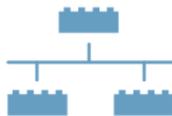


# Tutorial AristaFlow BPM Suite

## Next Generation Business Process Management



Geschäftsprozesse flexibel  
automatisieren



Anwendungen prozessorientiert  
orchestrieren



Human Workflows ohne  
Programmierung realisieren



Prozess-Steuerung nahtlos  
integrieren



# Modellieren, Validieren, Ausführen und Überwachen eines Urlaubsantragsprozesses mit



Die Freigabe von Urlaubsanträgen ist ein klassisches Beispiel für einen einfachen Freigabeprozess und wird deshalb gerne herangezogen, um die grundlegende Funktionsweise eines Workflow-Management-Systems kennenzulernen.

Im nachfolgenden Tutorial werden die folgenden Workflow-Schritte im Urlaubsantrag verwendet:

- Schritt 1: Mitarbeiter „Meier“ füllt einen Urlaubsantrag aus und schickt ihn an seinen Vorgesetzten Schulz.
- Schritt 2: Vorgesetzter „Schulz“ prüft den Urlaubsantrag und genehmigt diesen oder lehnt ihn ab.
- Schritt 3: Mitarbeiter „Meier“ wird informiert, ob der Urlaub genehmigt oder abgelehnt wurde. Bei Genehmigung tritt er den Urlaub an.
- Schritt 4: Der Sachbearbeiter „Schmidt“ aktualisiert das Urlaubskonto des Mitarbeiters „Meier“.
- Schritt 5: Bei Ablehnung kann „Meier“ den Antrag zurückziehen oder ihn ändern und erneut einreichen. Wenn der Antrag zurückgezogen wird, wird der Prozess beendet.
- Schritt 6: Bei erneuter Antragstellung wird der Freigabe-Prozess wieder von vorne gestartet.

## Umsetzung des Urlaubsantrags mit der AristaFlow BPM Suite

Die Umsetzung von Prozessen in der AristaFlow BPM Suite erfolgt im Wesentlichen mit den folgenden zwei Editoren.

- **Org.Model Editor**
- **Process Template Editor**

Damit die Editoren gestartet werden können, muss zunächst der AristaFlow-Server gestartet werden!

Den AristaFlow Server starten Sie über das Startmenü  
**AristaFlow BPM Suite -> Server with console**

oder alternativ im Installationsverzeichnis unter dem Pfad.

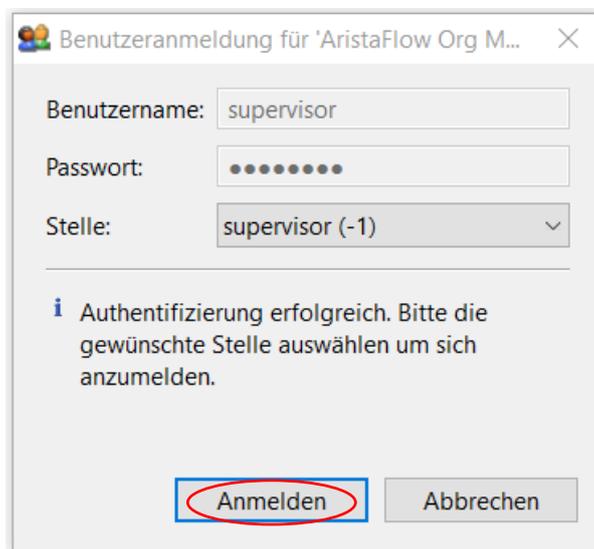
**...\AristaFlow-Chronos\Server\bin\server.bat**

# Anlegen der beteiligten Personen/Rollen und Abteilungen im Org.Model Editor

Starten Sie den ARISTAFLOW-*OrgModelEditor* über das Startmenü  
**AristaFlow BPM Suite -> Org.Model Editor**

oder alternativ im Installationsverzeichnis unter dem Pfad  
**...\\aristaflow-orgmodeleditor\\OrgModelEditor.exe**

Authentifizieren Sie sich im erscheinenden Login-Dialog mit einem gültigen Benutzernamen und zugehörigem Passwort (im Auslieferungszustand mit **supervisor/password**).



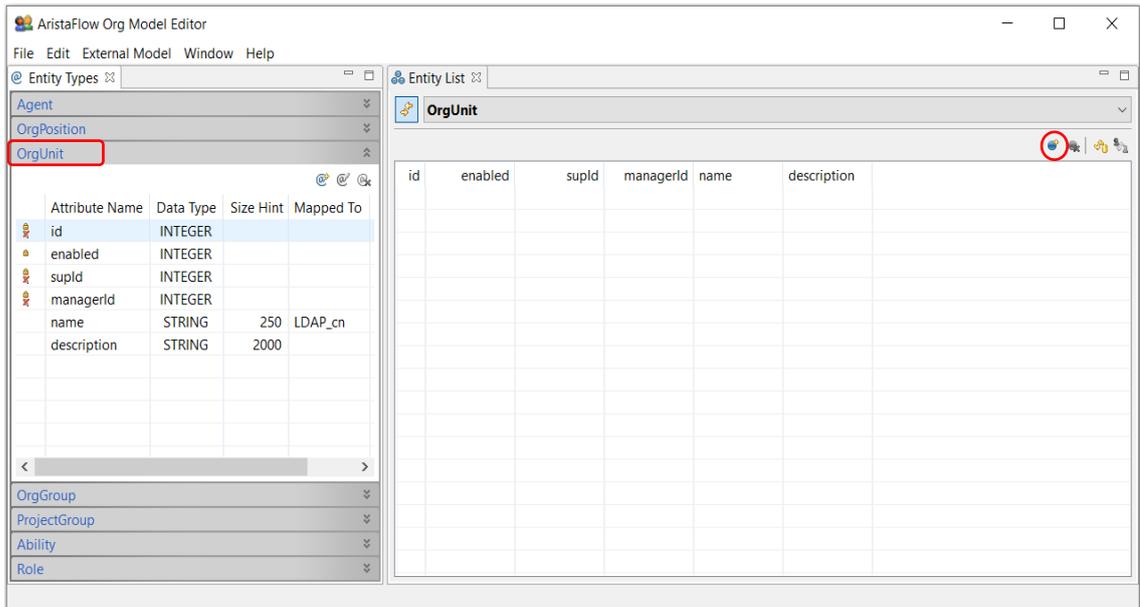
Nach der Authentifizierung öffnet sich der *OrgModelEditor*, mit dem das gesamte Organisationsmodell eines Unternehmens verwaltet werden kann.

Diesen Editor benutzen wir, um die beteiligten Personen, ihre Abteilungen und Stellen/Rollen anzulegen.

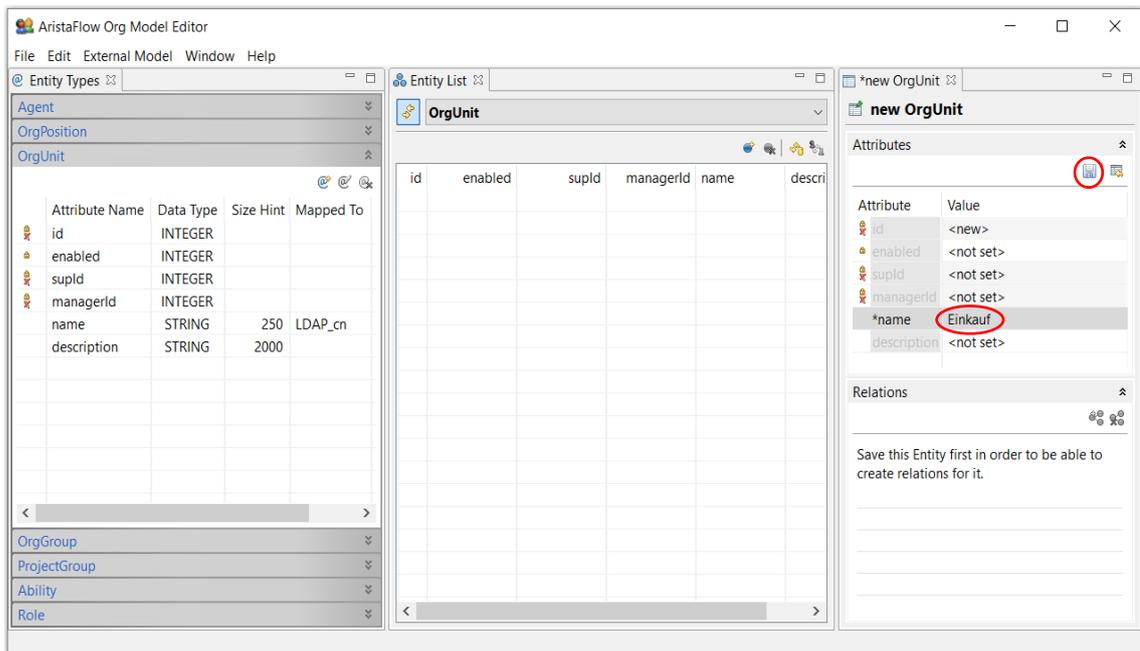
In unserem Beispiel-Urlaubsantragsprozess benötigen wir „Meier“ als Mitarbeiter (*Agent*) auf der Stelle (*OrgPos*) „Mitarbeiter“ der Abteilung bzw. Organisationseinheit (*OrgUnit*) „Einkauf“. „Schulz“ wird als Vorgesetzter (Manager) der Einkaufsabteilung und „Schmidt“ als Sachbearbeiter der Personalabteilung angelegt.

Für den Urlaubsantrag werden nun Abteilungen, Positionen und Agents nacheinander erstellt und wie folgt eingefügt:

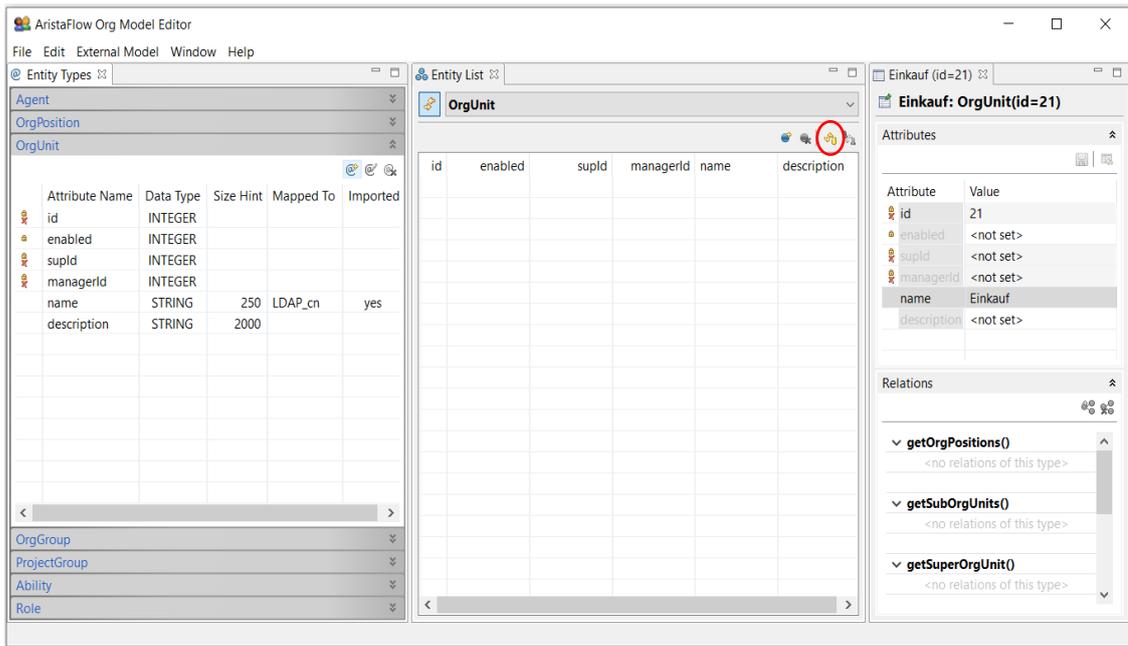
1. Eine Abteilung (*OrgUnit*) „Einkauf“, der der Antragsteller des Urlaubsprozesses angehört.  
→ *OrgUnit* auswählen (links) → *Add Entity* (rechts oben) klicken.



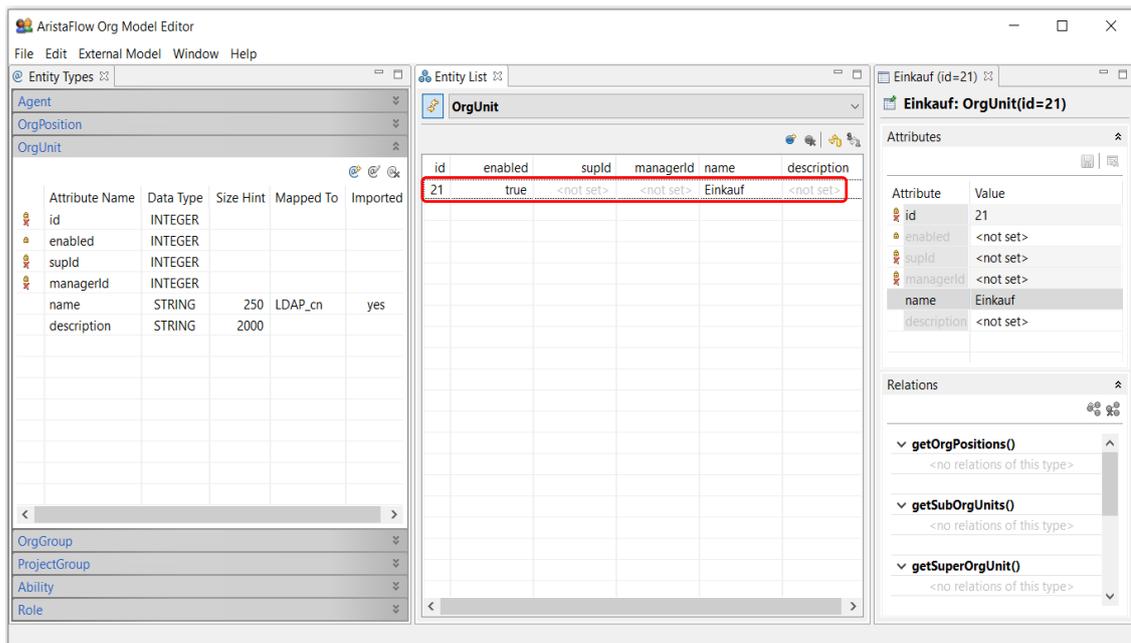
Auf der rechten Seite erscheint ein Dialog, indem die Attribute der neu angelegten Entität eingegeben werden können. Gehen Sie hier auf *name* und geben Sie „Einkauf“ ein. Speichern Sie dann die Entität mit dem Disketten-Symbol (rechts oben).



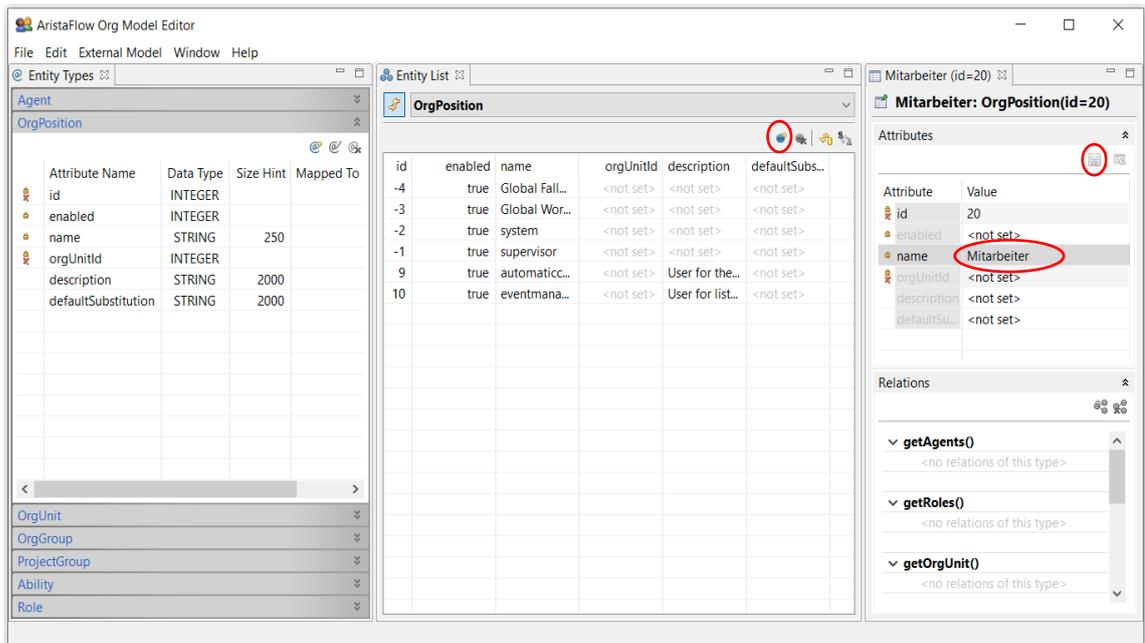
Aktualisieren Sie die Ansicht mit dem Refresh-Symbol im mittleren Tab *OrgUnit* (siehe Screenshot).



