20
Sequentielle Ausführung von Schritten	22
Erstellen eines Testplans.....	24
Erstellen eines Berichts (Report).....	27
Speichern der Testsuite	28
Schlusswort	28

Vorwort

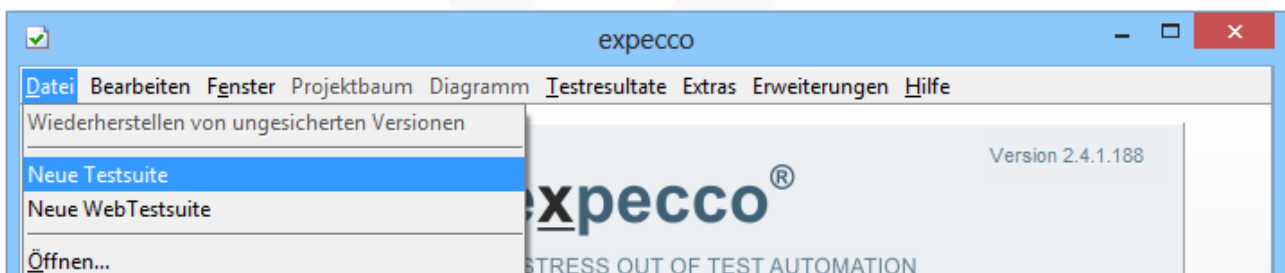
Die *Ersten Schritte* richten sich an alle, die **expecco** zum ersten Mal benutzen und soll als erste Starthilfe dienen.

Anhand eines einfachen Beispiels wollen wir Ihnen einen Überblick über **expecco** verschaffen.

Es empfiehlt sich **expecco** parallel zum Tutorial zu starten um alle Schritte innerhalb von **expecco** nachvollziehen zu können.

Erzeugen einer Testsuite

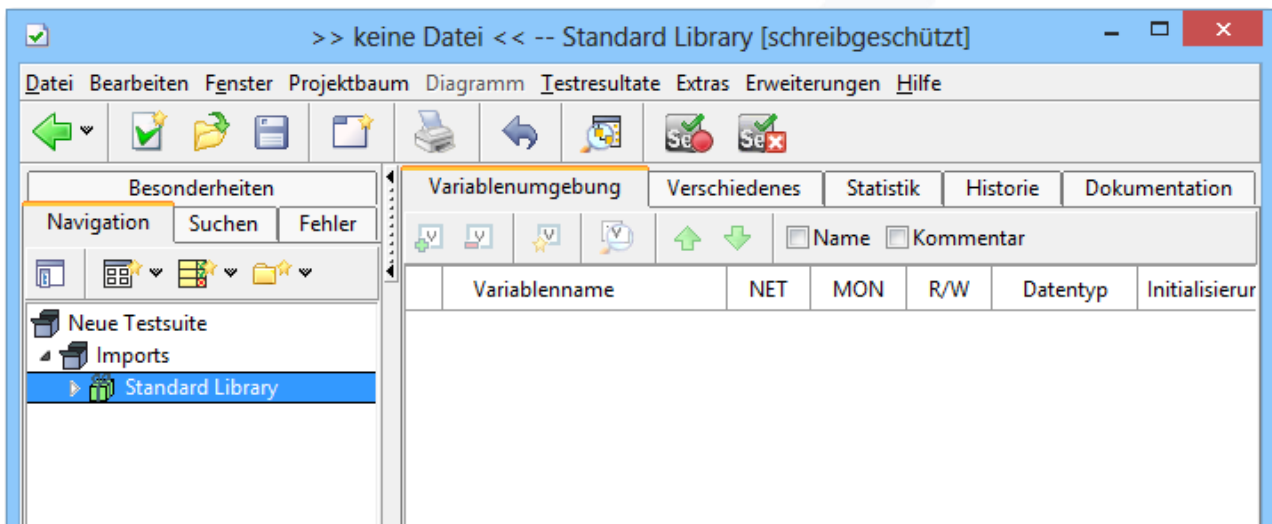
Eine Testsuite bezeichnet in expecco die Gesamtheit eines Testprojektes. Um eine neue Testsuite zu erzeugen, klicken Sie auf das Menü *Datei* → *Neue Testsuite*.



Bibliotheken und Bausteine

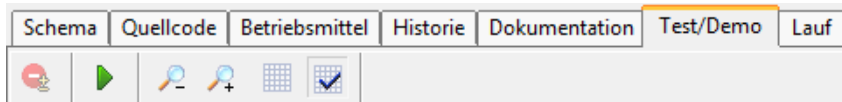
In den **Bibliotheken** sind die einzelnen Funktionsbausteine und auch die Testpläne organisiert. Bibliotheken finden Sie im Projektbaum (*Navigator*) unter der Kategorie „*Imports*“. Alle Bausteine innerhalb einer importierten Bibliothek sind schreibgeschützt. Zu jedem Baustein existiert eine Beschreibung und gegebenenfalls ein Beispiel, das Sie sofort ausführen können.

Selektieren Sie im Projektbaum die Standard-Bibliothek. Diese Bibliothek stellt grundlegende Funktionsbausteine für unterschiedliche Bereiche zur Verfügung.

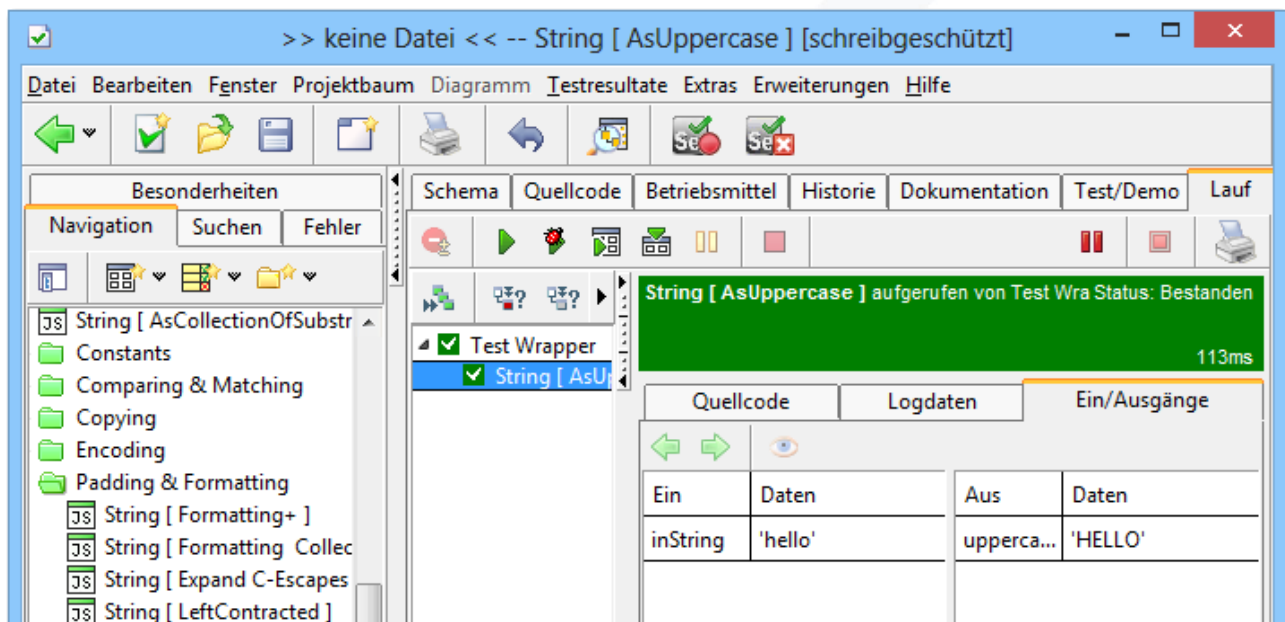


Innerhalb einer Bibliothek sind die Bausteine nach Kategorien in Ordnern organisiert. Zum Beispiel finden Sie diverse Bausteine zur Stringverarbeitung in einem gemeinsamen Ordner. Zur Veranschaulichung navigieren Sie bitte zu „*String Processing*“, dann zum Unterordner „*Padding & Formatting*“ und selektieren dort den Baustein „*String [AsUppercase]*“. Im rechten Fenster werden Editoren für das selektierte Element in mehreren Reitern dargestellt. In dem Reiter „*Dokumentation*“ finden sie zu jedem Baustein eine kurze Beschreibung der Funktionalität (diese wird auch links im Auswahlbaum als „*FlyBy-Help*“ angezeigt).

Zu den meisten Bausteinen gibt es auch einen kleinen Testrahmen, in dem die Funktionalität demonstriert bzw. validiert wird. Wählen Sie den Reiter „*Test/Demo*“ aus.



In diesem Fenster kann der Baustein durch Klicken der „Start-Testlauf“ Schaltfläche  direkt ausgeführt werden.



Das Resultat der Ausführung wird in einem separaten Fenster unter dem Reiter „*Lauf*“ angezeigt. Links daneben sehen Sie die hierarchische Aufrufkette, rechts die zu der linken Auswahl zugehörigen Detailinformationen. So sehen Sie im Reiter „*Ein/Ausgänge*“ die Werte der Ein- und Ausgänge.

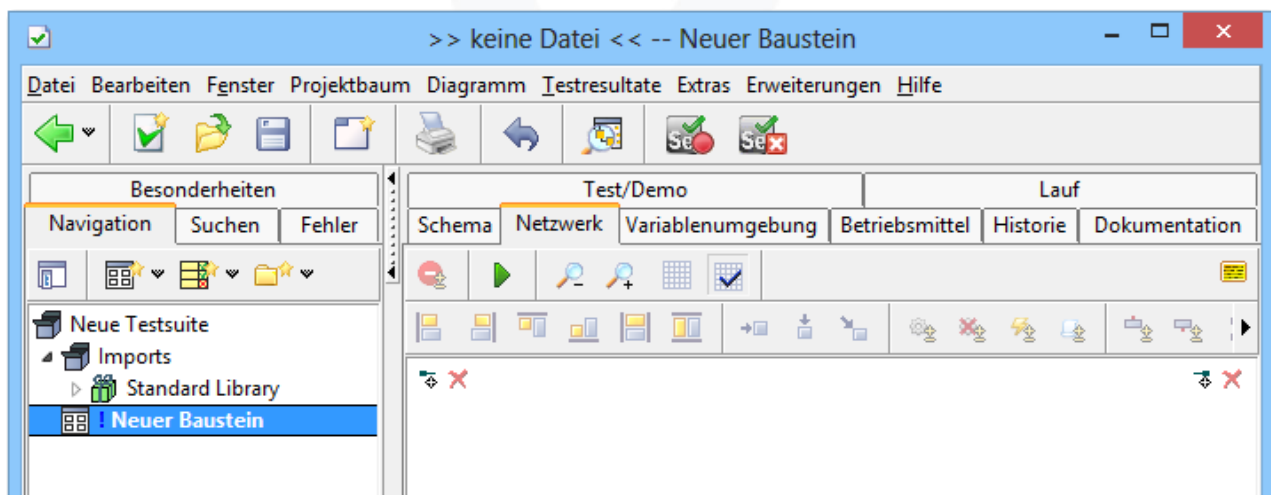
Erstellen eines zusammengesetzten Bausteins

Ein Baustein stellt die Definition eines Vorgangs dar. Man unterscheidet zwischen Elementarbausteinen und zusammengesetzten Bausteinen.