Die neu angelegte Abteilung „Einkauf“ erscheint unter *OrgUnit*.

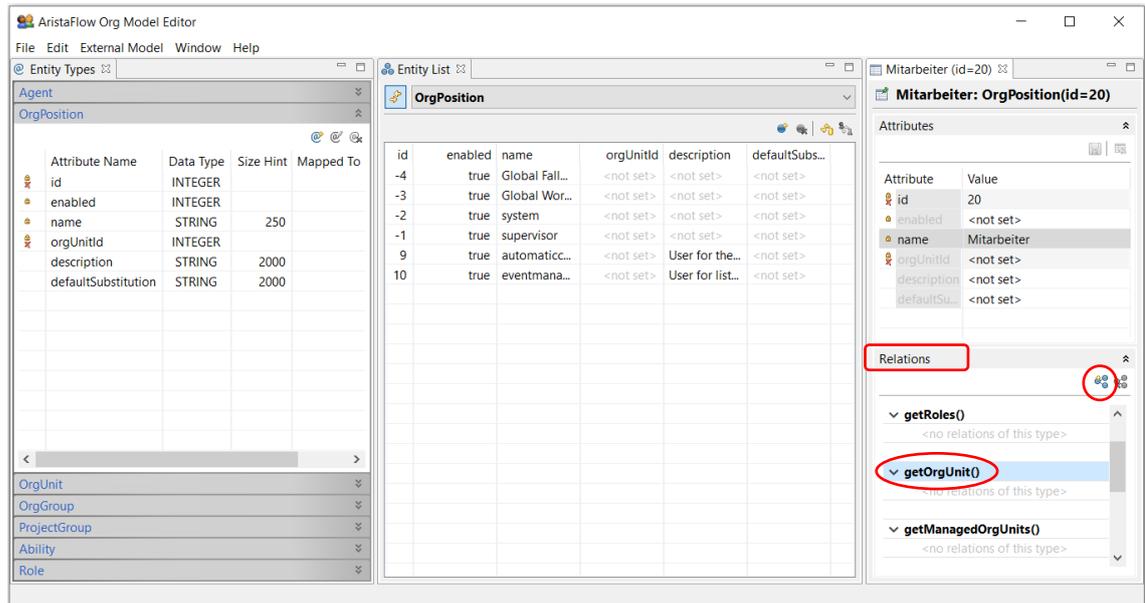


- Anlegen der Stelle (*OrgPosition*) „Mitarbeiter“ und Zuordnung zur in 1 erzeugten *OrgUnit*.  
 → *OrgPosition* auswählen (links) → Add Entity (Mitte) → Attributes name = „Mitarbeiter“ (rechts) eingeben und mit dem Disketten-Symbol speichern.

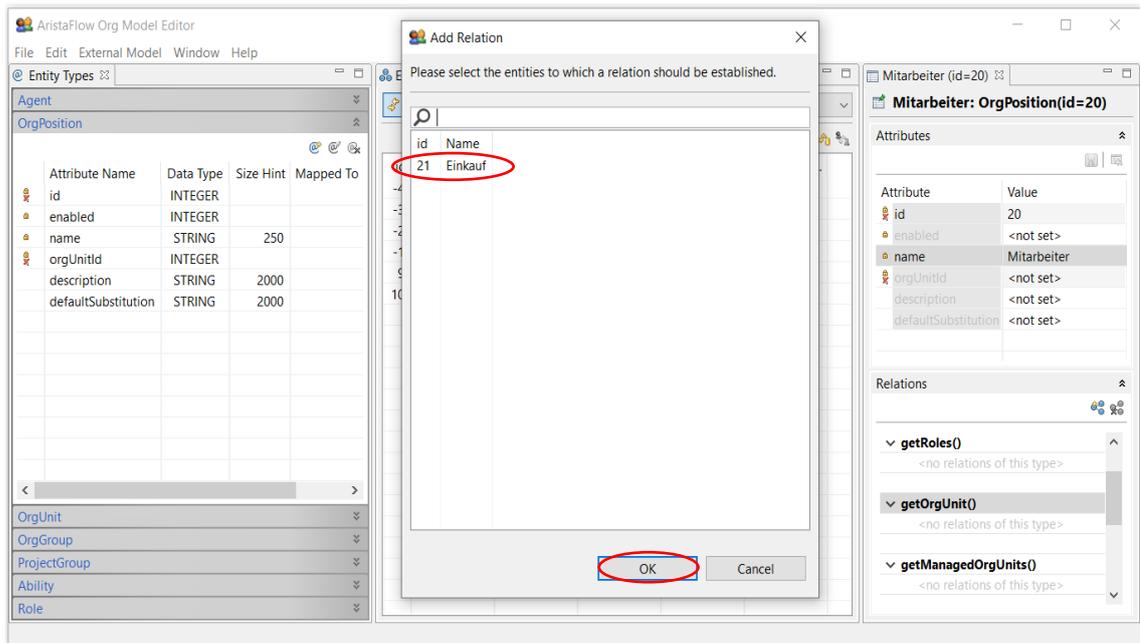


Nun muss die Stelle „Mitarbeiter“ noch der in Punkt 1 erzeugten Abteilung („Einkauf“) zugeordnet werden.

- Relations *getOrgUnit* (rechts unten) anklicken → *Add Relation* anwählen (rechts Mitte)
- oder nur Doppelklick auf *getOrgUnit*.

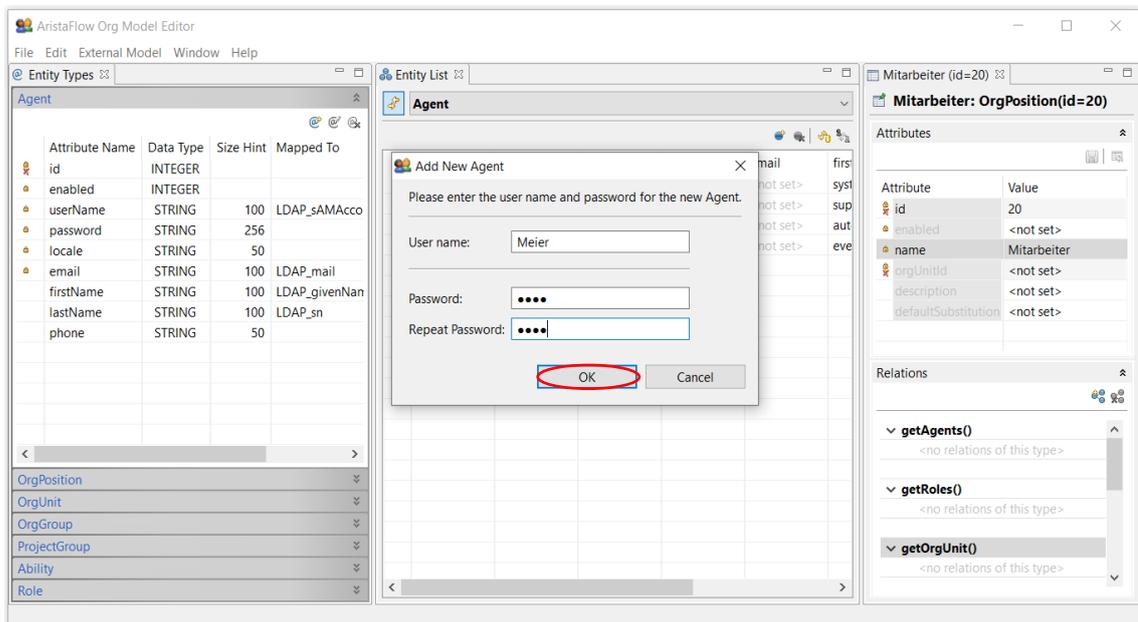


→ Im geöffneten (*Add Relation*-) Fenster „Einkauf“ auswählen und mit OK bestätigen.

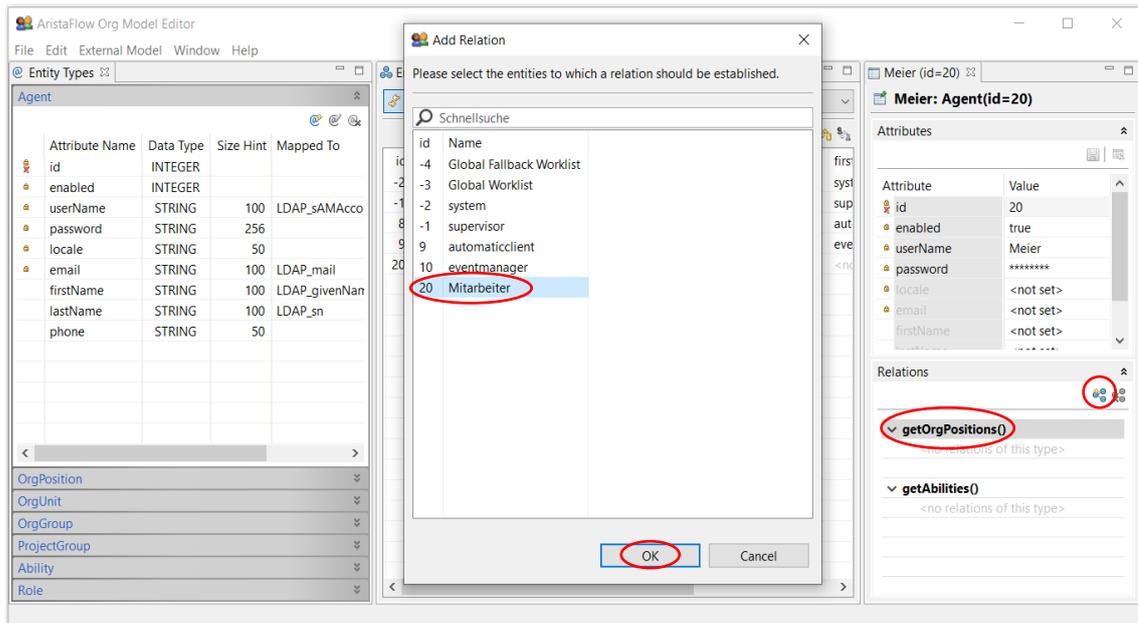


Nach jeder Änderung die Ansicht mit dem *Refresh* Symbol aktualisieren.

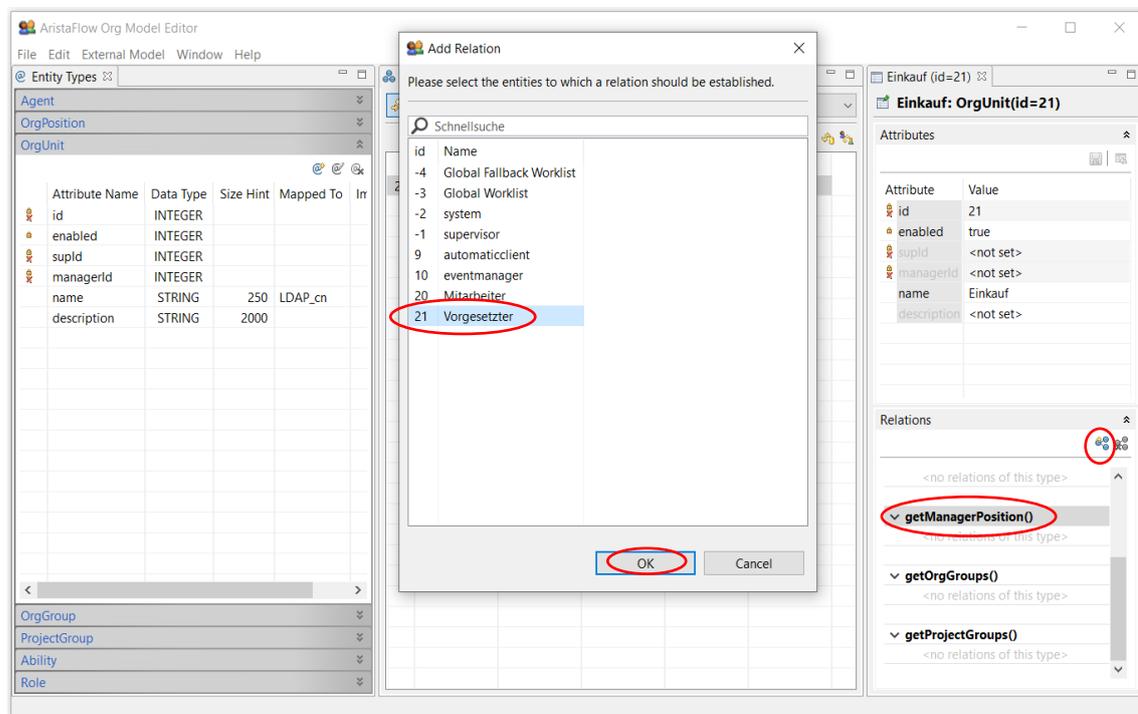
3. Anlegen von Personen (*Agents*) und Zuordnung zur Stelle „Mitarbeiter“.
  - *Agent* auswählen (links) → *Add Entity* klicken → Im erscheinenden *Add New Agent*-Fenster Benutzername und Passwort eingeben (z.B. Meier/test) → mit *OK* bestätigen.



- Zuordnung zur Stelle „Mitarbeiter“
- *getOrgPosition()* anwählen → *Add Relation*. Oder Doppelklick auf *getOrgPositions()*
- Stelle „Mitarbeiter“ auswählen und bestätigen.



4. Anlegen der Rolle (*OrgPosition*) „Vorgesetzter“ und Zuordnung zu „Einkauf“ (analog Punkt 2). Person (Agent) erzeugen (z.B. Schulz/test), die die *OrgPosition* einnimmt (analog Punkt 3).
5. Zuordnung der Rolle „Vorgesetzter“ zur *OrgUnit* „Einkauf“ als Manager.  
 → *OrgUnit* auswählen (links) → Einkauf doppelklicken (Mitte) → Relations *getManagerPosition()* auswählen und *Add Relation* anklicken oder Doppelklick auf *getManagerPosition()* (rechts unten)  
 → „Vorgesetzter“ auswählen und bestätigen.



6. Anlegen der Organisationseinheit (*OrgUnit*) „Personalabteilung“ (analog zu Punkt 1).
7. Anlegen einer Stelle (*OrgPosition*) „Sachbearbeiter“ und Zuordnung zur *OrgUnit* („Personalabteilung“) aus Punkt 6 (analog Punkt 2).

8. Anlegen weiterer Personen (Agents) z.B. „Schmidt“ mit dem Passwort „test“ und Zuordnung zur Rolle „Sachbearbeiter“ (analog zu Punkt 3).

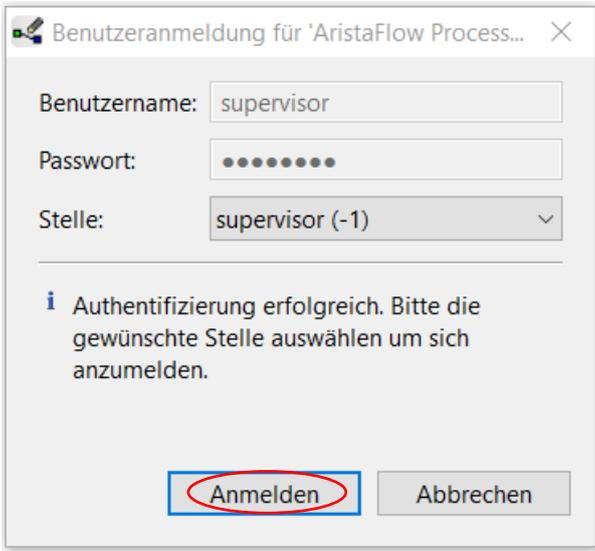
Alle für diesen Prozess notwendigen Abteilungen, Stellen und Personen sind nun angelegt und können im nächsten Schritt bei der Prozessmodellierung im *Process Template Editor* verwendet werden.

## Modellieren des Prozesses

### im AristaFlow Process Template Editor

Starten Sie den des *ARISTAFLOW-ProcessTemplateEditor* über das Startmenü  
**AristaFlow BPM Suite -> Process Template Editor**  
oder im Installationsverzeichnis über den Pfad  
**...\AristaFlow-Chronos\ProcessTemplateEditor\ProcessTemplateEditor.exe**

Authentifizieren Sie sich im erscheinenden Login-Dialog mit einem gültigen Benutzernamen und zugehörigem Passwort (im Auslieferungszustand mit **supervisor/password**).