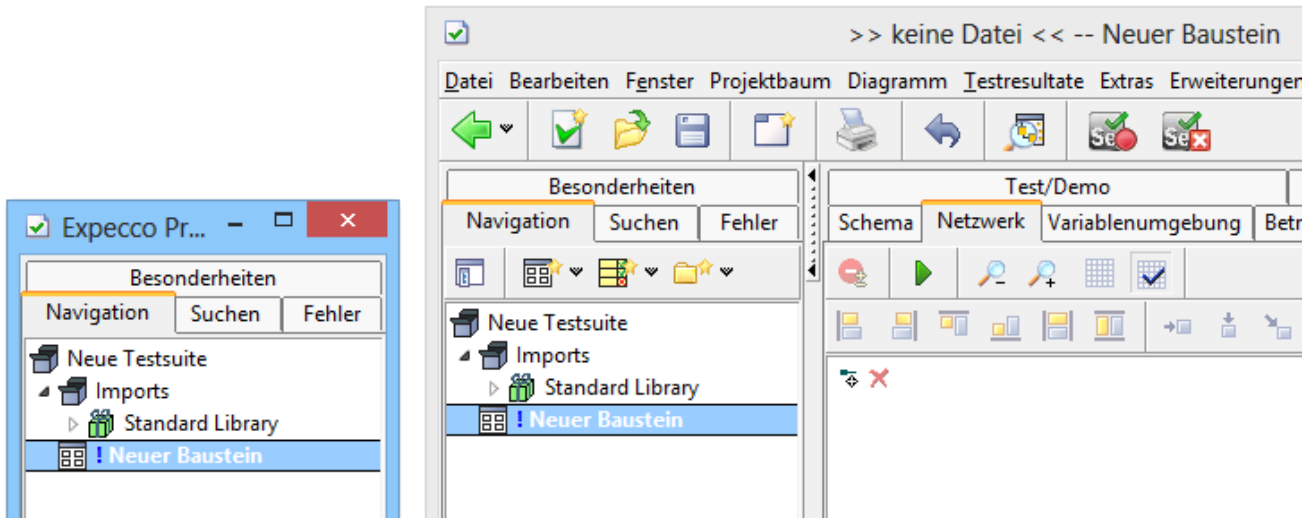
Ein Elementarbaustein definiert einen grundlegenden Vorgang, wie beispielsweise einen Protokollaufruf und wird durch Programmcode beschrieben.

Das Verhalten eines zusammengesetzten Bausteins dagegen wird durch ein Aktivitätsdiagramm beschrieben. Die Elemente eines Aktivitätsdiagramms sind wiederum Bausteine, welche über Daten- oder Kontrollflüsse verbunden sind. In einem Aktivitätsdiagramm platzierte Bausteine werden als Schritte bezeichnet.

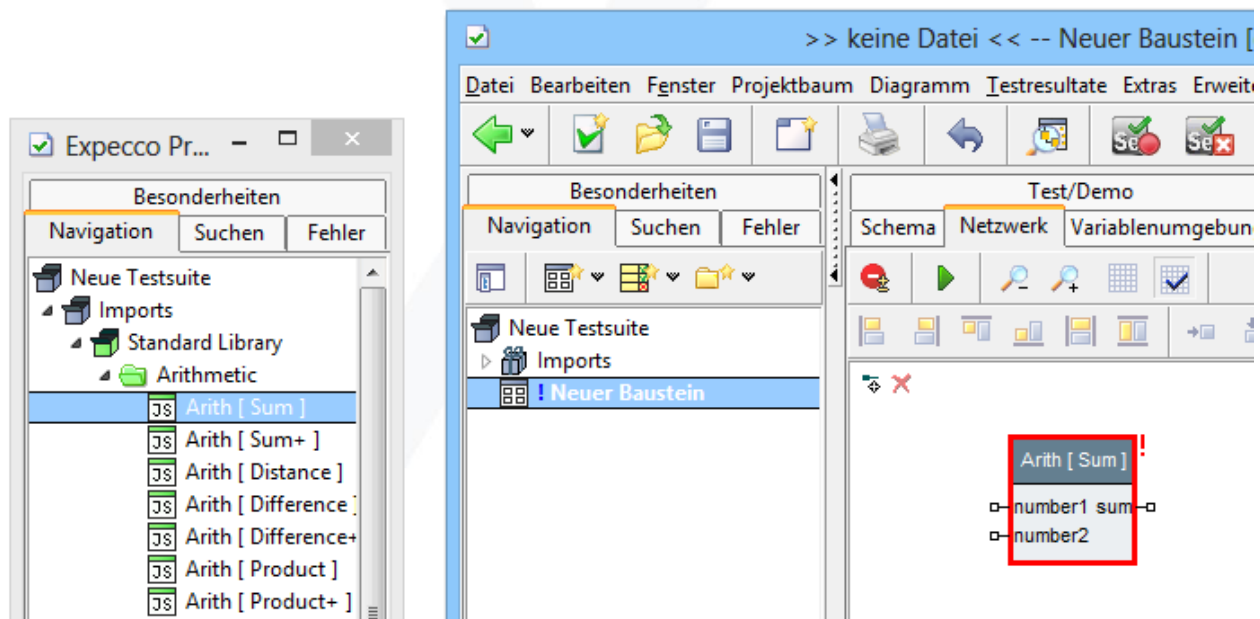
Klicken Sie in der Toolbar  die Schaltfläche  (Neue zusammengesetzte Aktion), um einen neuen zusammengesetzten Baustein zu erzeugen.



Klicken Sie die Schaltfläche . Sie erhalten ein zweites Fenster aus dem Bausteine per Drag & Drop in das „Netzwerk“-Fenster des neu erstellten Bausteins eingefügt werden können.



Selektieren Sie in der Kategorie „Arithmetic“ den Baustein „Arith [Sum]“ und ziehen Sie diesen Baustein per Drag & Drop in das Netzwerk des neu erstellten Bausteins.

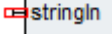


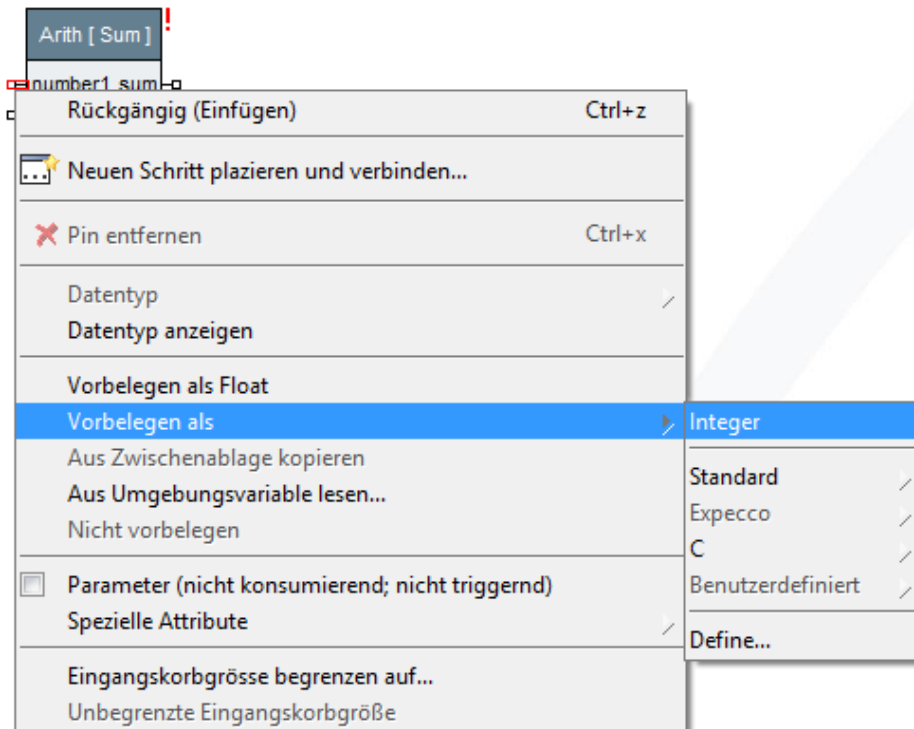
Durch Platzieren des Bausteins im Netzwerk wird ein neuer Schritt angelegt.

Da dieser Schritt Eingabewerte erfordert, kann er nicht ausgeführt werden (erkennbar am roten

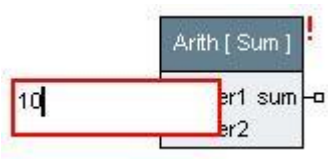


Ausrufezeichen ). Eingänge und Ausgänge werden als Pins (Anschlüsse) bezeichnet. Im ersten Schritt belegen wir die Eingänge mit festen Werten vor.

Dies erreichen Sie über einen Doppelklick auf den Eingangspin  oder über den entsprechenden Eintrag des Kontextmenüs:



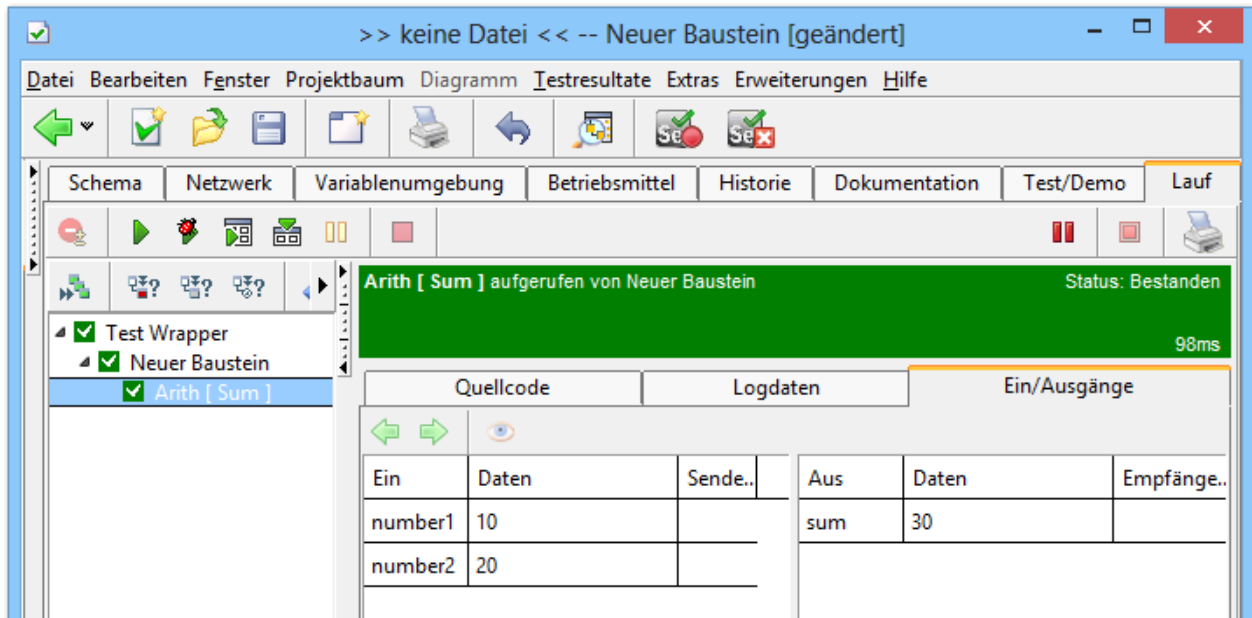
Geben Sie nun eine beliebige Zahl im Eingabefeld ein.



Belegen Sie nun auch den zweiten Eingangspin vor. Der Schritt besitzt nun alle zur Ausführung benötigten Eingangswerte.

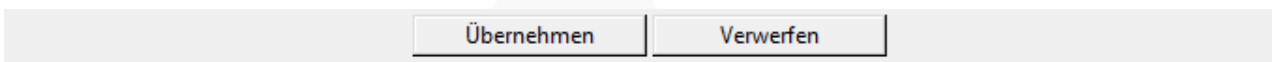


Führen Sie Ihren neu erstellten Baustein unter „Test/Demo“ aus.