Nach der Authentifizierung öffnet sich der *Process Template Editor*, mit dem die Prozesse eines Unternehmens graphisch als Workflows abgebildet werden können.

#### Info:

Im Gegensatz zu anderen Systemen, werden Prozessgraphen in AristaFlow nicht nur durch das Zeichnen von Knoten und Kanten erstellt, sondern gleichzeitig auf die Korrektheit hin überprüft. Durch die Anwendung eines wohl definierten Satzes an Änderungsoperationen werden Fehler wie Verklemmungen oder nicht erreichbare Knoten automatisch ausgeschlossen.

Datenflussprüfungen stellen sicher, dass der Informationsfluss zwischen den Prozessschritten fehlerfrei abgebildet wird. Der Process Template Editor lässt abhängig von der gewählten Markierung im Prozessgraph nur bestimmte Änderungsoperationen zur Auswahl zu. Die grüne Färbung markiert den Anfang, die blaue das Ende des Bereichs für die Operationen. Dadurch werden kontextabhängig nur solche Änderungen zugelassen, die einen konsistenten Prozessgraphen wieder in einen neuen konsistenten Graphen überführen. Dieses Konstruktionsprinzip wird als *Correctness by Construction* bezeichnet. Darüber hinaus überprüft der Process Template Editor das erstellte Prozessmodell (Template) durch die Prozessvorlagenverwaltung auf Korrektheit, so dass nur syntaktisch korrekte Modelle auf den Server hochgeladen werden können.

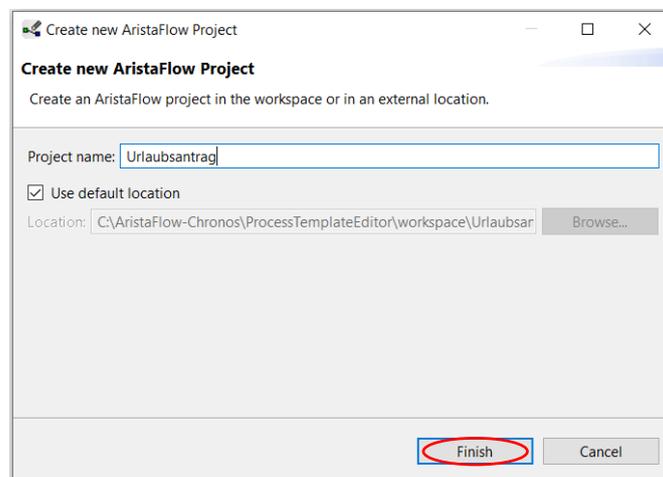
Starten wir mit der Modellierung des Urlaubsantrag-Freigabeprozesses durch Modellierung als gerichteter Prozessgraph im Process Template Editor.

- Dabei wird jeder Arbeitsschritt (wie z.B. Urlaubsantrag ausfüllen, Urlaubsantrag prüfen...) als Knoten im Graph dargestellt.
- Jedem Knoten wird ein Bearbeiter als *Agent* zur Ausführung der Aktivität zugeordnet.
- Verzweigungsentscheidungen werden über eine XOR Verzweigung (entweder-oder-Verzweigung) modelliert. Im Urlaubsantrag wird dies beispielsweise für die Darstellung der Fälle „Genehmigung“ oder „Ablehnung des Antrags“ verwendet.
- Rücksprünge/Schleifen im Prozess werden über einen Loop-Block abgebildet. Die Konfiguration der Schleifenabbruchbedingung erfolgt analog zur XOR-Konfiguration. Im Urlaubsantragsprozess wird dies für den Fall genutzt, dass der Antrag nach Ablehnung geändert und erneut gestellt werden kann. Datenflüsse, also die Kommunikation zwischen den Prozessbeteiligten werden modelliert. Dabei werden Datenelemente (wie z.B. Antragssteller, Urlaubsbeginn, Urlaubsende ...) und ihre Typen festgelegt. Diese werden dann von einzelnen Prozessschritten (Knoten) gelesen oder geschrieben.
- Anschließend müssen Aktivitäten (wie „Urlaubsformular ausfüllen“) durch das *Activity Repository* in den einzelnen Knoten integriert werden. Hierbei handelt es sich um Code-Fragmente, die zur Laufzeit von der Workflow-Engine ausgeführt werden und die aus einem modellierten Prozess einen digitalisierten Prozess machen.

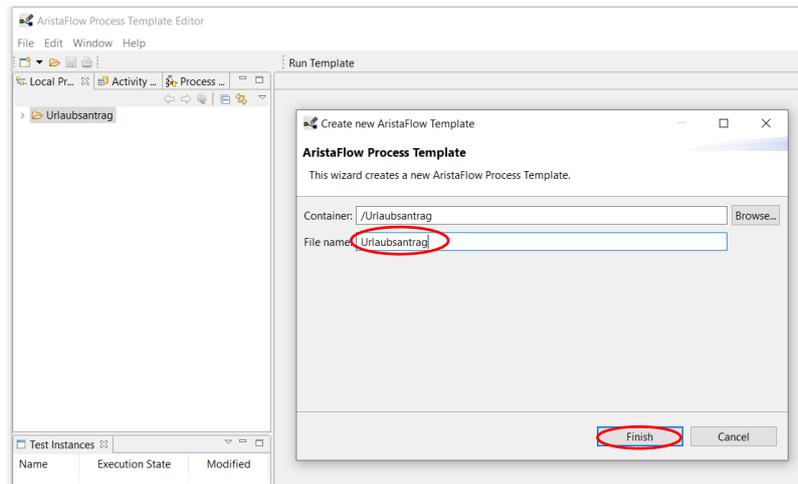
Für die Abbildung des Urlaubsantragsprozesses sind folgende Schritte notwendig:

## 1. Anlegen einer neuen Prozessvorlage

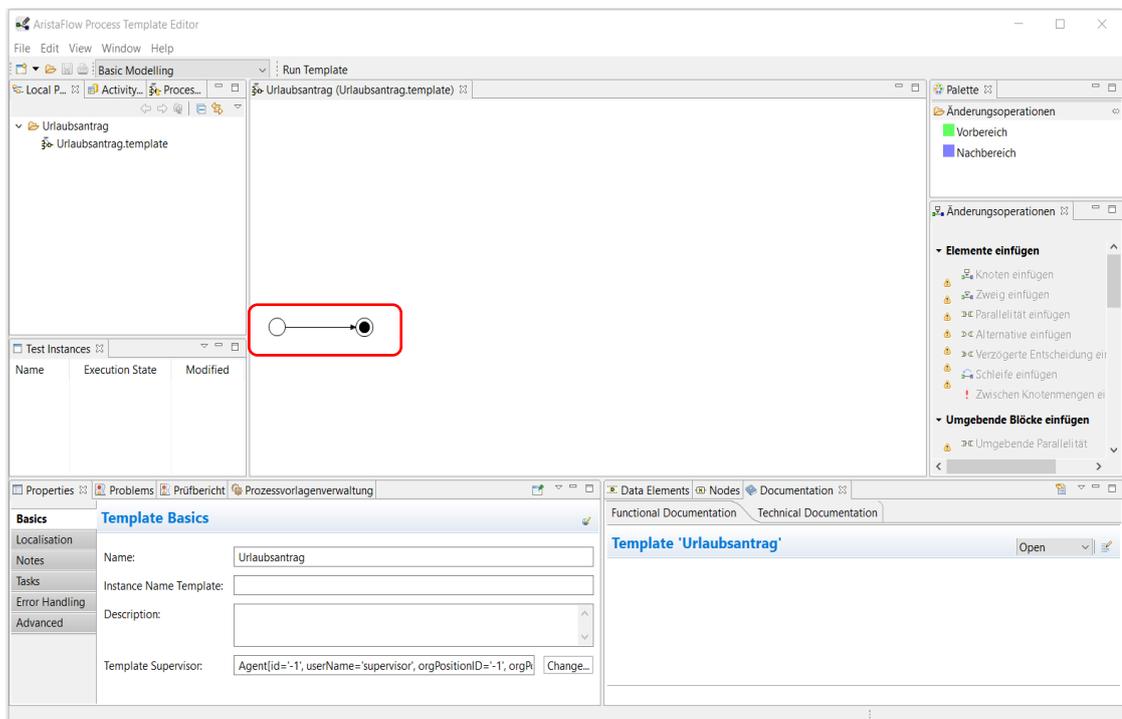
- a. Anlegen eines neuen Projektes (Navigator links oben, Tab *Local Projects*):  
Rechtsklick in weiße Fläche → New → AristaFlow Projekt → „Urlaubsantrag“ im erscheinenden Fenster eingeben und Dialog schließen.



- b. Anlegen einer neuen Prozessvorlage (Template):  
Rechtsklick auf das eben erstellte Projekt „Urlaubsantrag“ → New → AristaFlow-Prozessvorlage → im Dialog unter *File name*: „Urlaubsantrag“ eingeben.



Es erscheint im Editor der Ausgangspunkt für einen neuen Geschäftsprozess: Ein einfacher *Start-* und sein zugehöriger *End-*Knoten.



### **Tipp:**

Mit einem Rechtsklick auf den Editor und Wahl von „Display Options“ öffnet sich ein Dialog, in dem über „Graph figure representation“ (unterste Option) zwischen zwei Darstellungen *BPMN Symbols* und *ADEPT Symbols* der Prozessschritte gewechselt werden kann.

### **Hinweis:**

In diesem Beispiel wurde die Darstellung *BPMN Symbols* verwendet.

## 2. Modellierung der einzelnen Prozessschritte:

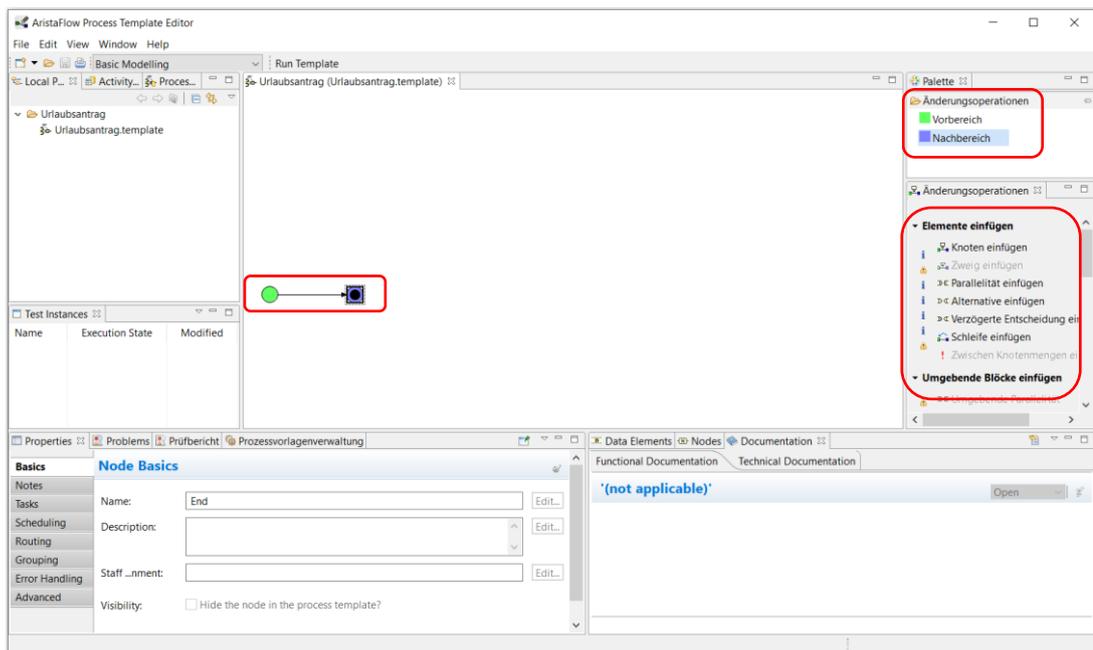
### Schritt 1:

- a. Modelliert wird der Kontrollfluss eines Prozesses durch Selektion des *Vor- und Nachbereichs* im Tab *Palette* (grünes und blaues Kästchen; rechts oben) mit anschließender Wahl der *Änderungsoperation* (rechts Mitte).

Beispiel: Einfügen des ersten Schritts „Urlaubsantrag ausfüllen“:

Zuerst Anklicken des Vorbereichs (des grünen Kästchens), dann des Start-Knotens (Knoten färbt sich grün)

Zuerst *Nachbereich* (blauer Kästchen) anwählen und danach den End-Knoten klicken (Knoten färbt sich blau).

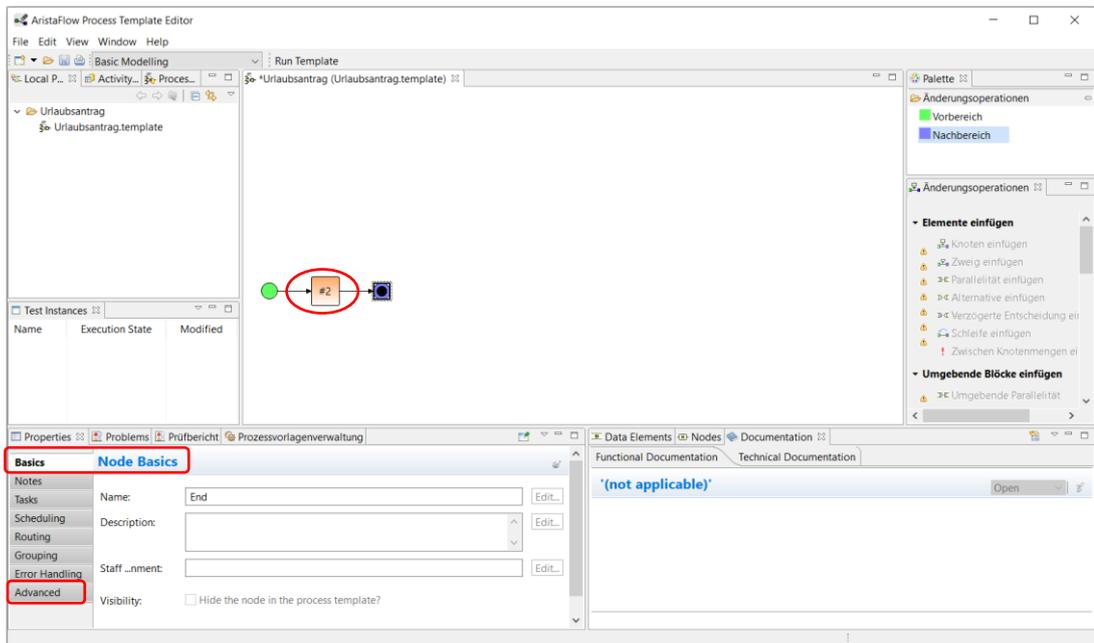


In der Ansicht „Änderungsoperationen“ (rechts Mitte) werden automatisch diejenigen Operationen aktiviert, die mit der gewählten Selektion und in der gegebenen Situation anwendbar sind. *Elemente einfügen* → *Knoten einfügen* auswählen → ein neuer Prozessschritt (Knoten) wird automatisch eingefügt (siehe Screenshot).

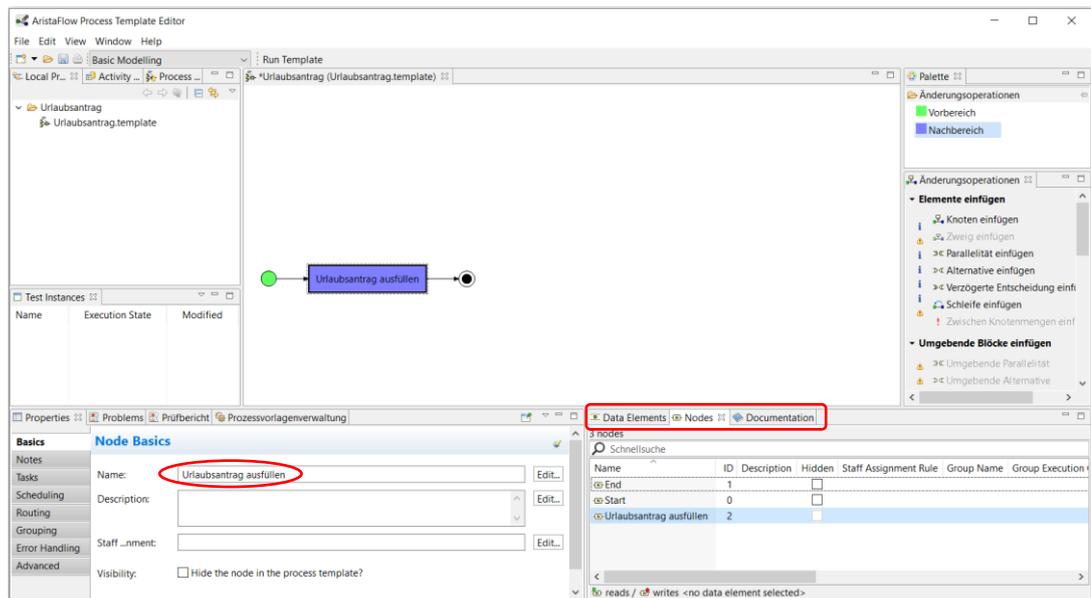
#### **Tipp:**

Tooltips auf nicht aktivierten Änderungsoperationen geben Aufschluss über den Grund der momentanen Inaktivität.