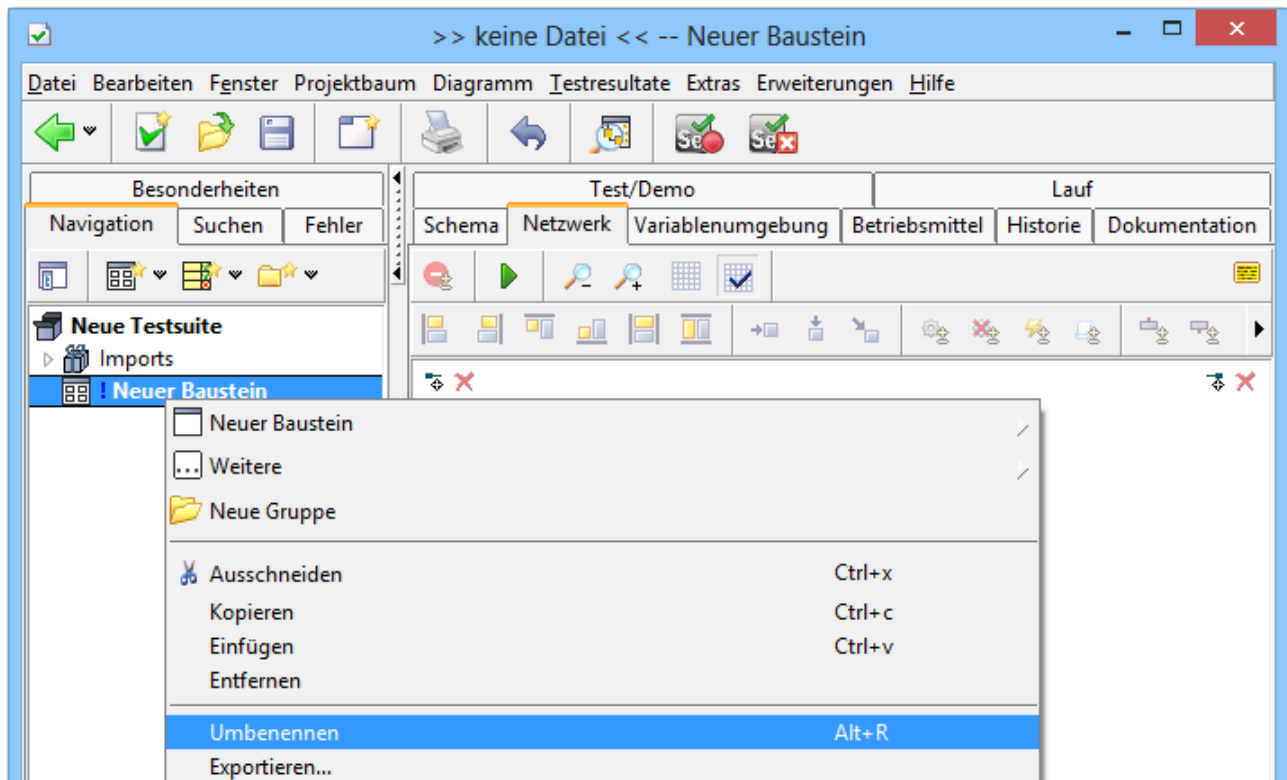


Unter „Ein/Ausgänge“ sehen Sie die Eingangswerte und den erwarteten Ausgangswert.

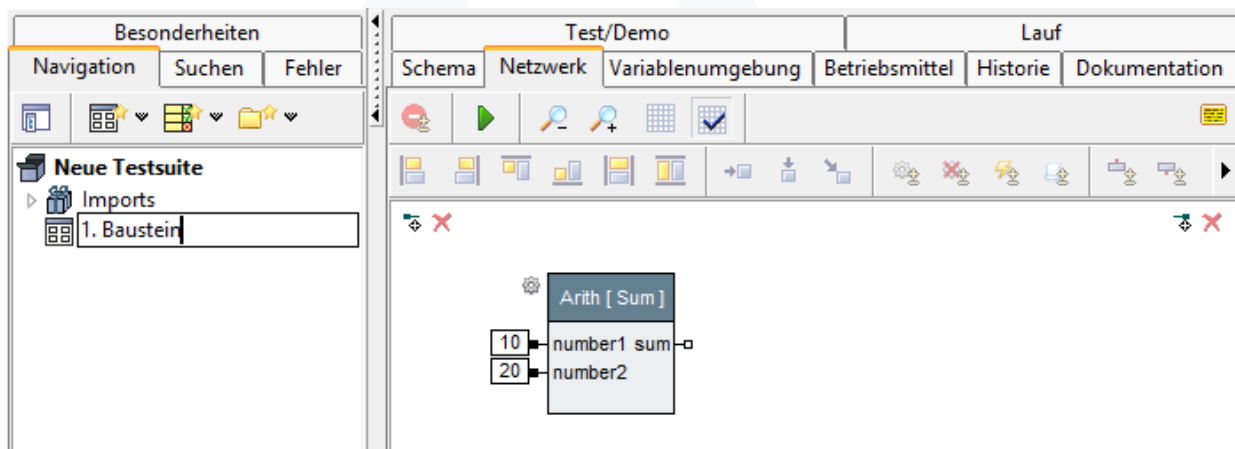
Um die Änderungen des Bausteins im System zu übernehmen, klicken Sie auf die Schaltfläche „Übernehmen“.



Geben Sie dem Baustein einen Namen; dies erreichen Sie über das Kontextmenü des neu angelegten Bausteins innerhalb des Projektbaums.

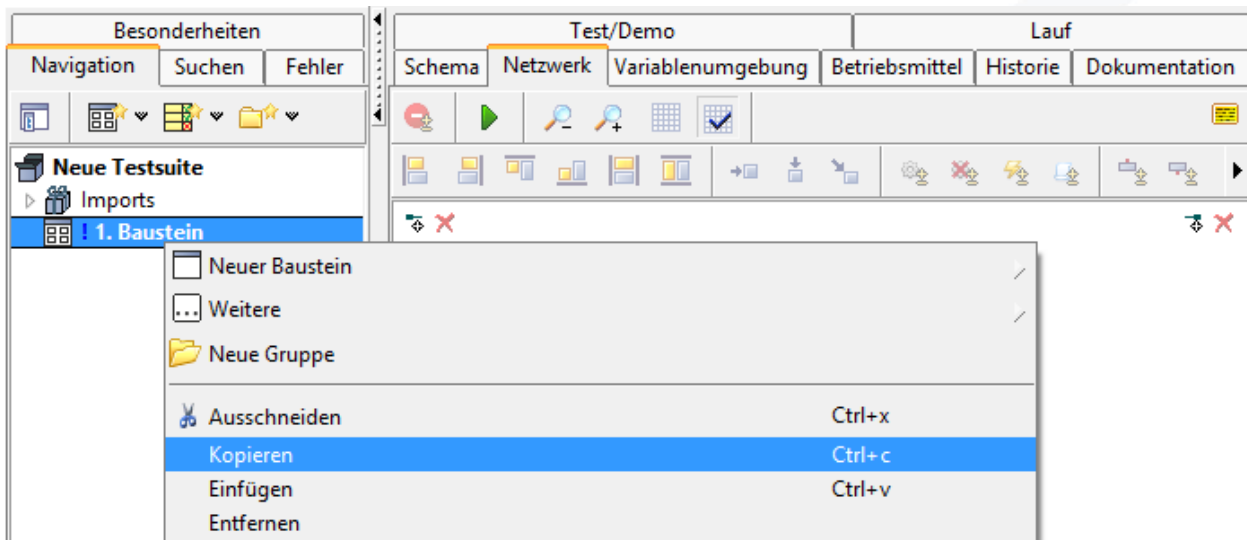


Geben Sie als Namen „1. Baustein“ in das Eingabefeld ein.

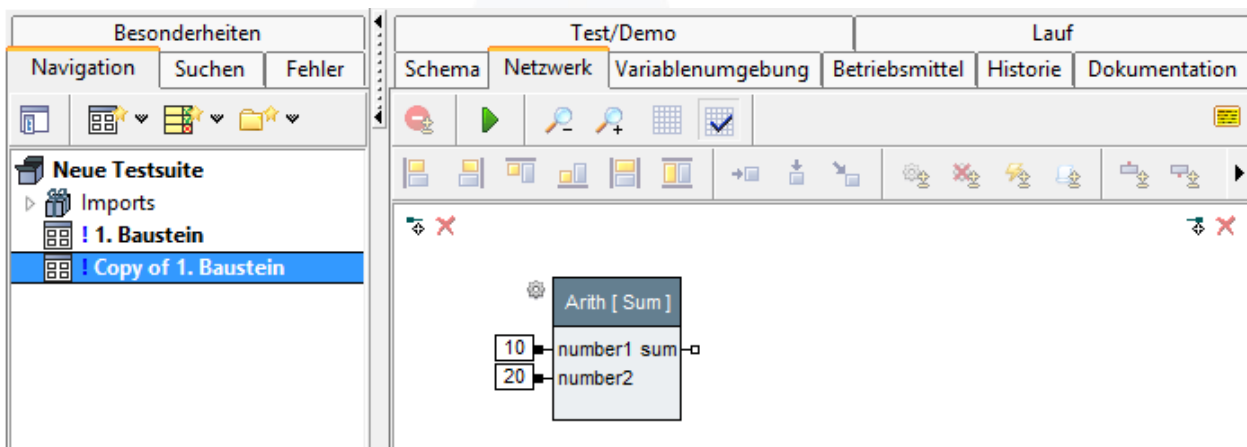


Erstellen eines komplexeren Bausteins

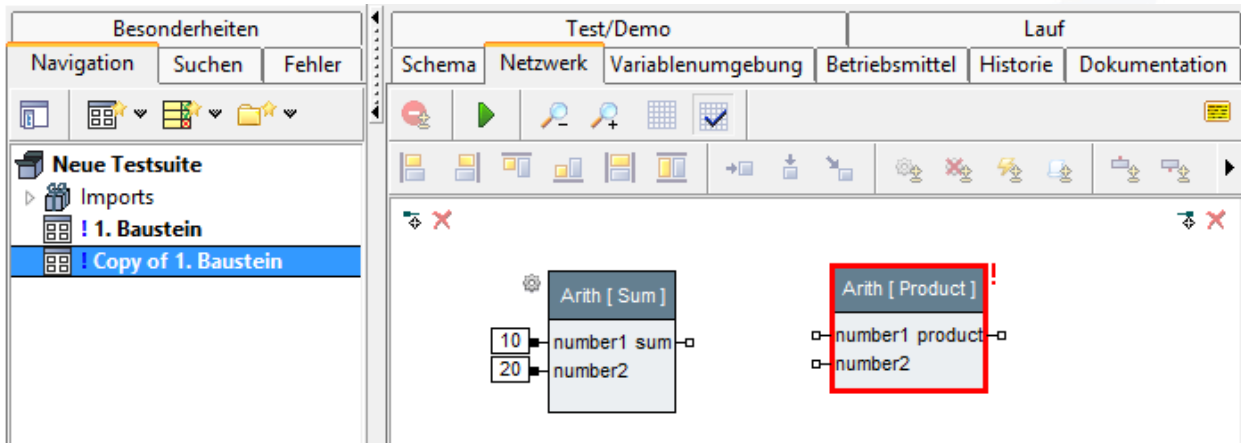
Als Basis hierzu verwenden wir eine Kopie des vorhergehenden Bausteins, welche Sie über das Kontextmenü des selektierten Elements anlegen können (Kopieren dann Einfügen).