- b. Die Eigenschaften eines Prozessschrittes werden über die *Properties View* (links unten) geändert. Hierbei gibt es zwei Modi:
  - Im *Basics Tab* können die wichtigsten Eigenschaften angepasst werden.
  - Das *Advanced Tab* (unterster Tab) hingegen enthält alle Eigenschaften eines Prozessschrittes und wird nur bei komplexeren Prozessschritten benötigt.

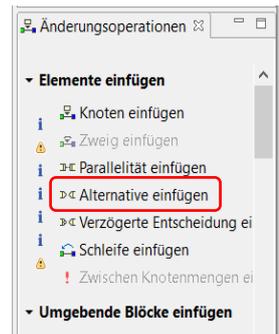
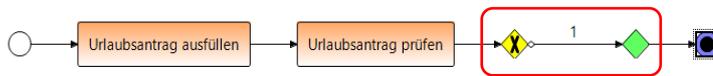


Markieren Sie den erstellten Knoten, gehen Sie zum *Basics* Tab und geben Sie unter „Name“ den Namen des Knotens an (hier „Urlaubsantrag ausfüllen“). Zusätzliche Informationen über den Graphen finden Sie rechts unten, unter den Tabs *Data Elements*, *Nodes* und *Documentation*.



## Schritt 2:

Modellieren Sie den zweiten Prozessschritt „Urlaubsantrag prüfen“ analog zu a und b. Nach diesem Knoten wird der Antrag genehmigt oder abgelehnt. Für diese Verzweigung wird ein XOR-Knoten benötigt. Dies können Sie bei den Änderungsoperationen mit *Alternative einfügen* realisieren. Es entstehen zwei Knoten; Verzweigungs- und Vereinigungsknoten verbunden mit einem leeren Zweig.



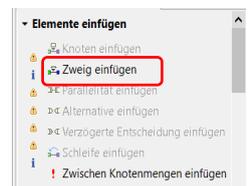
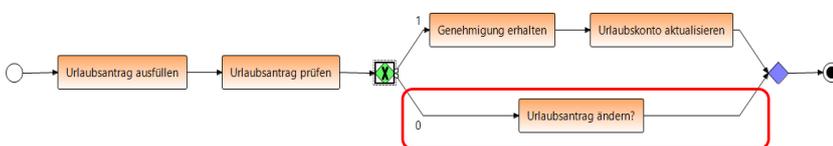
**Schritt 3 und 4:**

Auf diesem Zweig fügen Sie die zwei Knoten „Genehmigung erhalten“ (als Schritt 3) und „Urlaubskonto aktualisieren“ (als Schritt 4) analog zu a und b ein.



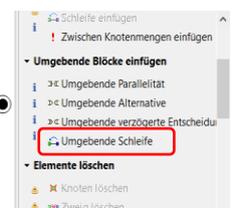
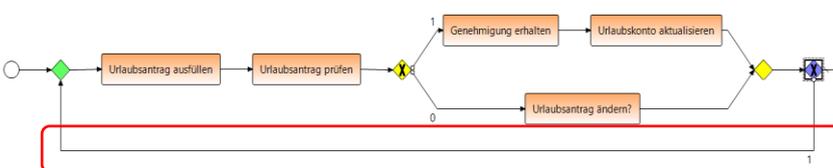
**Schritt 5:**

Für den Fall der Ablehnung wird noch ein Zweig benötigt. Das wird durch *Zweig einfügen* ermöglicht. Auf diesem zweiten Zweig bilden Sie den Knoten „Urlaubsantrag ändern?“ ab.



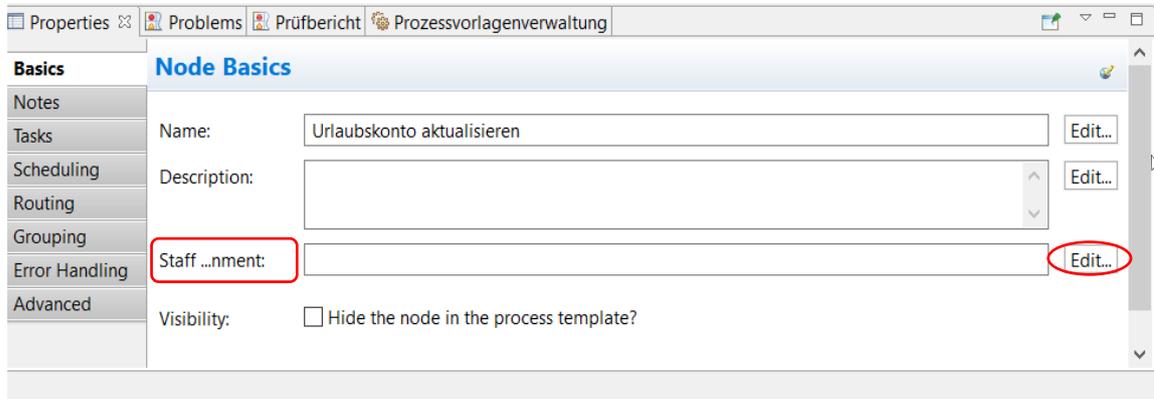
**Schritt 6:**

Im Fall einer erneuten Antragsstellung soll der Workflow wieder zum Anfang zurück. Diesen Loop-Block können Sie durch eine *Umgebende Schleife* über den ersten Knoten *Urlaubsantrag ausfüllen* (als Vorbereich) und den letzten XOR-Vereinigungsknoten (als Nachbereich) realisieren.



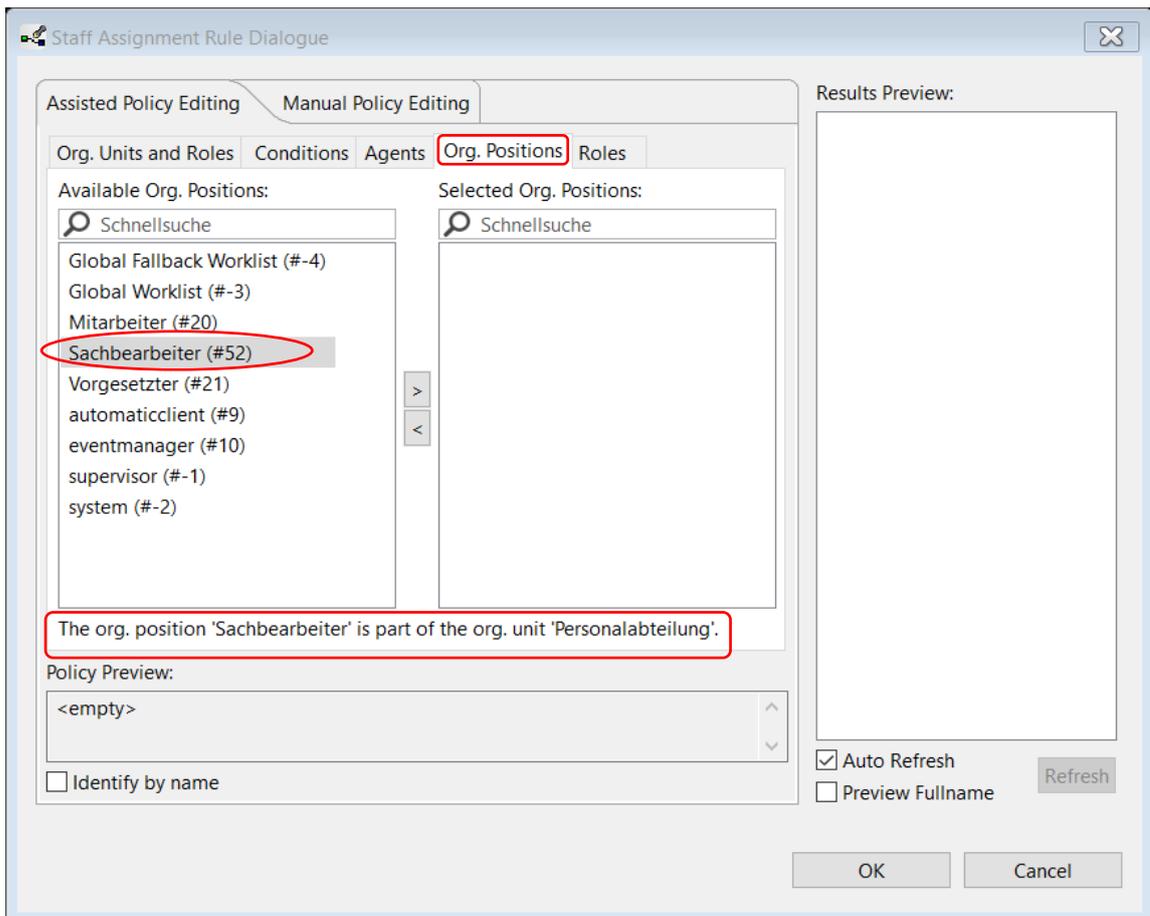
### 3. Zuweisung von Prozessbeteiligten (Bearbeitern) zu den Prozessschritten

- a. Selektieren Sie einen Prozessschritt → Anwählen der *Staff Assignment Rule* in der *Properties View* (links unten)  
→ Öffnen des *Staff Assignment Rule Wizards* durch Klick auf *Edit...*

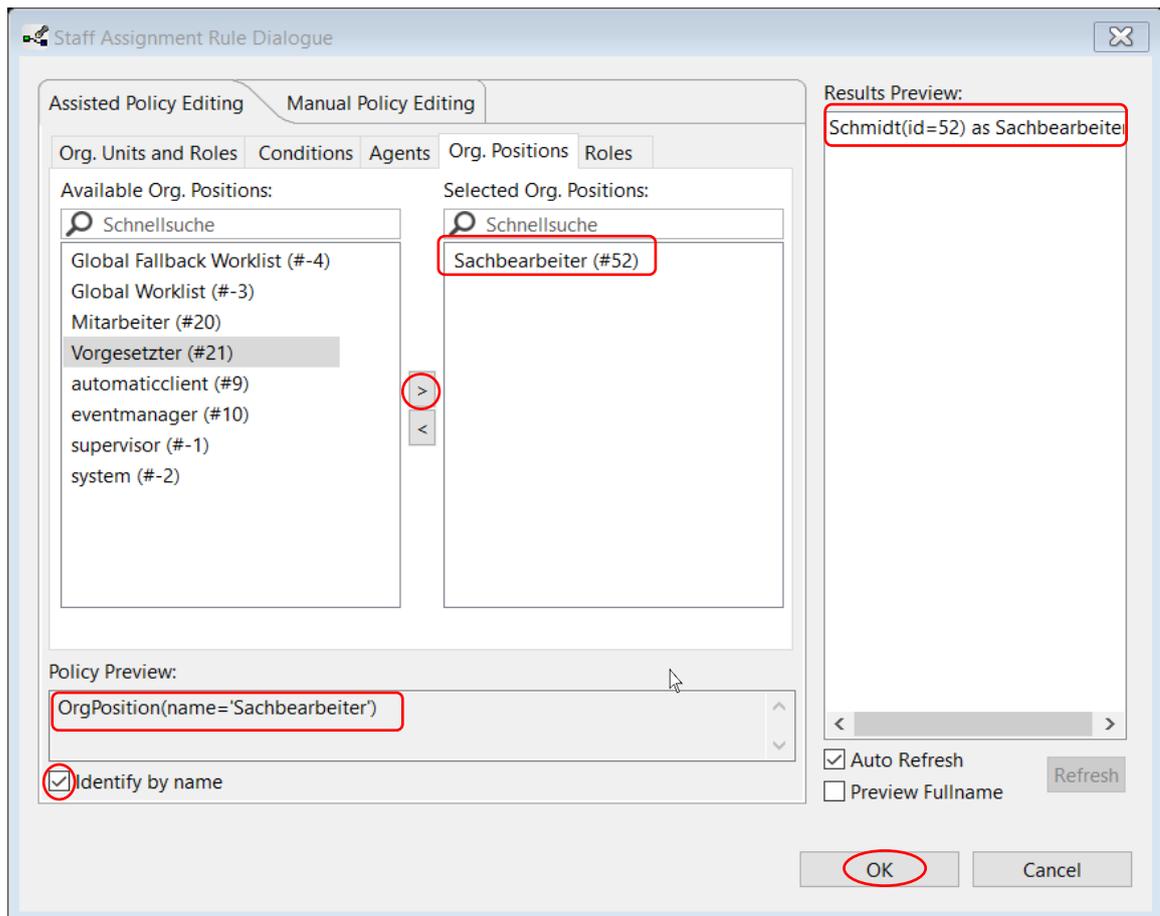


Beispiel: Knoten „Urlaubskonto aktualisieren“

→ Tab *OrgPositions* anwählen und „Sachbearbeiter“ selektieren. Die Organisationseinheit (*OrgUnit*), zu der diese Stelle gehört wird unten angezeigt.



→ In der Mitte des Fensters das Zeichen für die Zuweisung „>“ anklicken → unter *Results Preview* (rechts) werden alle Personen aufgelistet, die diese Stelle besetzen. Mit einem Häkchen bei *Identify by name*, wird der Bearbeiter unter *Policy Preview* durch den Namen anstatt der ID identifiziert.



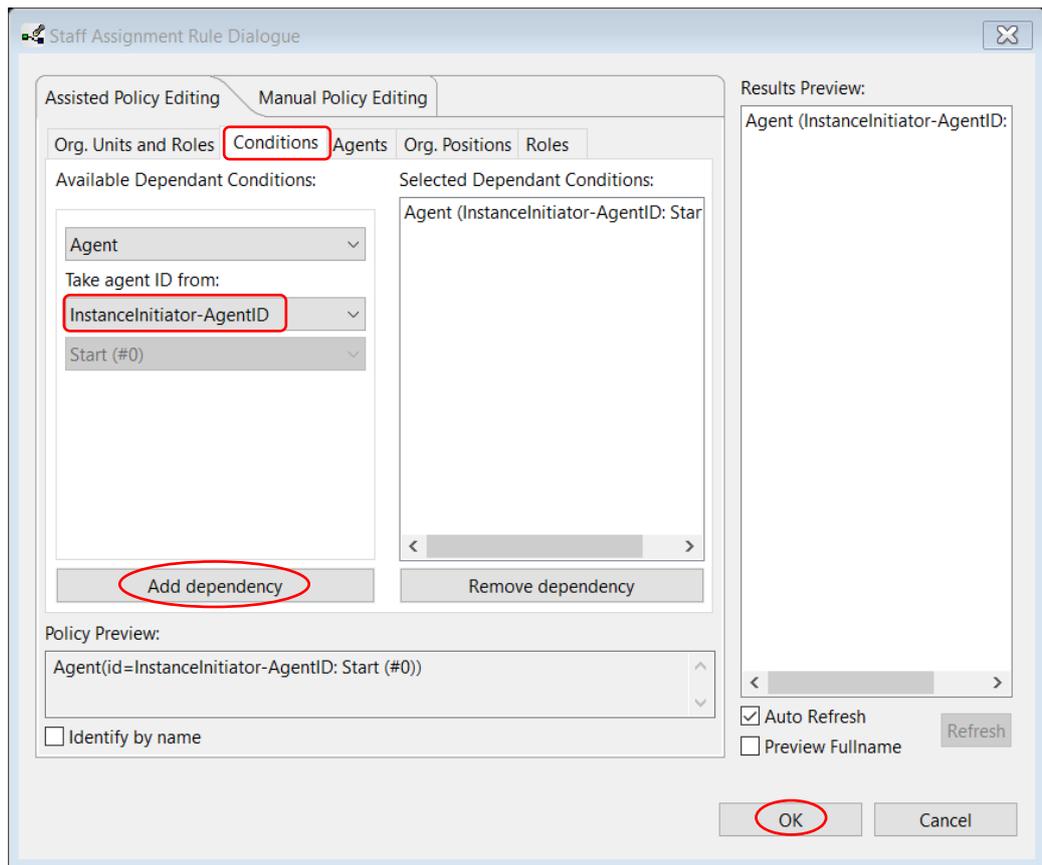
→ mit *OK-Button* schließen.

b. Analoges Zuweisen der Prozessbeteiligten zu den folgenden Prozessschritten:

- i. XOR-Verzweigungsknoten „genehmigt /abgelehnt“: Tab *Agents* → „automaticclient“ (beim automaticclient handelt es sich um einen Klienten der serverseitig automatisch Schritte ausführt, sobald diese in seine Arbeitsliste gelangen).
- ii. XOR-Verzweigungsknoten (LOOP Block) „Neuer Antrag?“: Tab *Agents* → „automaticclient“ (automatischer Schritt).

c. Der *Staff Assignment Rule Wizard* ermöglicht es neben der Zuweisung von Stellen (*OrgPositons*) und Personen (*Agents*) auch komplexe Abhängigkeiten (*dependency*) zwischen den Bearbeitern und den Prozessschritten zu erzeugen. Zum Beispiel:

- i. Prozessschritt „Urlaubsantrag ausfüllen“: Tab *Conditions* → *Take Agent ID from* → „InstanceInitiator-AgentID“ auswählen und *Add Dependency* anklicken. Die Bearbeiterzuordnung ist nun so gewählt, dass immer derjenige Bearbeiter den Schritt in seine Arbeitsliste bekommt, der den Prozess gestartet hat (InstanceInitiator). In diesem Prozess heißt das, dass der Antragsteller (gleichzeitig der Prozessstarter) „Meier“ den ersten Schritt „Urlaubsantrag ausfüllen“ ausführt.



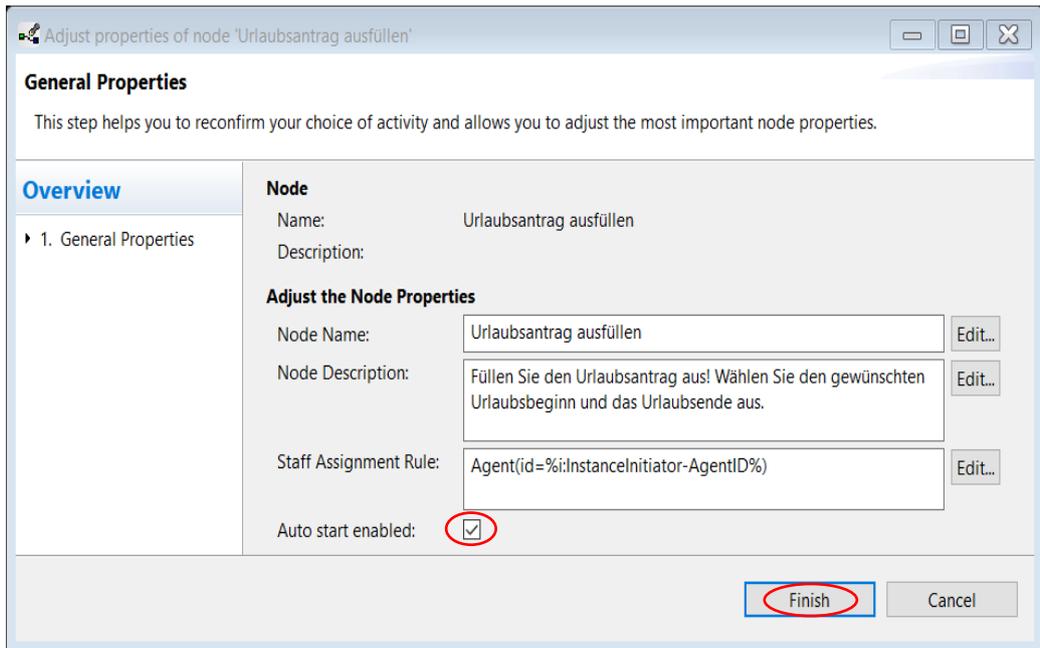
### Tipp:

Da es sich beim Bearbeiter des Schrittes „Urlaubsantrag ausfüllen“ um den Starter der Instanz handelt, kann man die Ausführung dieses Schrittes mit dem Starten der Instanz verknüpfen. Dies führt später bei der Ausführung dazu, dass der Schritt „Urlaubsantrag ausfüllen“ sofort nach Starten der Instanz automatisch ausgeführt wird. Klicken Sie hierzu doppelt auf den Schritt „Urlaubsantrag ausfüllen“ und setzen Sie im erscheinenden Dialog ein Häkchen bei „Auto start enabled“.

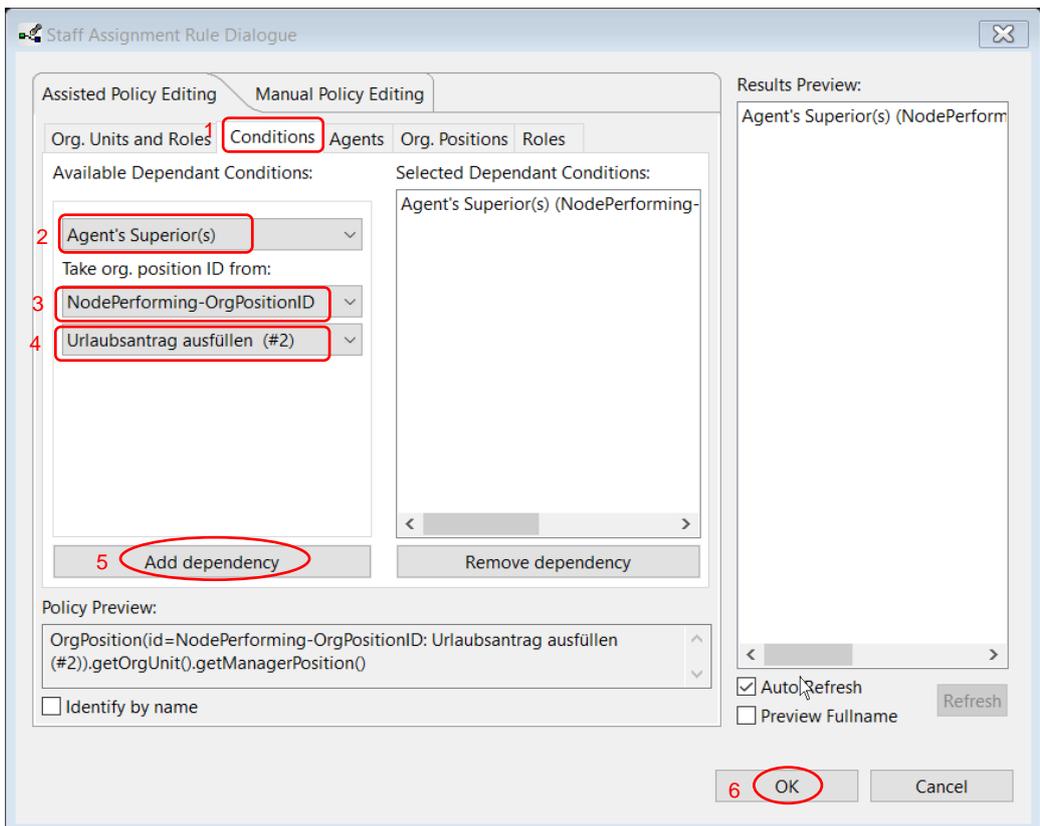


### Hinweis:

Bei komplexen Abhängigkeiten in *Staff Assignment Rule Wizard* können die Agents nicht durch den Namen identifiziert werden, da dies intern zu Uneindeutigkeiten bei der Bearbeiterzuweisung führen kann. Das ist in unserem Beispielsprozess bei „Urlaubsantrag ausfüllen“ (Prozessstarter) und „Urlaubsantrag prüfen“ (Vorgesetzter vom Prozessstarter) der Fall. Da werden die Agents durch ihre IDs identifiziert.



- ii. Analoges Zuweisen der Bearbeiterformeln zu den Prozessschritten „Genehmigung erhalten“ und „Urlaubsantrag ändern“, da diese Schritte auch vom Antragssteller „Meier“ erledigt werden (hier ohne „Auto start enabled“).
- iii. Prozessschritt „Urlaubsantrag prüfen“: Tab *Conditions* → „Agent’s Superior(s)“ auswählen → Als Agent-ID „NodePerforming – OrgPositionID“ auswählen → „Urlaubsantrag ausfüllen“ auswählen und *Add Dependency* anklicken. Die Bearbeiterzuordnung ist nun so gewählt, dass immer der Vorgesetzte des Bearbeiters aus dem Schritt „Urlaubsantrag ausfüllen“ also Schulz den Schritt „Urlaubsantrag prüfen“ in seine Arbeitsliste bekommt.





**Tipp:**

Der *Staff Assignment Rule Wizard* ermöglicht es im *Manual Policy Editing Tab* beliebig komplexe Bearbeiterzuordnungsregeln explizit einzugeben, um somit die gesamte Mächtigkeit des *ARISTAFLOW-OrgModelEditor* nutzen zu können (z. B. Projektgruppen, Fähigkeiten, etc.).

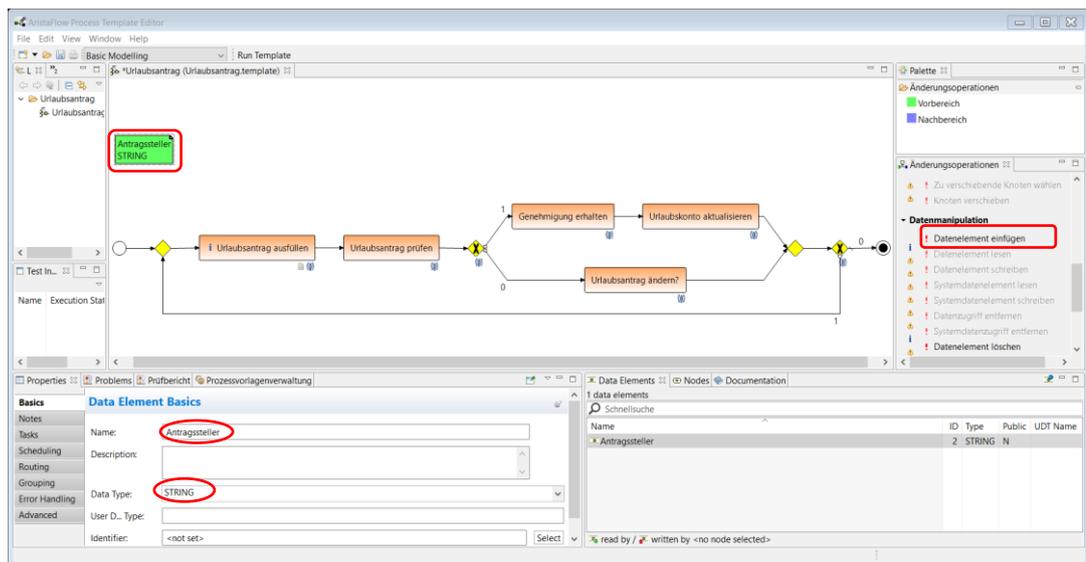
4. Modellieren des Datenflusses / der Kommunikation zwischen den Prozessbeteiligten.

a. Zuerst erstellen Sie die Datenelemente ohne Beachtung des Vor- und Nachbereichs im Editor. Die Modellierung der Zugriffe auf diese losen Elemente erfolgt später.

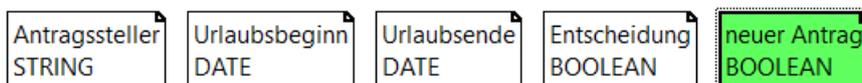
→ Wahl der Änderungsoperation *Datenmanipulation*-> *Datenelement einfügen* (rechts Mitte im Process Template Editor).

→ Ein Datenelement erscheint im Editor. Insgesamt brauchen Sie 5 Datenelemente mit folgenden Eigenschaften, die Sie in der *Properties View* anpassen können.

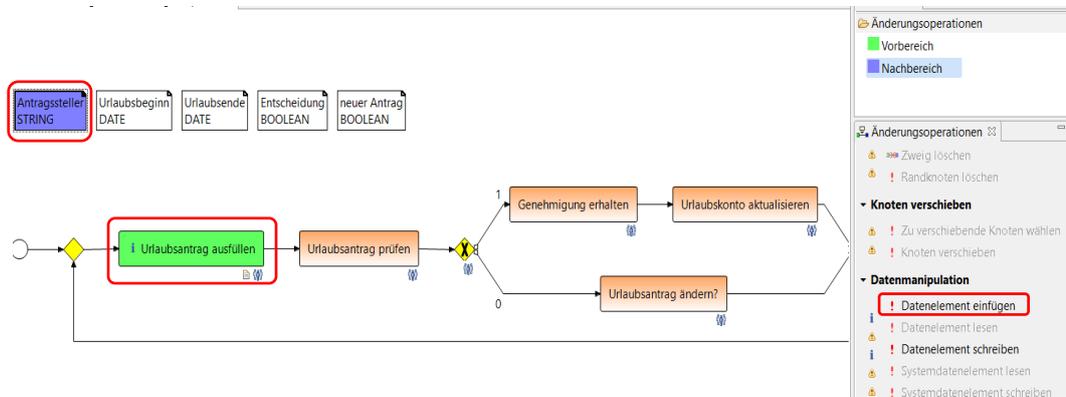
- i. Datenelement Name: „Antragsteller“  
Data Type = String
- ii. Datenelement Name: „Urlaubsbeginn“  
Data Type = Date
- iii. Datenelement Name: „Urlaubsende“  
Data Type = Date
- iv. Datenelement Name: „Entscheidung“  
Data Type = Boolean
- v. Datenelement Name: „neuer Antrag“  
Data Type = Boolean



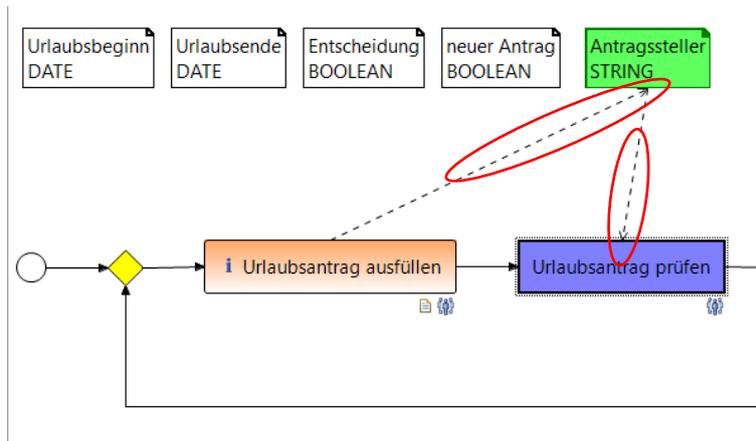
**Das Ergebnis:**



- b. Datenfluss / Informationsfluss: Zum „Schreiben“ eines Datenelementes muss ein Prozessschritt als Vor- und ein Datenelement als Nachbereich gewählt und die Änderungsoperation *Datenelement schreiben* angeklickt werden.



Analog muss zum „Lesen“ eines Datenelementes ein Datenelement als Vor- und ein Prozessschritt als Nachbereich gewählt und die Änderungsoperation *Datenelement lesen* angeklickt werden. Im Beispielprozess wird das Datenelement „Antragssteller“ vom Knoten „Urlaubsantrag ausfüllen“ geschrieben und vom Knoten „Urlaubsantrag prüfen“ gelesen. Als Ergebnis ergibt sich:



Die Pfeilrichtung bei Schreibzugriffen zeigt vom Knoten zum Datenelement, bei Lesezugriffen vom Datenelement zum Knoten.