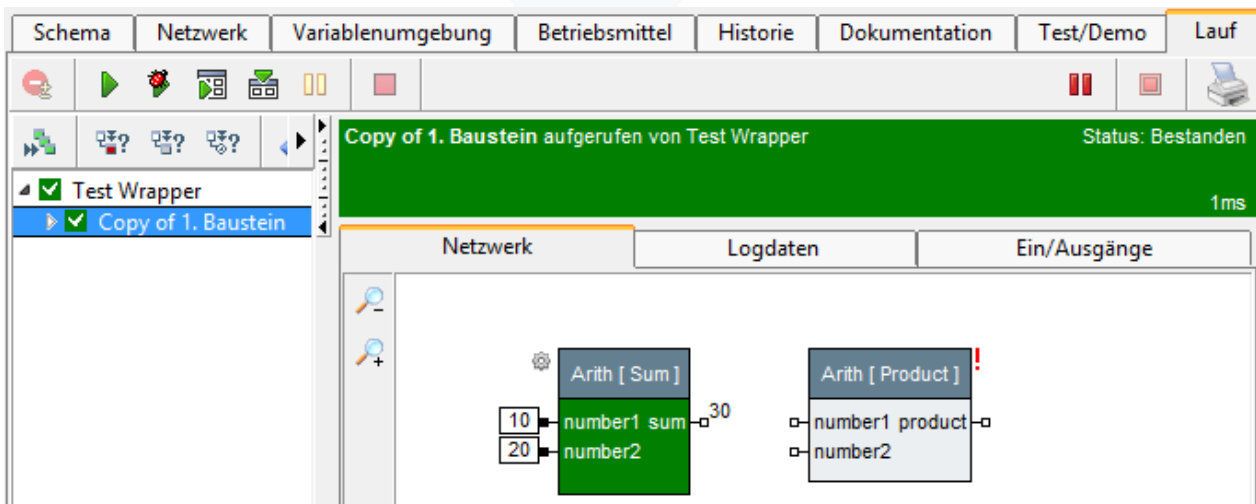
Nach dem Einfügen erhalten Sie eine Kopie Ihres vorhergehenden Bausteins.



Falls Sie sich noch unter „Lauf“ befinden, wechseln Sie zum Reiter „Netzwerk“. Platzieren Sie nun den Baustein „Arith [Product]“ über Drag & Drop in das Netzwerk. Danach sollte Ihr Netzwerk folgendermaßen aussehen:



Nach der Ausführung des Bausteins erhalten Sie folgendes Bild:



Die ausgeführten Aktivitäten werden abhängig von ihrem Ausführungszustand farblich dargestellt; hier „grün“ für erfolgreich ausgeführt. Für den neu platzierten Schritt wurde keine Aktivität gestartet, da diesem Schritt die benötigten Eingabewerte fehlen (erkennbar am roten Ausrufezeichen).

Dazu verbinden wir die Eingänge und Ausgänge der Schritte untereinander.

Die meisten Verbindungen sind Datentyp-orientiert, d.h. sie müssen den gleichen Datentyp am Anfang und am Ende der Verbindung haben. So kann beispielsweise kein Text in einen Additionsbaustein eingeführt werden.

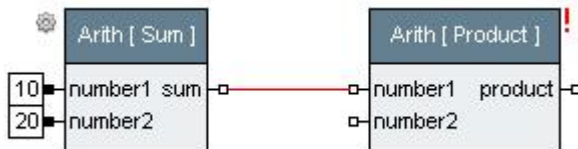
Falls Ein- und Ausgangsdattentyp nicht übereinstimmen, stehen in der Standard-Bibliothek im Ordner „*Type Conversion*“ verschiedene Konvertierungsbausteine zur Verfügung.

Um eine Verbindung einzurichten, gehen Sie folgendermaßen vor:

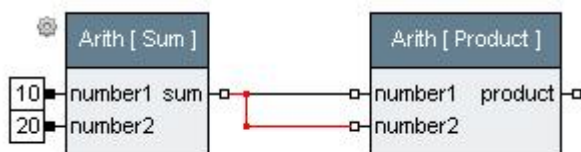
1. Selektieren Sie mit der linken Maustaste den Ausgangspin „*sum*“ am Schritt „*Arith [Sum]*“ und halten Sie diese gedrückt.



2. Ziehen Sie nun mit gedrückter Maustaste ein Verbindung zum Eingangspin „*number1*“ am Schritt „*Arith [Product]*“ und lösen Sie die Maustaste über dem Eingangspin □.



Wiederholen Sie diesen Schritt und verbinden Sie zusätzlich den Ausgangspin „*sum*“ mit dem Eingangspin „*number2*“.



Führen Sie den Test erneut aus; für alle Schritte wurden nun Aktivitäten ausgeführt.

The screenshot shows the 'Test/Demo' tab in the eXept interface. The main area displays 'Arith [Product]' with a status of 'Bestanden' (Passed) and a duration of '76ms'. Below this, there are three sections: 'Quellcode', 'Logdaten', and 'Ein/Ausgänge'.

Quellcode			Logdaten		Ein/Ausgänge	
Ein	Daten	Sender	Aus	Daten	Empfänge..	
number1	30	Arith [Sum]	product	900		
number2	30	Arith [Sum]				

Übernehmen Sie nun den Baustein und benennen Sie ihn in „2. Baustein“ um.

Hinzufügen von Ausgangspins zu einem Baustein

Als Basis hierzu verwenden wir eine Kopie des vorhergehenden Bausteins, welche Sie über das Kontextmenü anlegen können (Kopieren dann Einfügen).

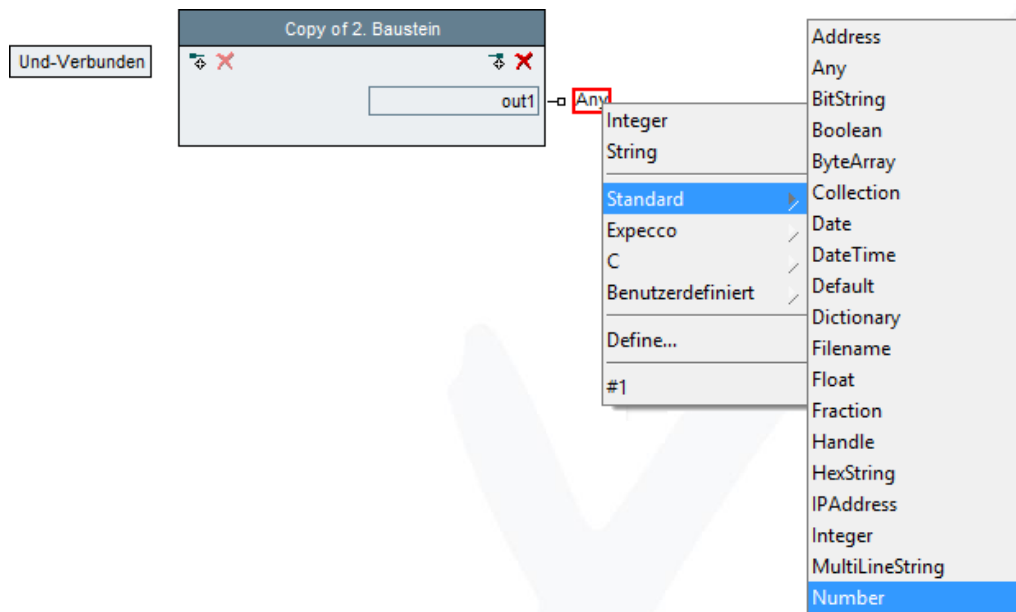
Sie erhalten folgendes Bild:

The screenshot shows the 'Schema' tab in the eXept interface. The left sidebar shows a tree view with 'Neue Testsuite' and 'Imports'. The main area shows a diagram with a block labeled 'Copy of 2. Baustein'. The block has two input pins on the left and two output pins on the right, each with a red 'X' icon. A label 'Und-Verbunden' is visible next to the block.

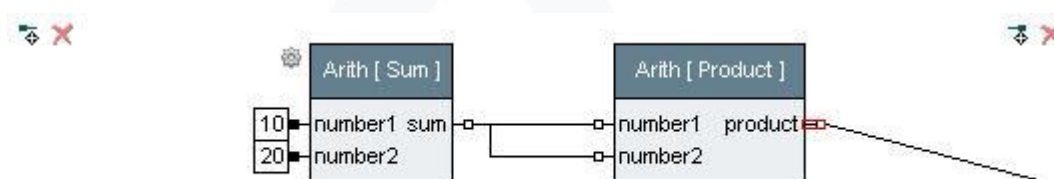
Die Schnittstelle eines Bausteins wird über seine Eingänge und Ausgänge beschrieben. Im Schema-Editor werden die Eingangspins auf der linken Seite, die Ausgangspins auf der rechten Seite dargestellt. Jedem Pin wird ein Datentyp zugewiesen.

Es gibt verschiedene Möglichkeiten um neue Pins zu erzeugen:

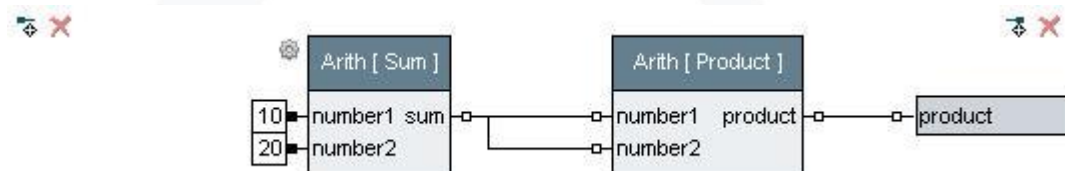
1. Klicken Sie auf die Schaltfläche  innerhalb des Schema-Editors um einen neuen Ein- bzw. Ausgangspin zu erzeugen. Um den voreingestellten Datentyp zu ändern, klicken Sie auf diesen und wählen aus dem erscheinenden Menü den gewünschten Datentyp aus.



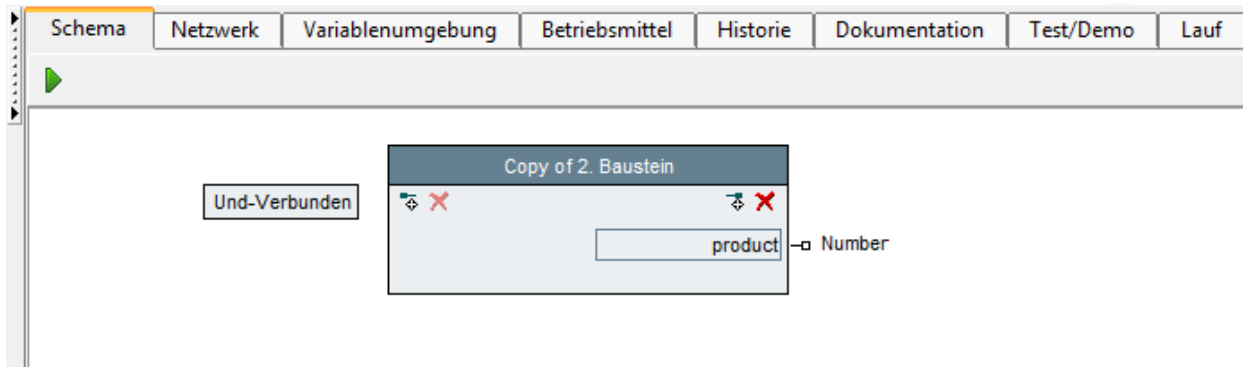
2. Ziehen Sie den entsprechenden Ein- bzw. Ausgangspin des Schrittes innerhalb des Netzwerks an die linke bzw. rechte Seite des Editors.



Nach dem Lösen der Maustaste am linken Rand sehen Sie folgendes Bild:



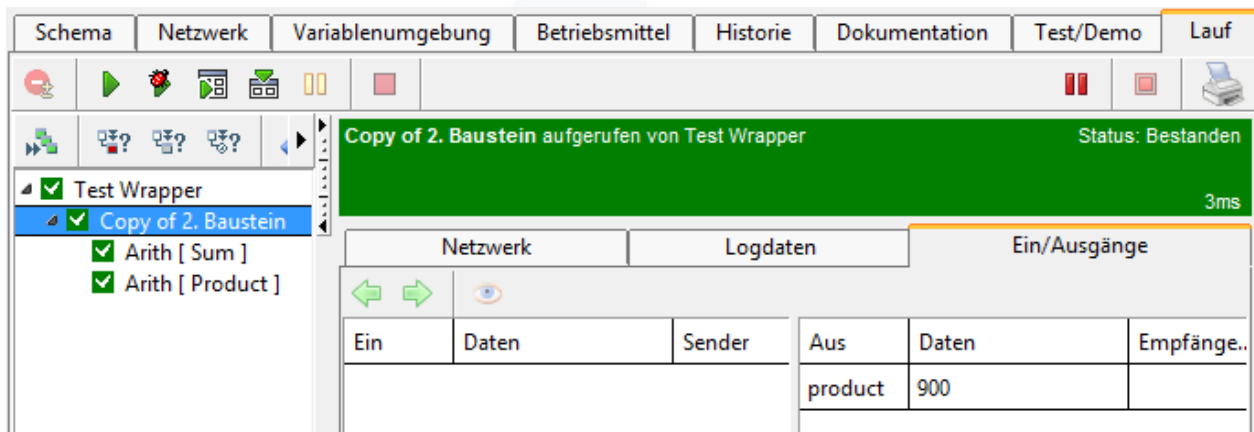
Erzeugen Sie den Ausgangspin wie in Beispiel 2 beschrieben.
Nach dem Selektieren des Reiters „Schema“, erhalten Sie folgendes Bild:



Der Baustein hat einen neuen Ausgangspin erhalten. Der Datentyp sowie der Name des Ausgangspins wurden vom intern verbundenen Ausgangspin im Netzwerk übernommen. Durch Klicken auf den Namen des Pins kann dieser umbenannt werden.

Um einen bestehenden Ein- bzw. Ausgangspin zu löschen, selektieren Sie den entsprechenden Pin und klicken auf die Schaltfläche .

Führen Sie den Baustein erneut aus und Sie erhalten folgendes Bild:

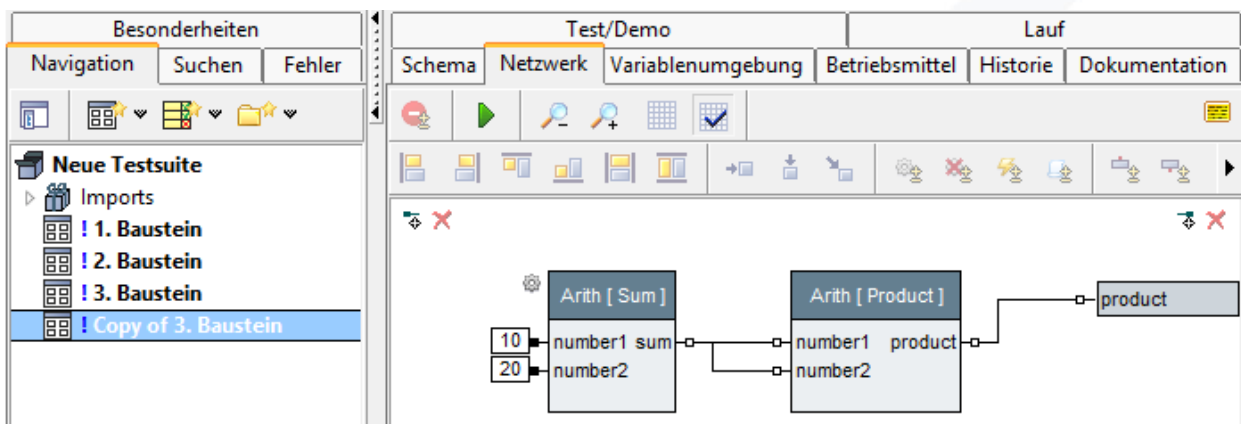


Der Ausgangswert steht nun zur weiteren Verwendung am Ausgangspin zur Verfügung. Übernehmen Sie nun den Baustein und benennen Sie ihn in „3. Baustein“ um.

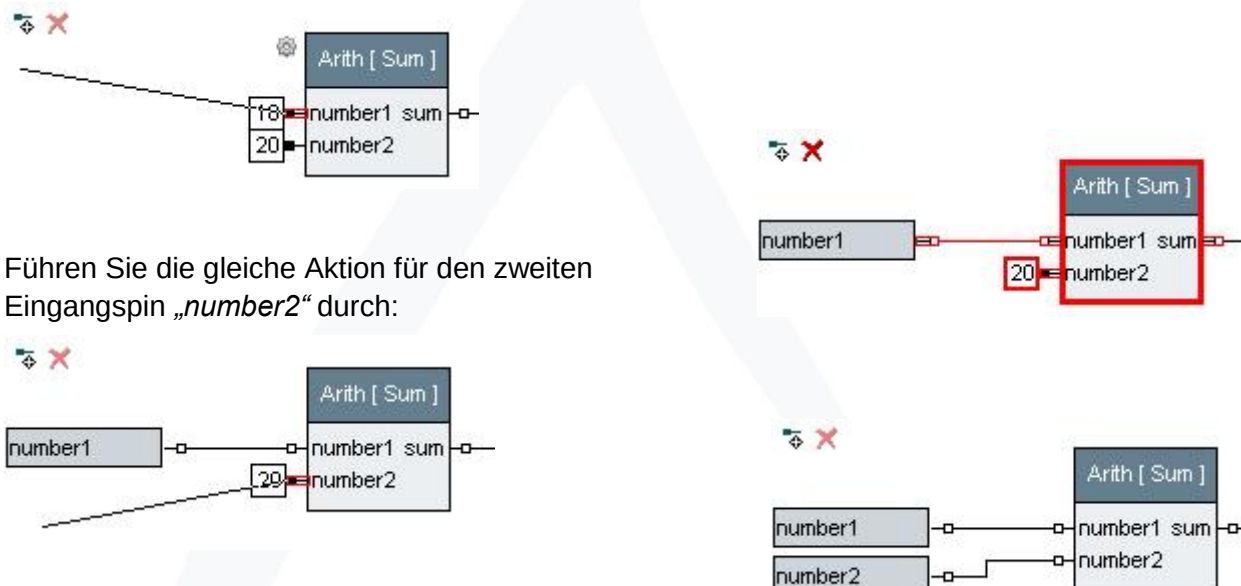
Hinzufügen von Eingangspins zu einem Baustein

Als Basis hierzu verwenden wir eine Kopie des vorhergehenden Bausteins, welche Sie über das Kontextmenü anlegen können (Kopieren dann Einfügen).

Sie erhalten folgendes Bild:

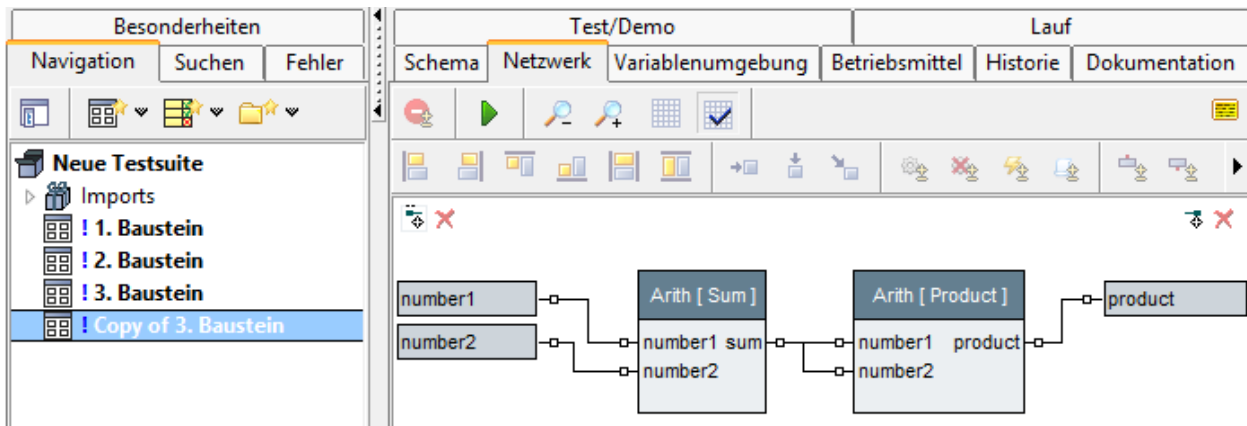


Um den Schritt „Arith [Sum]“ von außen zu versorgen, erzeugen Sie zwei Eingangspins (gleiche Vorgehensweise wie bei den Ausgangspins):

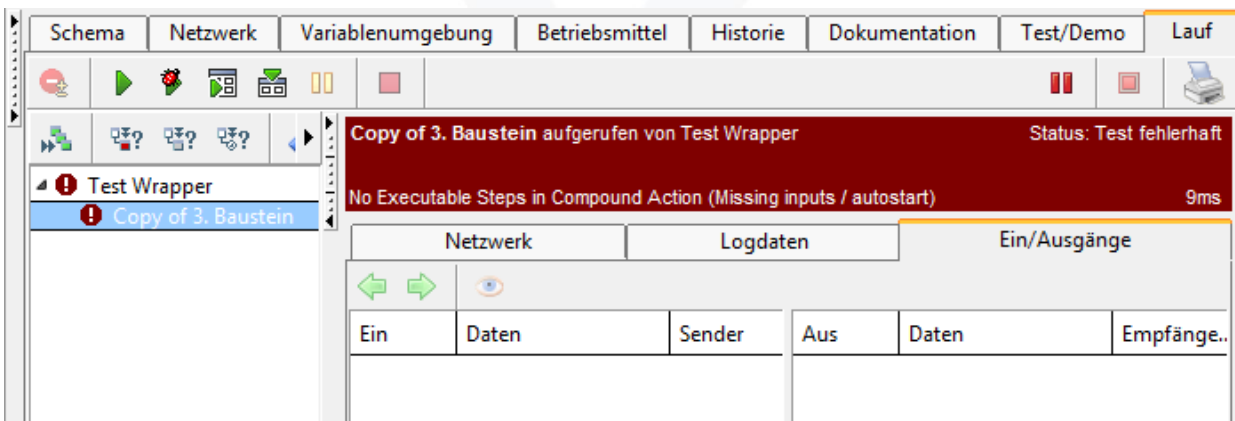


Führen Sie die gleiche Aktion für den zweiten Eingangspin „number2“ durch:

Sie erhalten folgendes Bild:

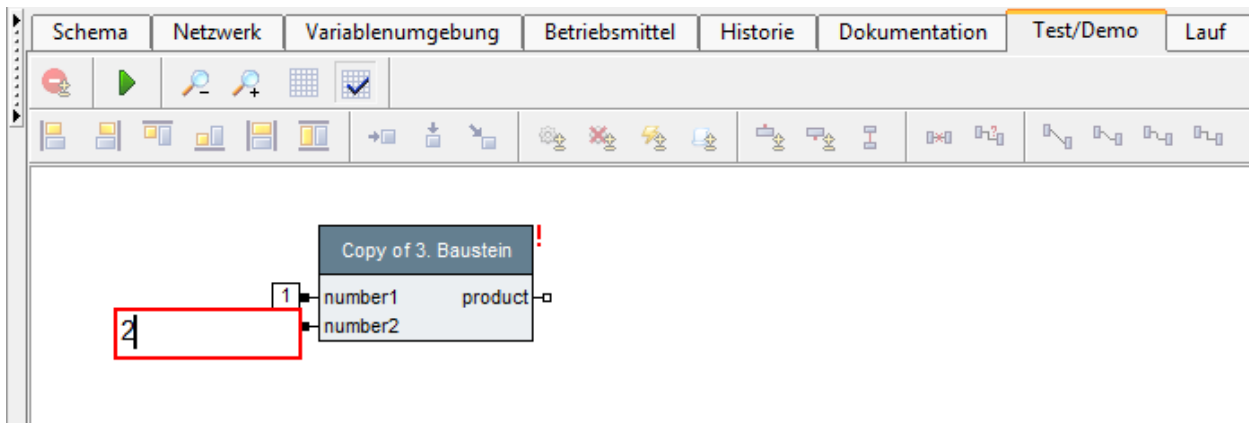


Führen Sie den Baustein erneut aus, erhalten Sie folgendes Bild:



Der Baustein konnte nicht ausgeführt werden, da die erforderlichen Eingabewerte an den Eingangspins unseres Testbausteins fehlen.