Lesen und schreiben der restlichen Datenelemente analog:

- i. Prozessschritt „Urlaubsantrag ausfüllen“:

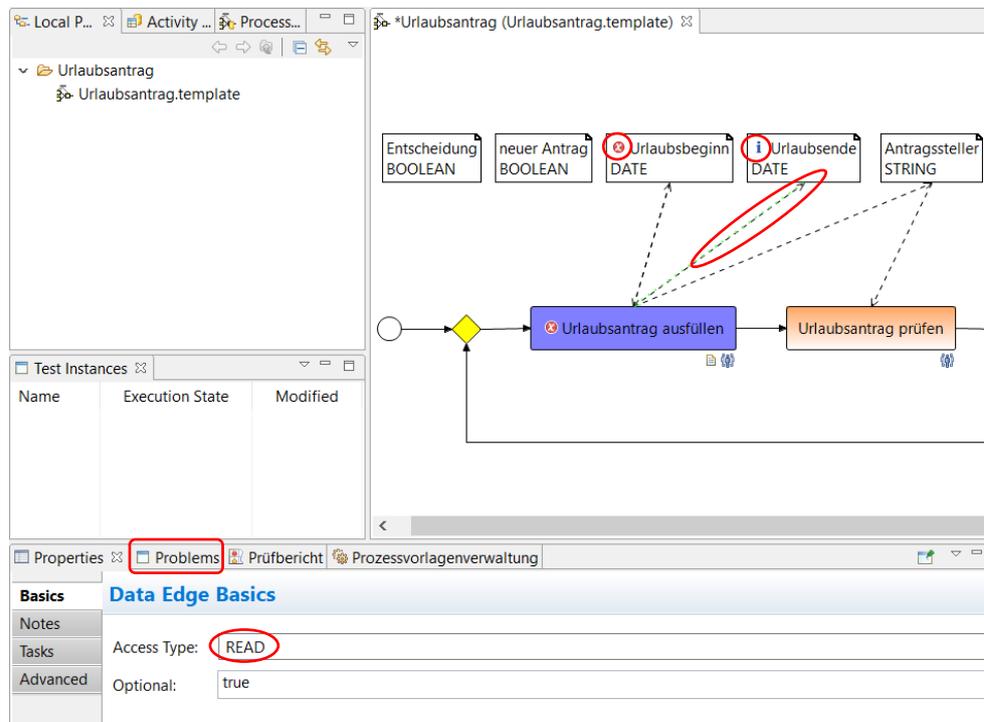
- Schreibzugriffe: „Antragsteller“, „Urlaubsbeginn“, „Urlaubsende“  
Lesezugriffe: „Antragsteller“, „Urlaubsbeginn“, „Urlaubsende“

Das Lesen ist nicht unbedingt erforderlich, erleichtert aber dem Antragsteller in einer erneuten Iteration der Schleife (bei Antragsänderung), seine bisherigen Daten zu korrigieren. Für eine korrekte Modellierung dürfen die Datenelemente nur *optional* gelesen werden, da sie in der ersten Iteration noch nicht geschrieben wurden.

Dazu Doppelklick auf den Datenpfeil (der Pfeil wird grün) → Tab „Basics“  
→ Optional auf „true“ setzen.

Ansonsten wird ein Fehler gemeldet und unter dem Tab *Problems* (links unten) werden Warnungen aufgelistet.

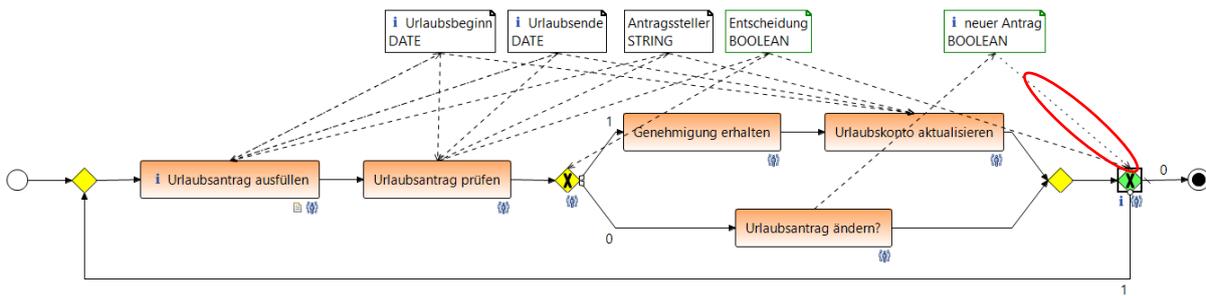
Wenn die Datenelemente optional gelesen werden, verschwinden diese Fehlermeldungen und Informationen über die Lesezugriffe werden unter dem Tab *Prüfbericht* aufgeführt. Als Hinweis auf diese Information, findet man ein Symbol an der linken oberen Ecke des Knotens (siehe Screenshot).



- ii. Prozessschritt „Urlaubsantrag prüfen“:
  - Lesezugriffe: „Antragsteller“, „Urlaubsbeginn“, „Urlaubsende“
  - Schreibzugriff: „Entscheidung“
- iii. Prozessentscheidung (1. XOR Verzweigung) „genehmigt/abgelehnt“:
  - Lesezugriff: „Entscheidung“
- iv. Prozessschritt „Urlaubskonto aktualisieren“:
  - Lesezugriffe: „Antragsteller“, „Urlaubsbeginn“, „Urlaubsende“
- v. Prozessschritt „Urlaubsantrag ändern?“:
  - Schreibzugriff: „neuer Antrag“
- vi. Prozessentscheidung (2. XOR Verzweigung) „Neuer Antrag?“:
 

Lesezugriffe: „Entscheidung“, „neuer Antrag“ („neuer Antrag“ hier nur optional lesen, da dieses Datenelement nur dann einen Wert hat, wenn der untere Zweig mit dem Schritt „Urlaubsantrag ändern“ durchlaufen wurde).

## Ergebnis:



### Tip:

Mit einem Rechtsklick auf den Editor und Wahl von „Display Options“ öffnet sich ein Dialog in dem über *Data Elements Visible* und *Data Edges Visible* die Datenelemente und Datenkante aus- und eingeblendet werden können. Bei genauem Hinsehen im Graph erkennt man den optionalen Datenfluss an den gepunkteten Datenlinien. Dagegen wird der normale Datenfluss durch gestrichelte Linien dargestellt (siehe Screenshot oben). Zur Übersicht über die einzelnen Datenelemente kann man mit der Option *Dependency Visibility Mode* nur den Datenfluss von einem angeklickten Knoten anzeigen lassen.

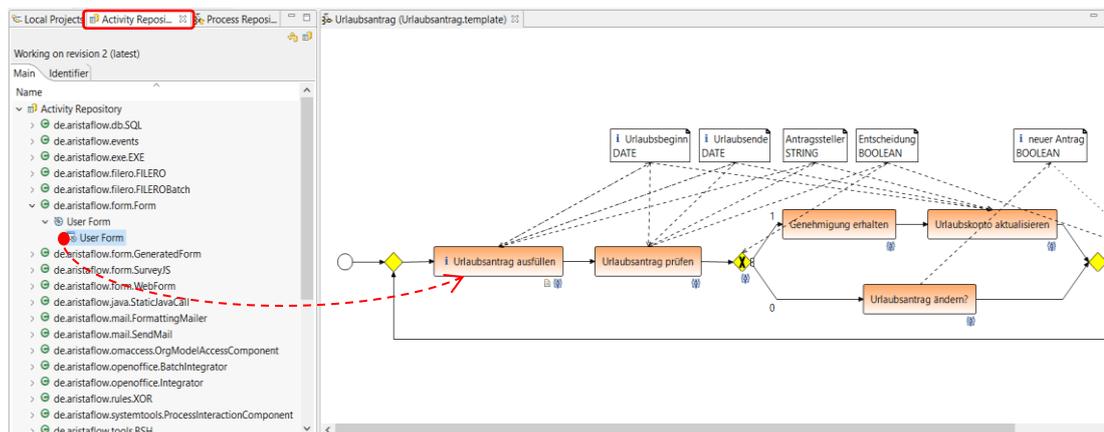
## 5. Zuweisen von Aktivitäten zu Prozessschritten:

Bisher wurde dem Geschäftsprozess nur Bearbeiter und Datenelemente zugewiesen. Dies kann durchaus hilfreich sein, um einen ersten Eindruck des Geschäftsprozesses zu bekommen (analog *Rapid Prototyping*). Für einen echten Produktiveinsatz müssen jedoch Aktivitäten bzw. Services in den Prozess integriert werden. Hierzu dient das *Activity Repository* (Tab, links oben) in dem alle für die Verwendung in Geschäftsprozessen zur Verfügung stehenden Komponenten, Funktionen und Services aufgelistet sind.

Um eine Aktivität zu integrieren, wird diese angewählt und per Drag & Drop aus dem *Activity Repository* heraus auf den entsprechenden Prozessschritt gezogen.

- a. Prozessschritt „Urlaubsantrag ausfüllen“: Hier wird ein User-Formular vom Antragssteller ausgefüllt.

→ de.aristaflow.form.Form aus dem Activity Repository wählen → User Form anklicken, Maustaste gedrückt halten, auf den Prozessschritt ziehen und Maustaste loslassen.



Es öffnet sich automatisch ein Wizard, der Sie durch die weiteren Schritte führt: Im Feld *Node Description* kann für den Prozessschritt eine genauere Beschreibung der auszuführenden Tätigkeit hinterlegt werden. Diese wird zur Ausführungszeit dem Bearbeiter des einzelnen Schrittes angezeigt. Ebenso kann auch hier direkt eine Mitarbeiterzuordnungsregel über *Staff Assignment Rule* angegeben werden.

**Tipp:**

Einige Aktivitäten enthalten bereits Vorschläge für den Knotennamen und dessen Beschreibung. Diese können über Auswahl des Pfeils links neben dem Knotennamen übernommen werden.

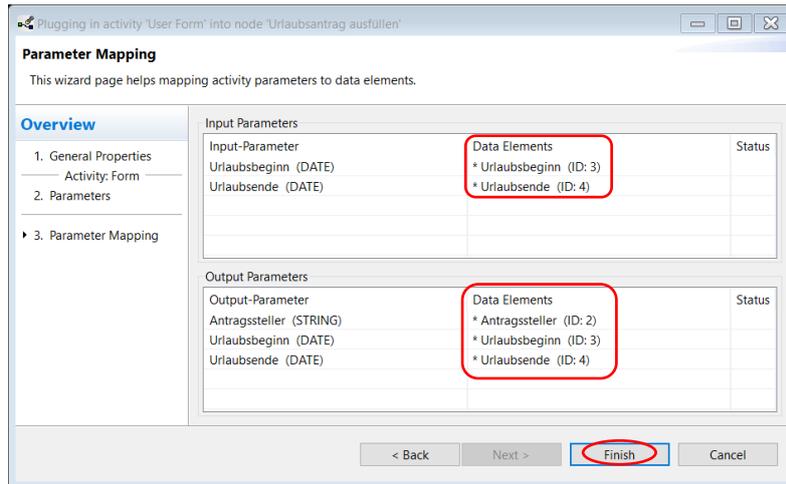
Durch Klicken von *Next* → gelangt man zu der im folgenden Screenshot gezeigten Wizardseite. Hier haben Sie einen Überblick über die zu lesenden („Input Parameters“) und schreibenden („Output Parameters“) Parameter.

Name	Description	Data Type	Optional	Virtual
Urlaubsbeginn		DATE	yes	no
Urlaubsende		DATE	yes	no

Name	Description	Data Type	Optional	Virtual
Antragssteller		STRING	no	no
Urlaubsbeginn		DATE	no	no
Urlaubsende		DATE	no	no

Die nachfolgende Wizardseite dient dem automatischen Verschalten von Aktivitäten mit dem Datenfluss. Es werden automatisch diejenigen Datenelemente (bereits mit dem Knoten verbundene markiert mit „\*“) ausgewählt/angezeigt, die mit den Parametern der Aktivität kompatibel sind. Im hier gezeigten Fall bestehen bereits alle Datenelemente, so dass der Wizard sofort mit *Finish* beendet werden kann.



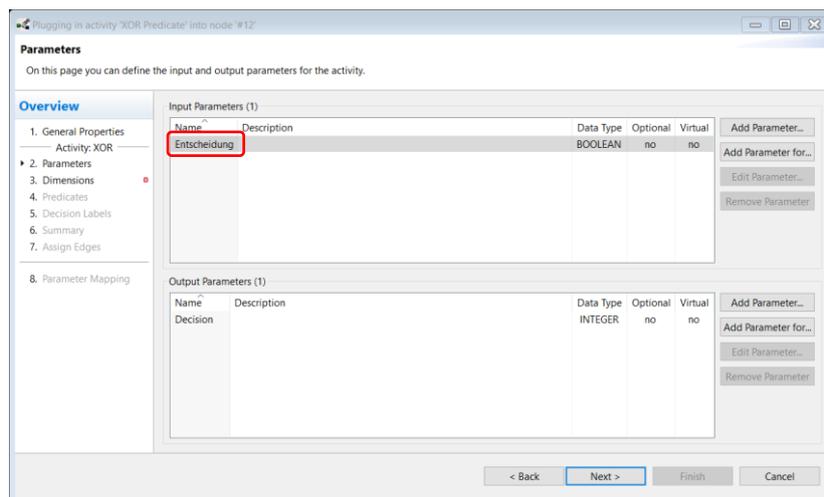
b. Prozessschritt „Urlaubsantrag prüfen“: *de.aristaflow.form.Form* → *User Form* → *User Form* (analog zu Schritt a, den zugewiesenen Bearbeiter *Staff Assignment Rule* → „*NodePerforming–OrgPositonID*“ nicht ändern). In diesem Schritt wird der Parameter „Entscheidung“ geschrieben, also über den Urlaubsantrag entschieden.

c. Prozessentscheidung (1. XOR Verzweigung) „genehmigt /abgelehnt“: Hierbei handelt es sich um eine Verzweigungsaktivität mit der von einfachen Verzweigungsentscheidungen bis hin zu komplexen Geschäftsregeln (Business Rules) jegliche Form von Verzweigungsentscheidungen definiert werden können.

→ *de.aristaflow.rules.XOR* → *XOR Predicate* → *XOR Predicate*.

Bei Genehmigung wir der obere Zweig, bei Ablehnung der untere durchlaufen. Dafür wird der Parameter „Entscheidung“ gelesen und bei *true* auf den oberen Zweig („genehmigt“), bei *false* auf den unteren („abgelehnt“) gemappt.

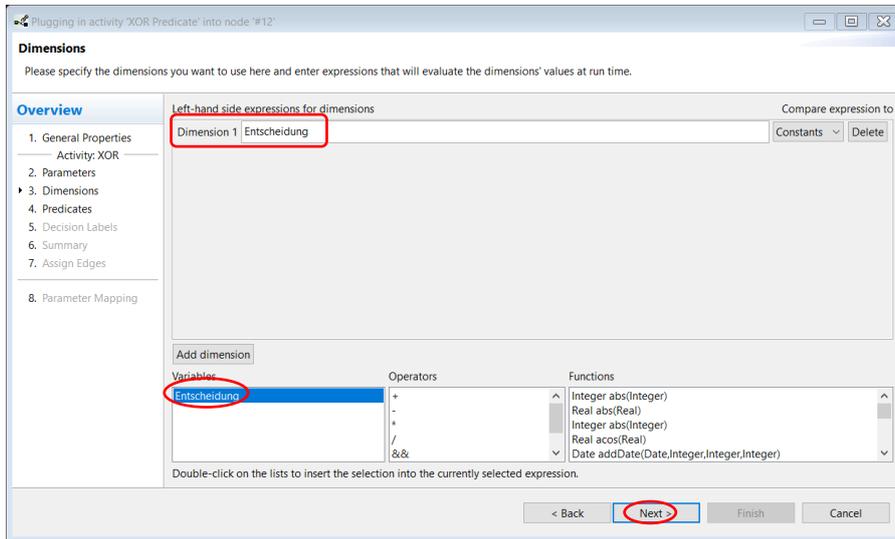
Nach der allgemeinen Einführungsseite gelangt man über zweimaliges Drücken von *Next* → zu der im Folgenden gezeigten Wizardseite.



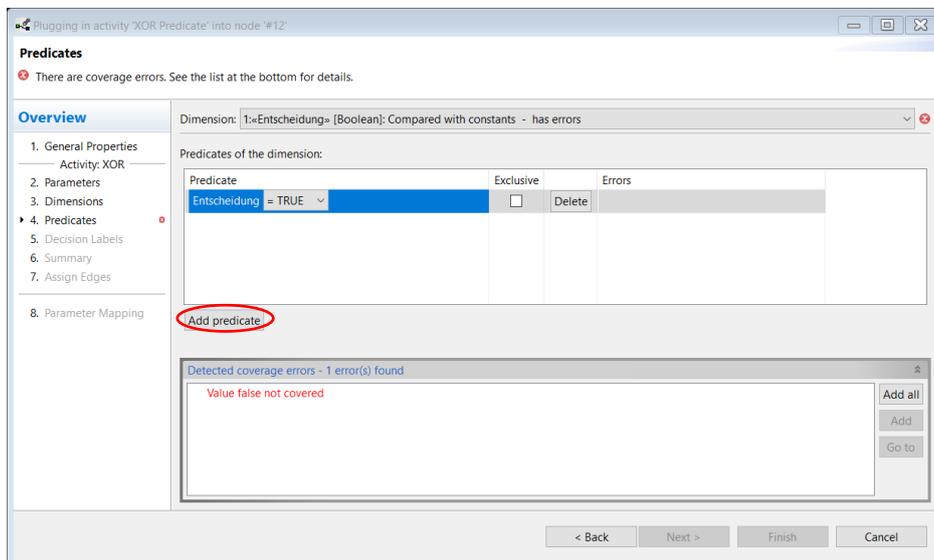
Hier können die Parameter definiert werden, die für eine Verzweigungsentscheidung herangezogen werden können. Da der Prozessschritt bereits eine eingehende Datenkante besitzt, wird hierzu automatisch ein kompatibler Parameter angelegt. Bei dem

Ausgabeparameter „Decision“ handelt es sich um einen Systemparameter, der bei dieser Aktivität immer vorhanden ist. Sie können ihn hier ignorieren.

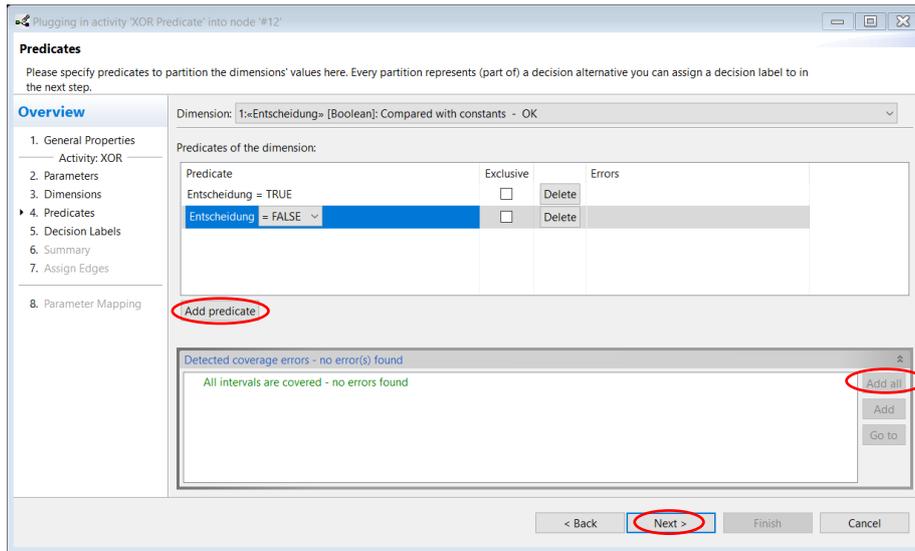
Auf der nächsten Wizardseite werden die Dimensionen der „Entscheidung“ festgelegt. Durch Selektion des Textfeldes neben Dimension 1 und anschließendem Doppelklick auf Variables „Entscheidung“ wird die erste Dimension der Verzweigungsentscheidung angelegt. Da es sich im vorliegenden Beispiel um eine einfache Auswertung eines bool'schen Wertes handelt, genügt die angelegte Dimension bereits.



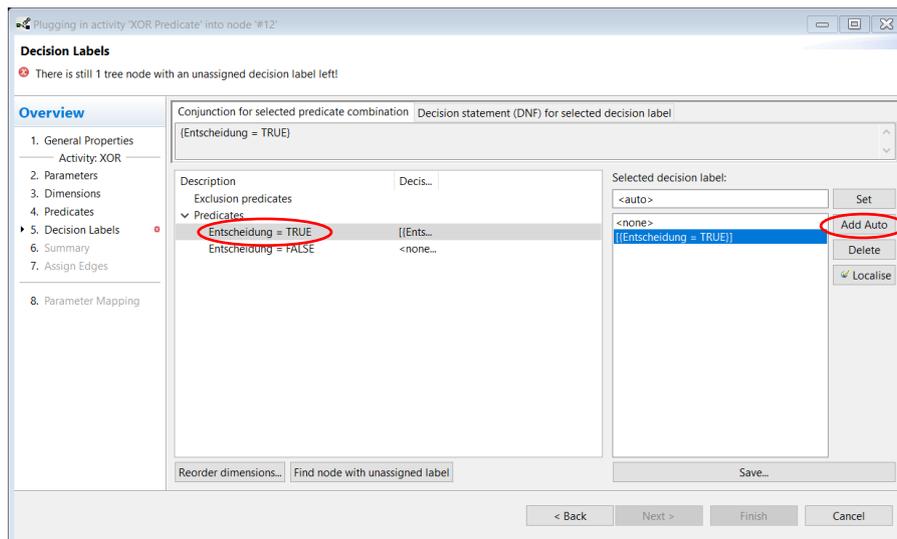
Auf der nächsten Seite werden die Prädikate und die Werte für die angelegte Dimension angegeben.



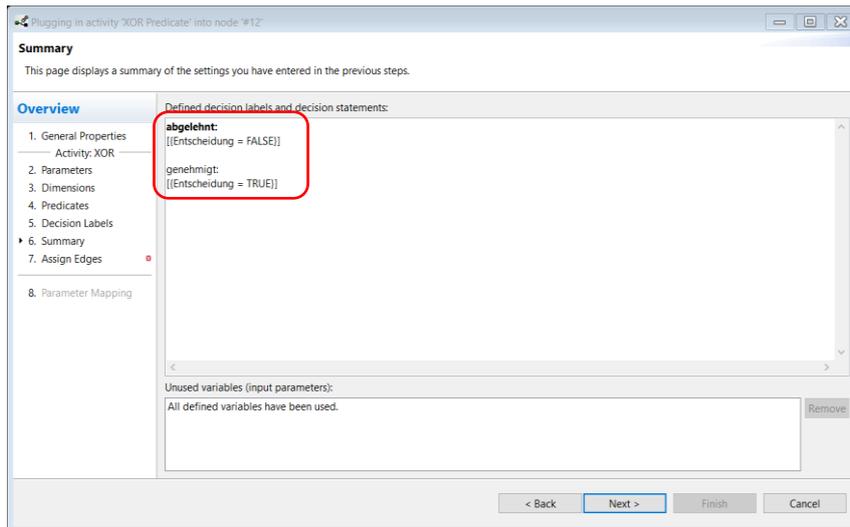
Wie aus dem Screenshot ersichtlich, berechnet der XOR-Wizard automatisch, welche möglichen Werte noch nicht berücksichtigt wurden. Im hier gezeigten Fall wurde bisher nur der Fall „Entscheidung“ = *true* abgedeckt. Um den Wert „Entscheidung“ = *false* anzulegen wird über den Button *Add Predicate* ein neues Prädikat hinzugefügt und der Wert *false* gewählt. Alternativ können die fehlenden Werte auch über den Button *Add all* automatisch hinzugefügt werden. Der Wizard bestätigt sofort, dass alle Werte abgedeckt wurden.



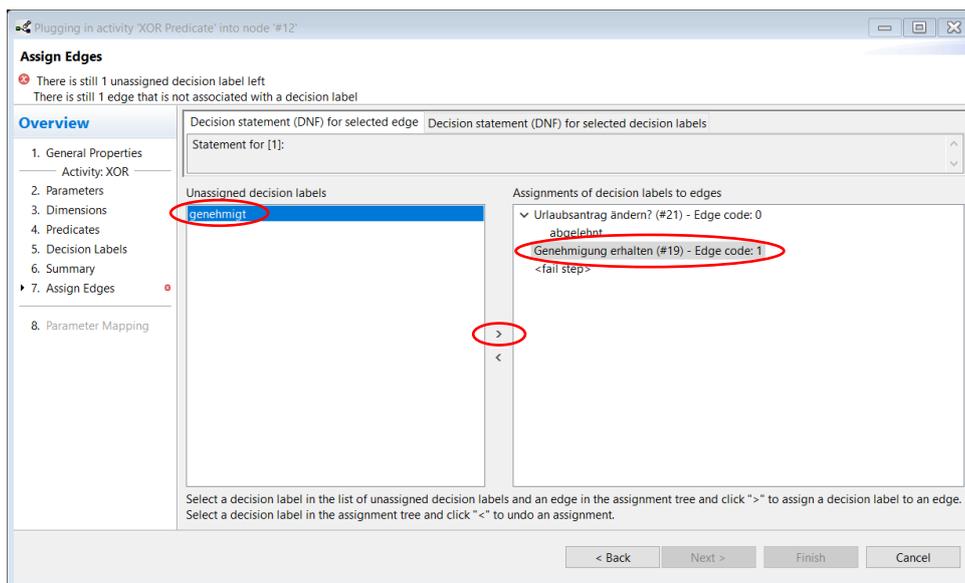
Auf der nächsten Wizard-Seite müssen den Prädikaten Entscheidungs-IDs zugewiesen werden. Hierzu selektiert man „Entscheidung“ = *true* und wählt *Add Auto* (rechts Mitte) um eine automatische ID anzulegen oder man gibt im Textfeld eine eigene ID an und wählt *Set*.

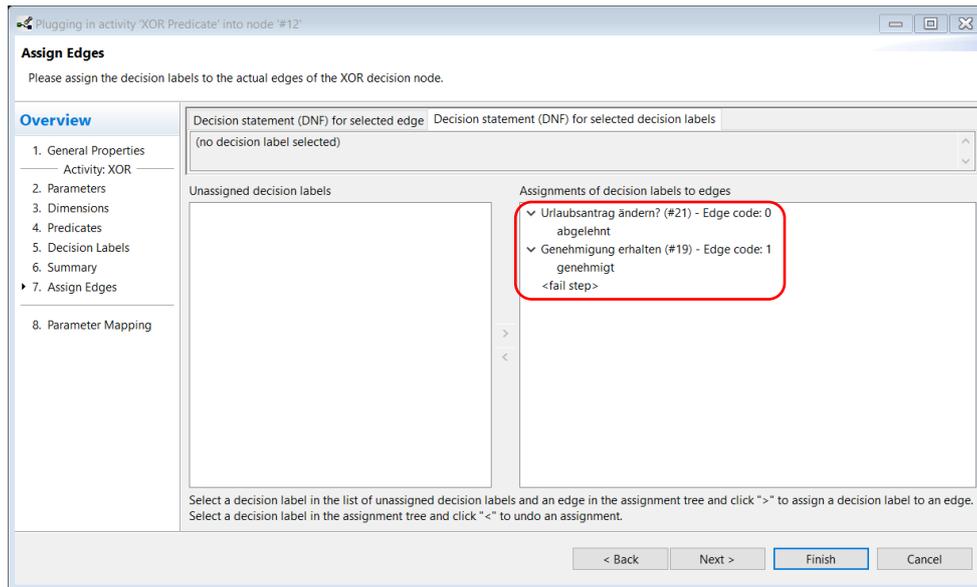


Wir wählen eine eigene ID und nennen diese „genehmigt“. Analog „Entscheidung“ = *false* setzen wir auf ID „abgelehnt“.  
Die folgende Seite zeigt die erstellte Verzweigungsentscheidung nochmals im Überblick.

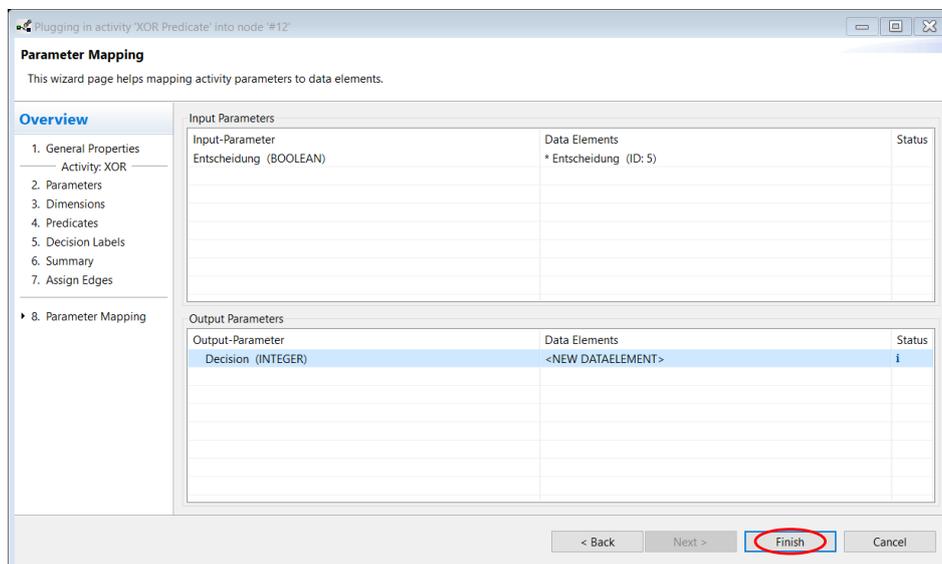


Im nächsten Schritt werden die Verzweigungs-IDs mit den ausgehenden Kanten des XOR-Entscheidungsknotens verknüpft. Hierzu wird die Verzweigungs-ID „genehmigt“ der Kante mit dem Zielknoten „Genehmigung erhalten“ und die ID „abgelehnt“ der Kante mit dem Zielknoten „Antrag ändern?“ zugewiesen.

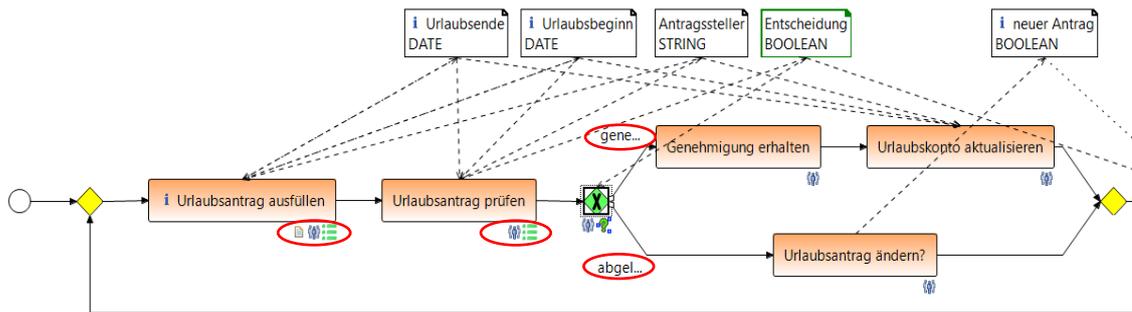




Im letzten Schritt des Wizards werden die angelegten Parameter mit dem über den bereits bekannten Mapping-Dialog mit dem Datenfluss verbunden. Bei dem Output-Parameter *Decision* handelt es sich um einen Systemparameter, der nach Ausführung der Verzweigungsentscheidung den Wert der Entscheidung enthält. Hier wird gespeichert, welche Kante (*Edge 0* oder *Edge 1*) genommen wird. Da diese intern verwendet wird, brauchen wir uns darum nicht kümmern und lassen es auf `<NEW DATAELEMENT>`, also vom System neu generieren.



Über *Finish* wird der Wizard beendet. An den ausgehenden Kontrollflusskanten des Verzweigungsknotens ist nun zu erkennen, bei welchem Entscheidungswert welcher Zweig genommen wird. Außerdem erkennt man aus den blauen und grünen Symbolen rechts unter den Knoten, dass Agents bzw. Aktivitäten und bei „Urlaubsantrag ausfüllen“ auch schon eine Beschreibung hinterlegt wurde.