Belegen Sie nun die Eingänge mit festen Eingangswerten vor:



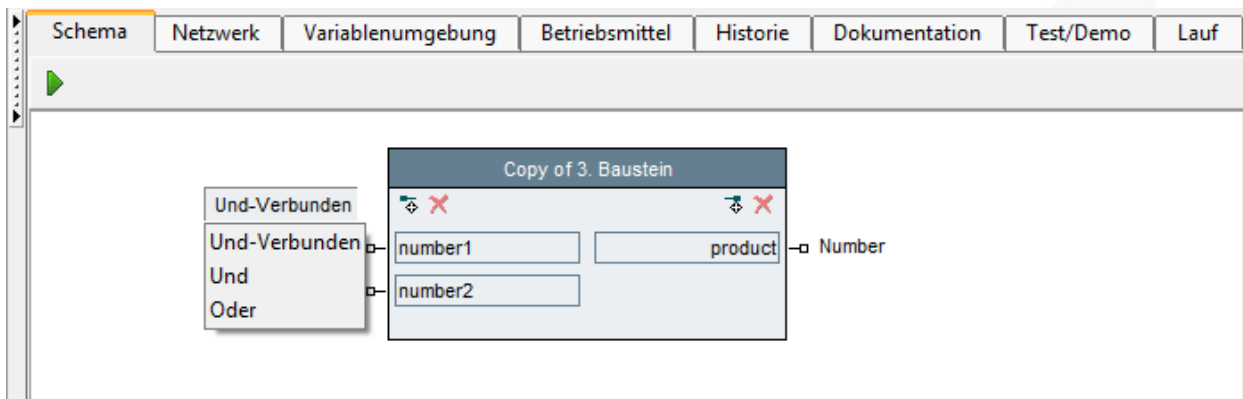
Führen Sie den Baustein erneut aus, erhalten Sie folgendes Bild:

The screenshot shows the 'Test/Demo' tab in the expecco software. The component 'Copy of 3. Baustein' is highlighted in green, indicating a successful execution. The 'Ein/Ausgänge' (Inputs/Outputs) table shows the input values and the resulting output.

Netzwerk			Logdaten		Ein/Ausgänge	
Ein	Daten	Sender	Aus	Daten	Empfänge.	
number1	1		product	9		
number2	2					

Schalten Sie nun auf „Schema“ um. Hier können Sie die Ausführungsbedingung für den Baustein festlegen. Klicken Sie hierfür auf das Feld „Und-Verbunden“.

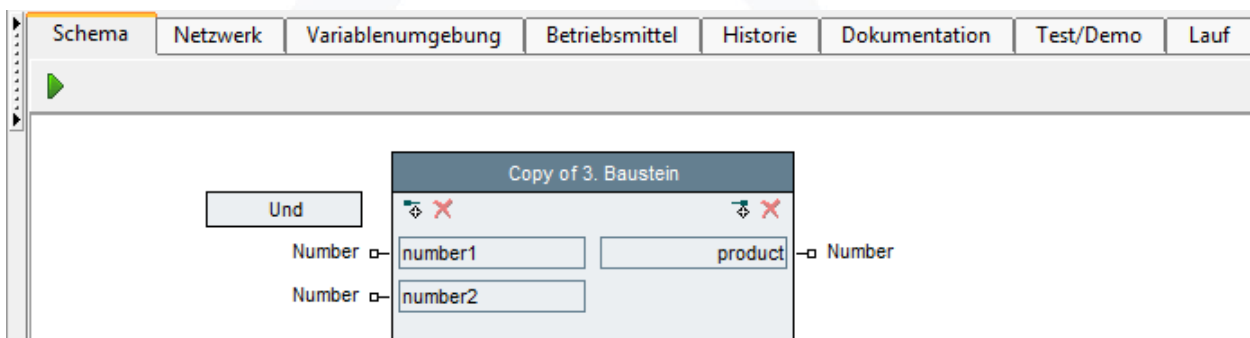
Ein Untermenü zur Konfiguration wird angezeigt:



Hier können Sie spezifizieren, welche Eingangspins Werte erhalten haben müssen, damit der Baustein ausgeführt wird. Hierzu werden 3 Möglichkeiten unterstützt:

1. *Und-Verbunden*
Alle verbundenen Eingangspins müssen Werte erhalten haben, offene Eingänge werden ignoriert.
2. *Und*
Alle Eingangspins müssen Werte erhalten haben.
3. *Oder*
Mindestens ein Eingangspin muss einen Wert erhalten haben.

In unserem Fall müssen alle Eingangspins einen Wert erhalten haben, also wählen Sie die „Und“ Bedingung aus.

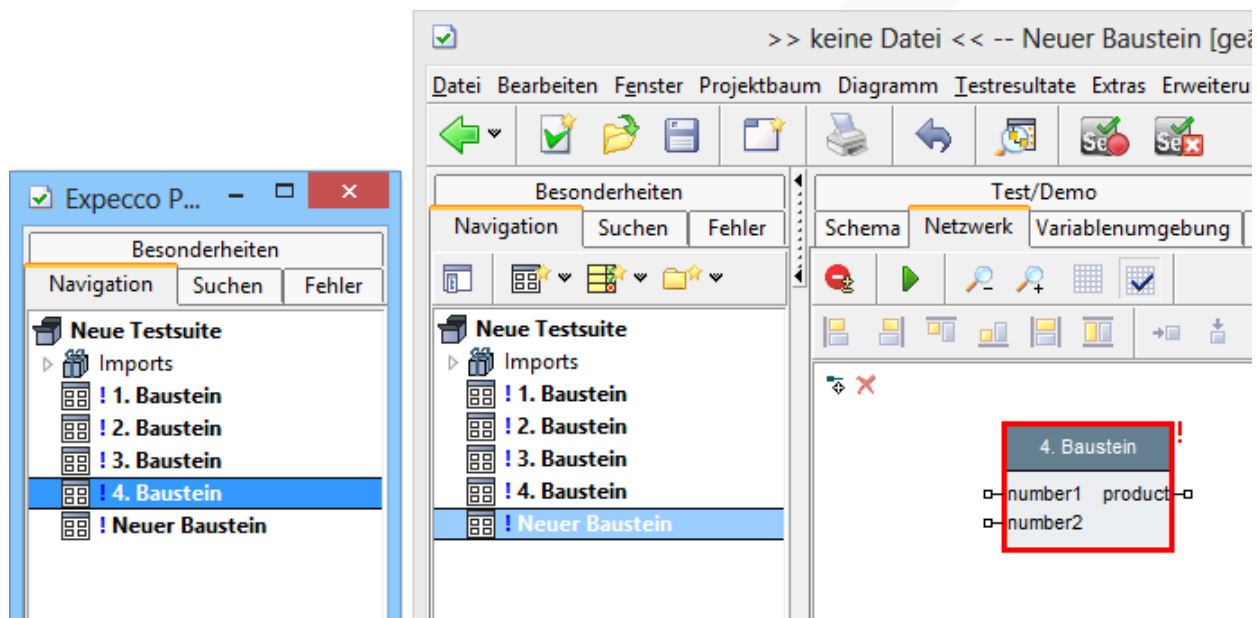


Übernehmen Sie nun den Baustein und benennen Sie ihn in „4. Baustein“ um.

Wiederverwendung von Bausteinen

Die Arbeitsweise mit Bausteinen ist der Schlüssel zu Wiederverwendbarkeit und Flexibilität. Einmal erstellte Bausteine können in beliebigen zusammengesetzten Bausteinen wiederverwendet werden.

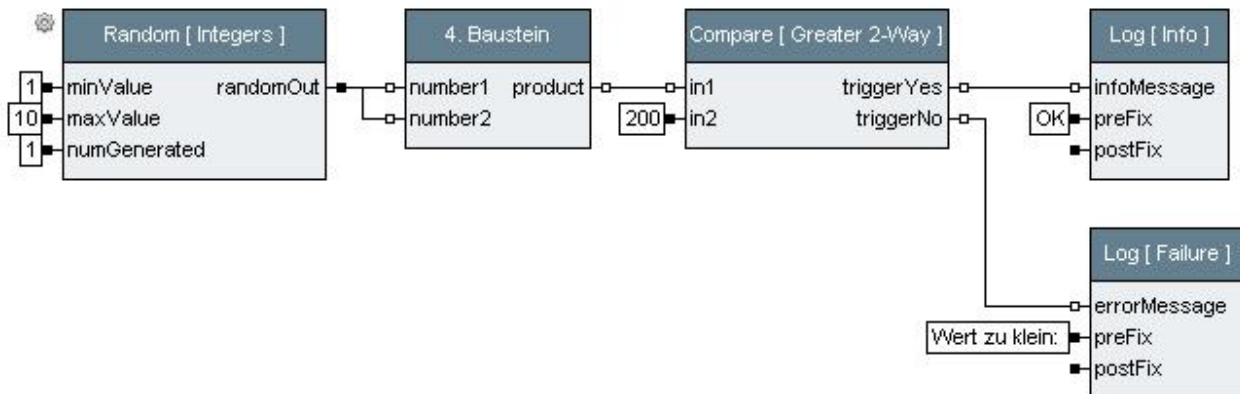
Legen Sie einen neuen zusammengesetzten Baustein durch Klicken der Schaltfläche  an. Nun ziehen Sie per Drag & Drop den vorher erstellten Baustein in das Netzwerk (Aktivitätsdiagramm) des neu angelegten Bausteins.



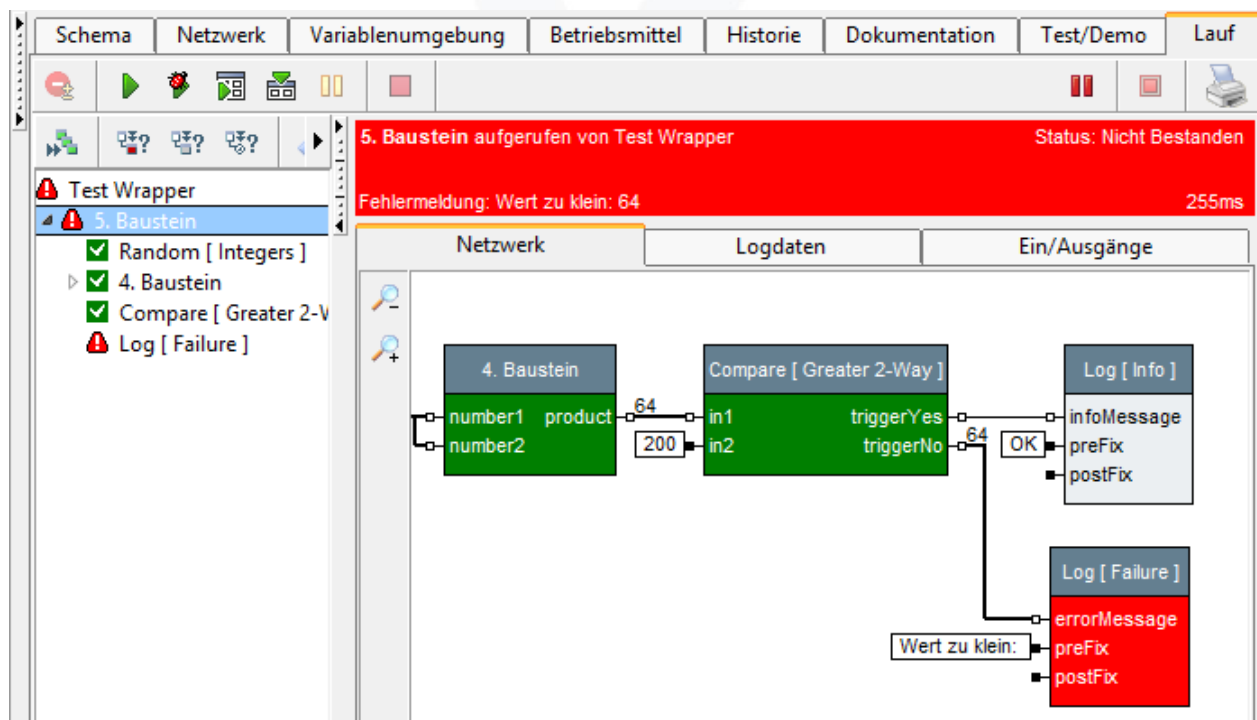
Fügen Sie zusätzlich noch Bausteine aus der Standard-Bibliothek hinzu und stellen Sie die Verbindungen wie im folgenden Netzwerk her.

Sie finden die Bausteine unter den Kategorien:

- *Data Generators*
Random [Integers]
- *Compare*
Compare [Greater 2-Way]
- *Assertions, Exceptions & Logging -> Logging & Messages*
Log [Info], Log [Failure]



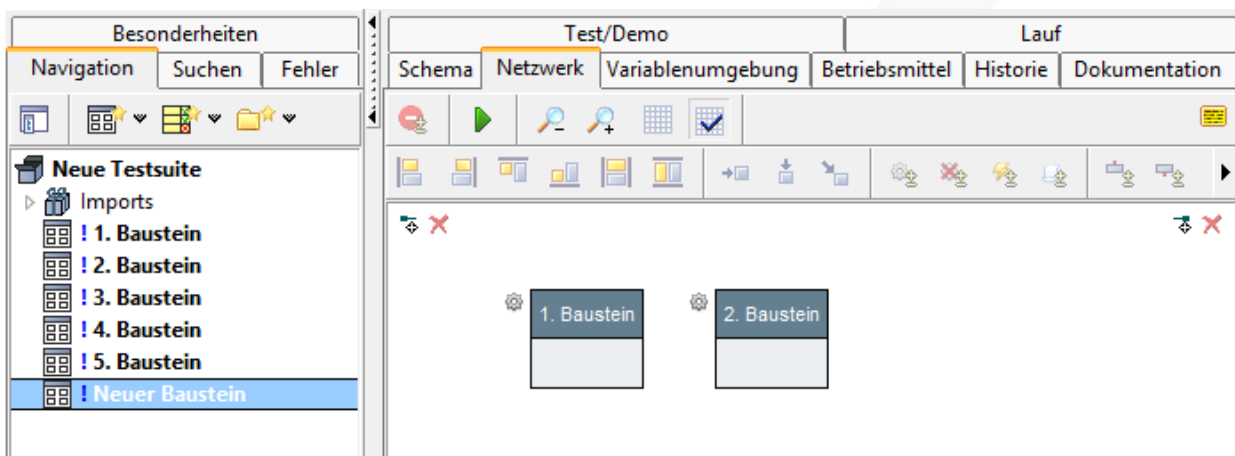
Durch den vorgeschalteten „Random [Integers]“ Baustein wird der erstellte „4. Baustein“ bei jedem Durchlauf mit anderen Werten versorgt, was zu unterschiedlichen Resultaten führt. Dies können Sie am besten sehen, wenn Sie den Baustein unter „Test/Demo“ mehrfach ausführen.



Übernehmen Sie nun den Baustein und benennen Sie ihn in „5. Baustein“ um.

Sequenzielle Ausführung von Schritten

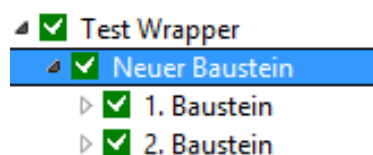
Bei einem Diagramm ohne Datenflüsse (also mit ausschließlichen Kontrollflüssen) werden Triggerein- mit Triggerausgängen verbunden, wodurch ein reines Flussdiagramm entsteht. Zur Vorbereitung legen Sie einen neuen, zusammengesetzten Baustein durch Klicken der Schaltfläche  an. Ziehen Sie per Drag & Drop die Bausteine 1 und 2 in dessen Netzwerk damit Sie folgendes Bild erhalten:



Selektieren Sie den Schritt „1. Baustein“. Durch halten der *Shift*-Taste können Sie den Schritt „2. Baustein“ zur Selektion hinzufügen. Beide Schritte sind jetzt in der Reihenfolge selektiert, in der diese sequenziell ausgeführt werden sollen. Durch Klicken der Schaltfläche  wird jetzt eine Kontrollflussverbindung von Schritt 1 zu Schritt 2 eingerichtet.



Nach dem Ausführen des Bausteins im Test Fenster erhalten Sie folgendes Bild. Wie Sie erkennen können, wurde der Schritt „1. Baustein“ vor Schritt „2. Baustein“ ausgeführt.

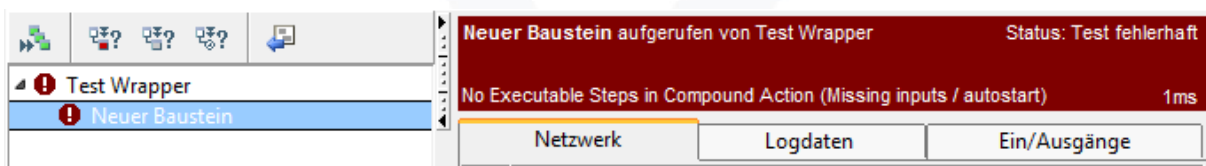


Zum Löschen einer Verbindung wird diese selektiert und anschließend die Taste *Entfernen* oder die Schaltfläche  betätigt.

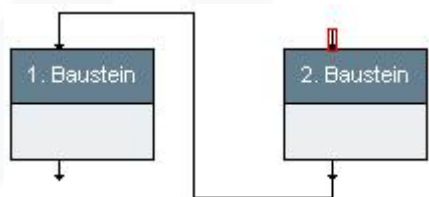
Löschen Sie nun die bestehende Kontrollflussverbindung und richten Sie den Kontrollfluss in umgekehrter Richtung ein. Gehen Sie dazu wie oben beschrieben vor, nur dass Sie dieses Mal den Schritt „2.Baustein“ vor Schritt „1.Baustein“ selektieren.



Nach dem Ausführen des Bausteins erhalten Sie nun folgendes Fehlerbild:



Wie Sie im Netzwerk Editor erkennen können, ist der Schritt „2.Baustein“ durch ein *rotes Ausrufezeichen* markiert. Er ist also nicht bereit zur Ausführung.

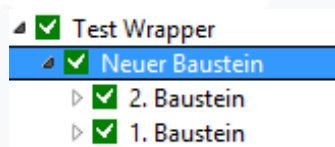


Der Schritt besitzt einen offenen Trigger-Eingangspin. Durch Verbinden des Triggereingangs mit einem beliebigen anderen Ausgangspin kann die Aktion des Schrittes ebenfalls ausgelöst werden. Löschen Sie in unserem Fall den Triggereingang durch Selektion und betätigen Sie dann die Taste *Entfernen*.

Um einen Schritt zur Ausführung zu bringen, muss dieser mindestens einen verbundenen nicht-Parameterpin oder die Autostart-Eigenschaft besitzen. Diese wird durch ein kleines Zahnrad dargestellt. Um einen Schritt als *Autostart* zu *markieren*, selektieren Sie diesen und klicken auf die Schaltfläche .



Nach dem Ausführen des Bausteins unter „*Test/Demo*“ erhalten Sie folgendes Bild. Hier wurde der Schritt „*2. Baustein*“ vor Schritt „*1. Baustein*“ ausgeführt.

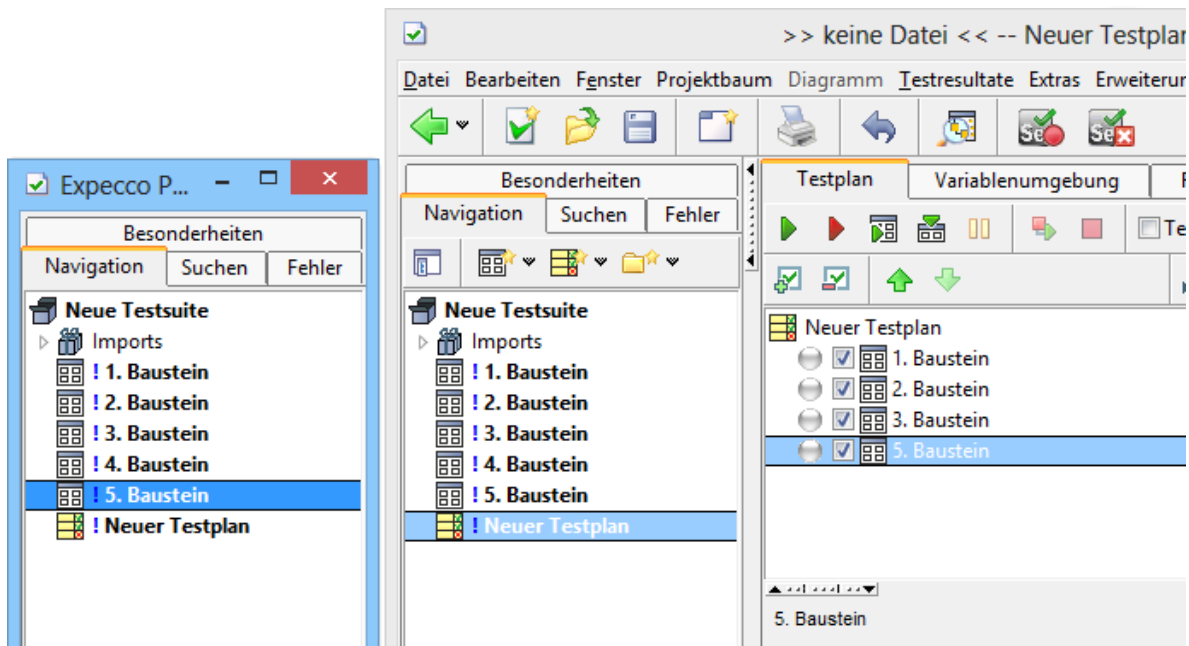


Erstellen eines Testplans

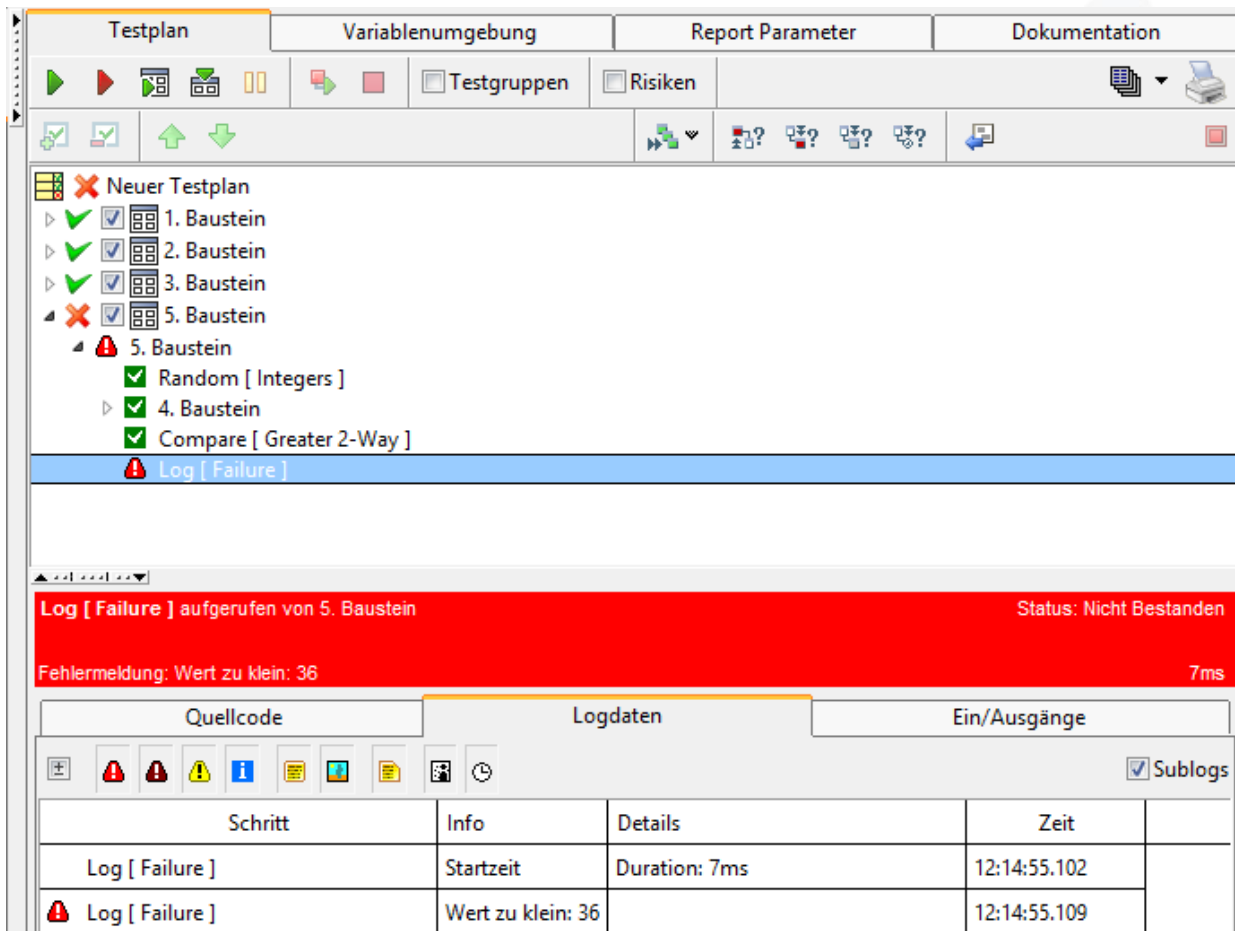
In expecco wird eine Sammlung mehrerer Testfälle als Testplan bezeichnet. Zur Durchführung eines Testplans werden alle Einzeltestfälle sequenziell ausgeführt.

Klicken Sie in der Toolbar  die Schaltfläche  um einen neuen Testplan zu erzeugen.

Ziehen Sie nun per Drag & Drop die neu erzeugten Bausteine, außer den 4. Baustein, in den Testplan-Editor. Sie erhalten folgendes Bild:



Übernehmen Sie den Testplan und führen Sie ihn durch Betätigen der Schaltfläche  aus.
Sie erhalten folgendes Bild:



The screenshot shows the eXept software interface with the 'Testplan' tab selected. The left sidebar displays a test plan structure with five building blocks ('Baustein'). The fifth building block is expanded, showing a failure log entry: 'Log [Failure]'. The main area displays a red banner with the message: 'Log [Failure] aufgerufen von 5. Baustein' and 'Status: Nicht Bestanden'. Below this, a table shows the execution details for the failure log entry.

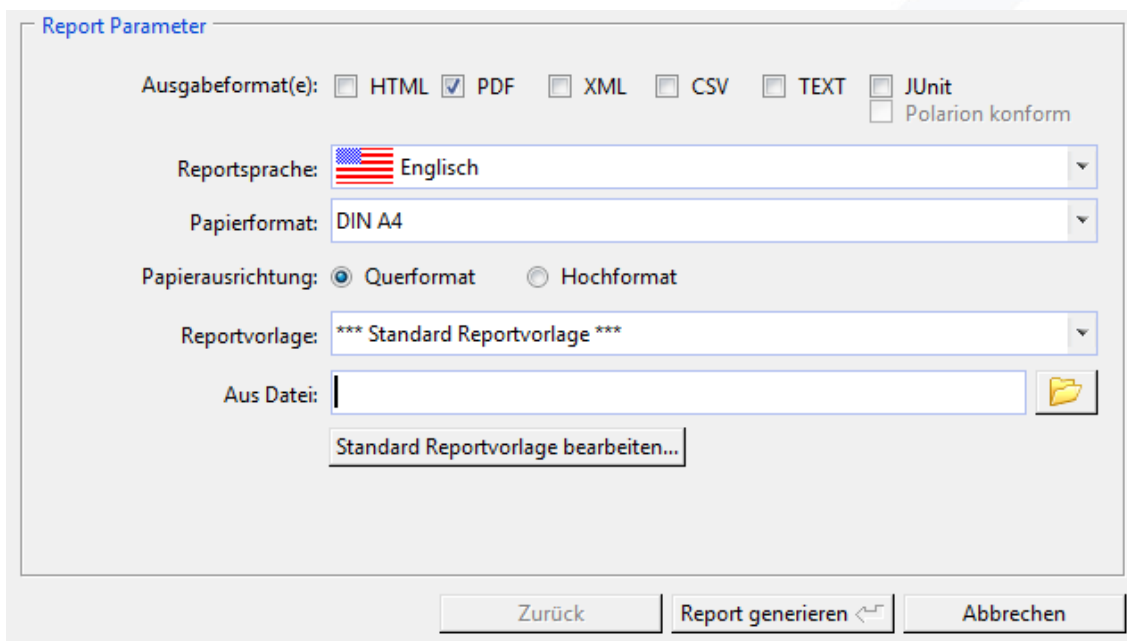
Schritt	Info	Details	Zeit
Log [Failure]	Startzeit	Duration: 7ms	12:14:55.102
Log [Failure]	Wert zu klein: 36		12:14:55.109

Da der „5. Baustein“ intern mit zufällig erzeugten Eingabewerten arbeitet, erhalten Sie bei mehrfacher Ausführung des Testplans unterschiedliche Ergebnisse.

Erstellen eines Berichts (Report)

Zur Vorbereitung führen Sie den Testplan aus. Um einen detaillierten Bericht über Ihre Testausführung zu erstellen, klicken Sie auf die Schaltfläche .

Das Dialogfenster „*Reportgenerierung / Druck*“ wird geöffnet. Vor der Erzeugung Ihres Berichts müssen noch einige Einstellungen festgelegt werden:



Report Parameter

Ausgabeformat(e): ☐ HTML ☒ PDF ☐ XML ☐ CSV ☐ TEXT ☐ JUnit
☐ Polarion konform

Reportsprache:  Englisch

Papierformat: DIN A4

Papierausrichtung: ☒ Querformat ☐ Hochformat

Reportvorlage: *** Standard Reportvorlage ***

Aus Datei: 

[Standard Reportvorlage bearbeiten...](#)

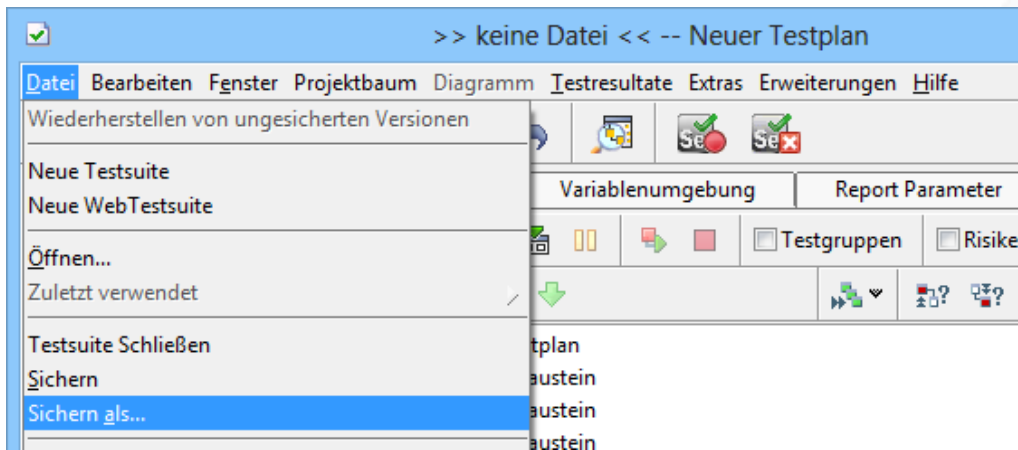
Zurück Report generieren Abbrechen

- **Ausgabeformat:**
Die Berichte können in verschiedenen Ausgabeformaten erzeugt werden. Eine Mehrfachauswahl ist möglich.
- **Reportsprache:**
Einstellung der Sprache für den Bericht festlegen.
- **Papierformat:**
Ausschließlich beim Ausgabeformat *PDF* wird hier das Papierformat spezifiziert. Für alle anderen Formate wird diese Einstellung ignoriert.
- **Reportvorlage:**
Über Reportvorlagen lassen sich Inhalt und Erscheinungsbild des Berichts bestimmen.

Klicken Sie auf die Schaltfläche „*Report generieren*“ um die Berichtserstellung zu starten. Nach Erstellung wird die Datei in einem Browser geöffnet.

Speichern der Testsuite

Um die neu erzeugte Testsuite zu speichern, klicken Sie auf das Menü „Datei“ → „Sichern als...“. Ein Dateibrowser wird geöffnet.



Schlusswort

In unserem Beispiel haben wir mit Absicht einfache Bausteine aus der Standard Bibliothek ausgewählt um Ihnen die Arbeitsweise mit **expecco** näher zu bringen.

Allerdings sind diese Beispiele insofern unrealistisch, als dass kein *System Under Test* angebunden, mit Signalen versorgt oder geprüft wurde. Da die Anbindungen an das zu testende System oft geräte- und protokollspezifisch erfolgt, sind diese Anbindungen nicht Teil des expecco Basissystems, sondern werden über eine Reihe von Bibliotheken und Plugins realisiert. Diese können Sie einerseits von eXept oder von Drittanbietern erhalten, andererseits aber auch selbst anlegen und pflegen.

Informationen zu spezifischen Protokollbausteinen (TCP/IP, DLL-Calls, FTP, ...) und Anbindungen an diverse Geräte zur Kommunikation mit dem *System Under Test* finden Sie in der Dokumentation zu Bibliotheken und Plugins.

Sie finden diese Informationen im Menü unter:

- Hilfe
- expecco Wiki