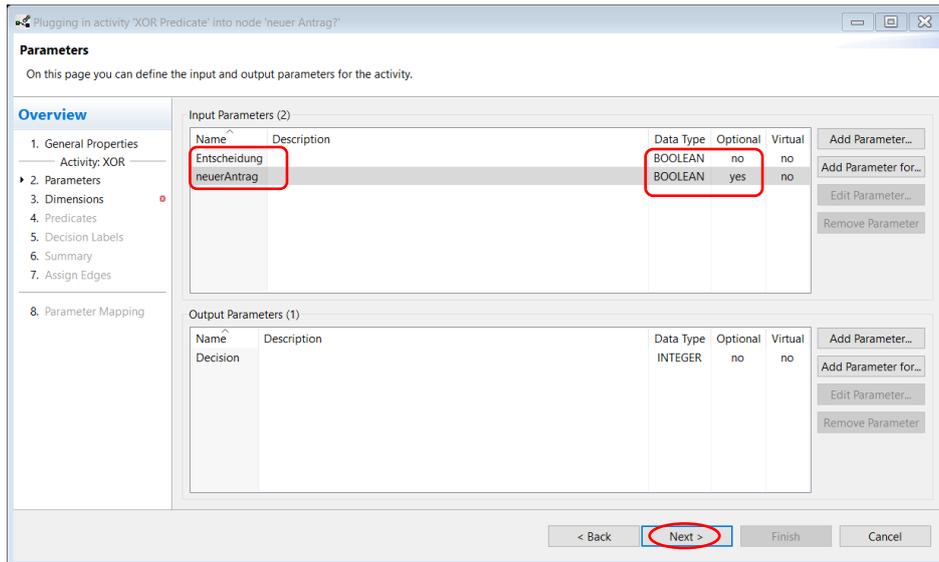


- d. Prozessschritt „Genehmigung erhalten“:  
 de.aristaflow.form.Form → User Form → User Form (analog zu Schritt a, hier als Bearbeiter „Instancelniator-AgentID“ lassen). Bei Genehmigung wird der Antragssteller „Meier“ hier über seinen genehmigten Urlaub informiert.
- e. Prozessschritt „Urlaubskonto aktualisieren“:  
 de.aristaflow.form.Form → User Form → User Form (analog zu Schritt a, den zugewiesenen Bearbeiter Sachbearbeiter „Schmidt“ nicht ändern). Der Sachbearbeiter aktualisiert das Urlaubskonto. Dazu liest er die Daten „Antragssteller“, „Urlaubsbeginn“ und „Urlaubsende“.
- f. Prozessschritt „Urlaubsantrag ändern?“:  
 de.aristaflow.form.Form → User Form → User Form (analog zu Schritt a, als Bearbeiter „Instancelniator-AgentID“ lassen). Bei Ablehnung kann der Antragssteller „Meier“ die Urlaubsdaten ändern und einen neuen Urlaub beantragen. In diesem Schritt entscheidet er, ob er einen neuen Antrag stellen möchte oder nicht. Hier wird der Parameter „neuerAntrag“ geschrieben.
- g. Prozessentscheidung (2. XOR Verzweigung) „Neuer Antrag?“:  
 de.aristaflow.rules.XOR → XOR Predicate → XOR Predicate

**Hinweis:**

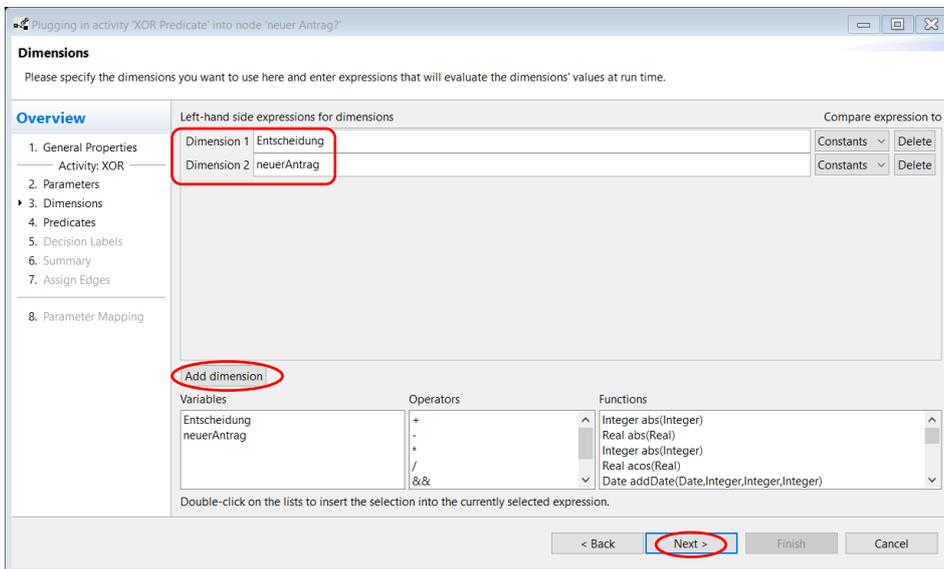
Das zu lesende Datenelement „neuer Antrag“ wird im XOR-Knoten auf „neuerAntrag“ umbenannt, da Leerzeichen in Variablennamen (Parametern) nicht zugelassen sind.

Hier haben wir zwei Parameter; „Entscheidung“ und „neuerAntrag“. Diese werden gebraucht, um zu entscheiden, ob der Prozess beendet oder der Graph noch mal durchlaufen wird.

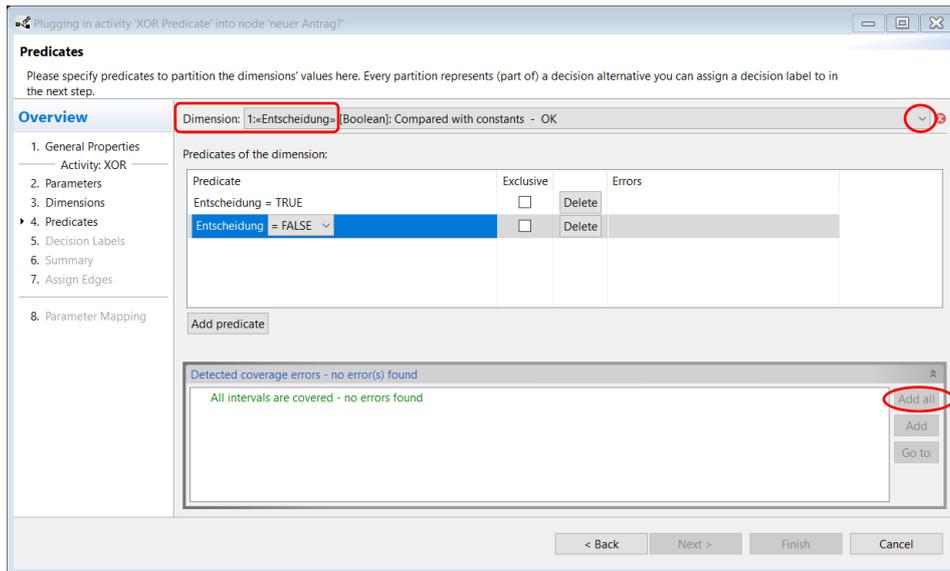


Wenn die „Entscheidung“ = *true* ist, wird der Parameter „neuerAntrag“ nicht mehr benötigt und der Prozess wird automatisch beendet. Diesen Fall mappen wir auf den „Nein“ Zweig, also „neuer Antrag?“ -> *Nein*. Nur der Fall „Entscheidung“ = *false* und „neuer Antrag“ = *true* wird auf den Zweig „ja“ gemappt, d.h. „neuer Antrag?“ -> *Ja*.

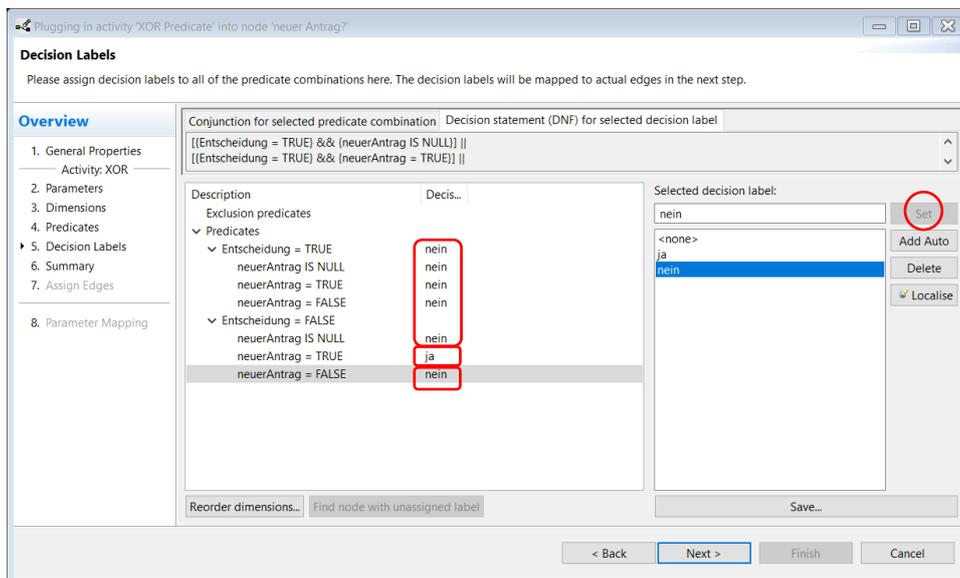
Wir nehmen „Entscheidung“ als erste Dimension: -> Doppelklick auf „Entscheidung“, „neuerAntrag“ als zweite Dimension: -> Klick auf *Add dimension*. -> Doppelklick auf „neuerAntrag“ -> *Next*.



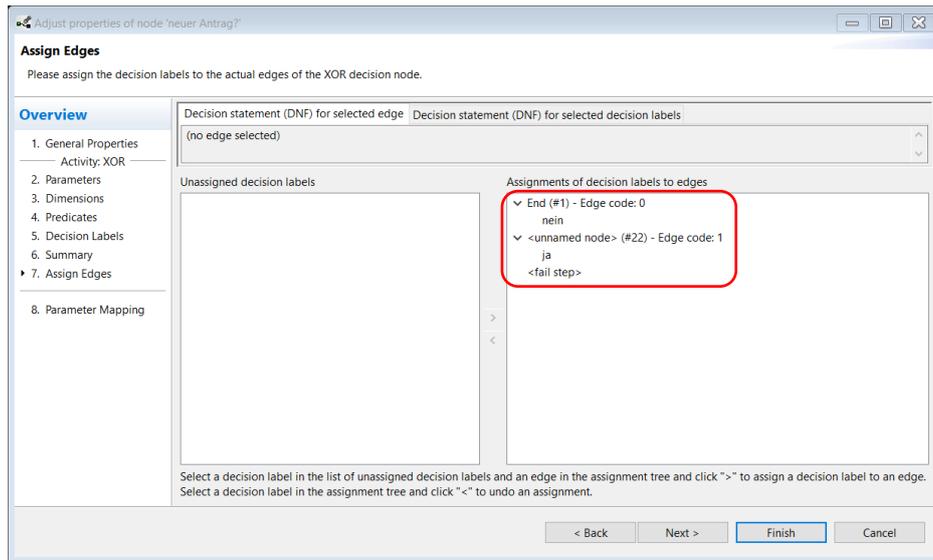
Nun alle möglichen Prädikate beider Dimensionen anlegen. 1. Dimension wird schon angezeigt. Wir fügen automatisch die noch fehlenden Prädikate hinzu -> Klick auf *Add all*  
Wählen der zweiten Dimension aus Combo Box rechts oben -> Klick auf *Add all*.



Im nächsten Schritt legen wir die Entscheidungs-IDs fest. Wir sehen alle möglichen Kombinationen von Prädikaten. Nur der Fall „Entscheidung“= *false* mit „neuerAntrag“ = *true* wird auf Entscheidungs-ID „ja“ abgebildet. Alle anderen Fälle werden auf die Entscheidungs-ID „nein“ gesetzt.

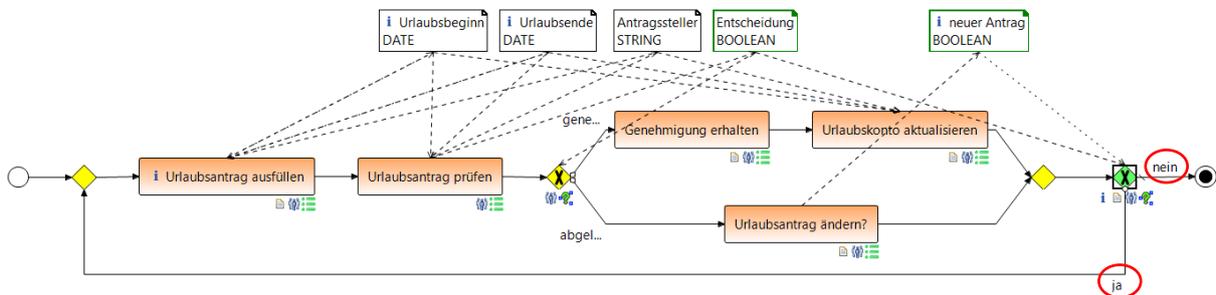


Die nächste Wizardseite dient nur zur Übersicht. Zum Schluss werden die Entscheidungs-IDs mit den ausgehenden Kanten verknüpft. Dazu Entscheidungs-ID „ja“ mit der Kante Edge code: 1 verbinden, Entscheidungs-ID „nein“ mit der Kante Edge code: 0, die zum Ende führt.



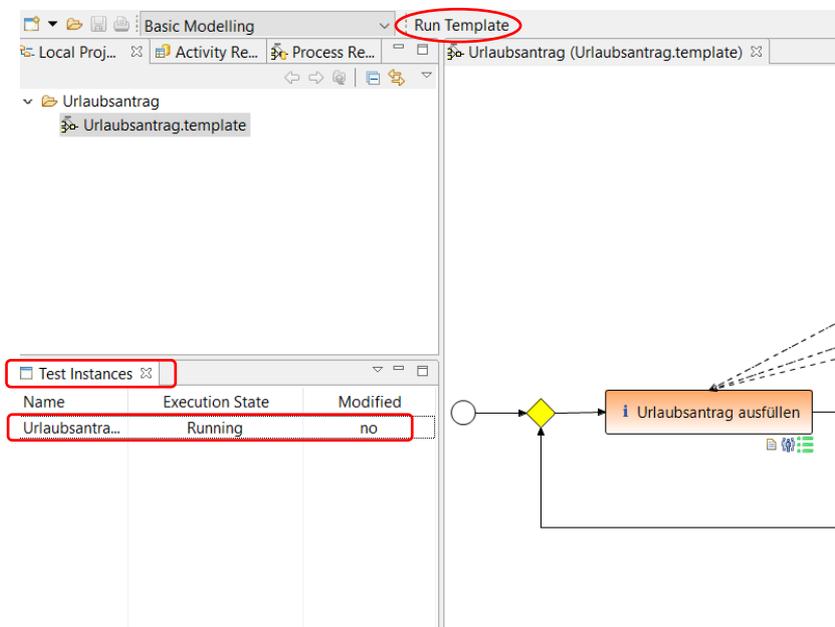
Mit dem Zuweisen der Aktivitäten zu den Knoten ist die Prozessmodellierung abgeschlossen. Im nächsten Schritt folgt das Deployment des Prozesses auf den Server.

### Ergebnis der gesamten Prozessmodellierung:

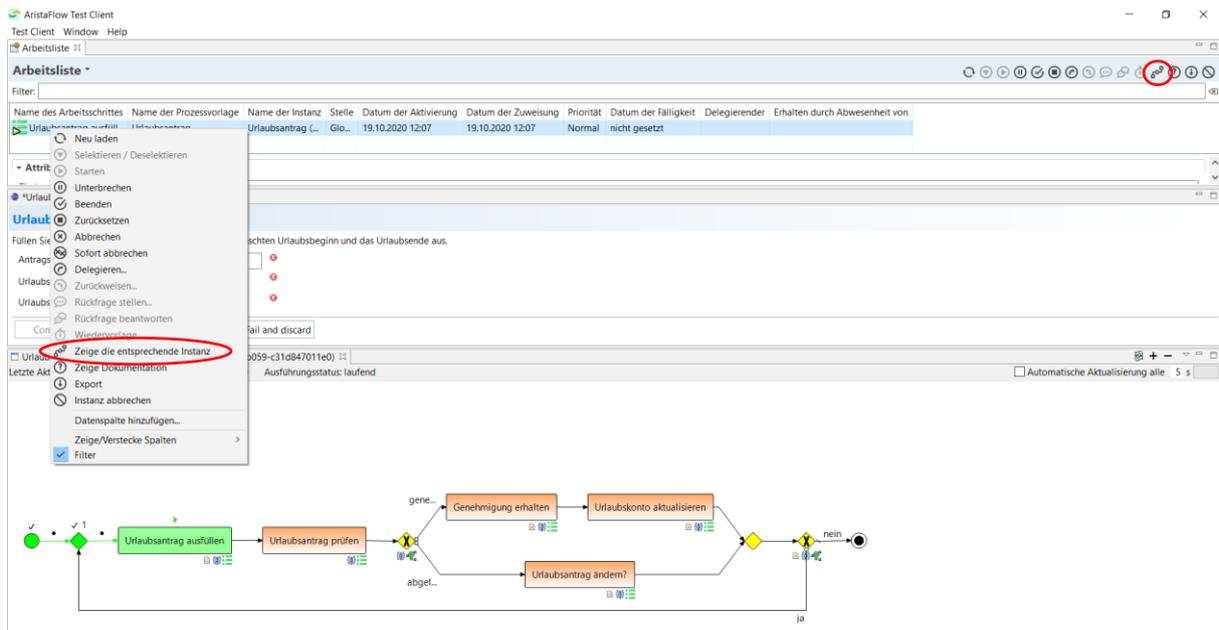


Strukturelle Fehler wie Deadlocks oder fehlende Parameter werden bereits bei der Modellierung automatisch erkannt bzw. können gar nicht erst modelliert werden. Semantische Fehler im Sinne von Abweichungen vom real existierenden Prozess lassen sich prüfen, indem der Prozess mit allen Beteiligten „durchgespielt“ wird. Bevor man den Prozess von allen Agents (hier „Meier“, „Schulz“ und „Schmidt“) einzeln ausführen lässt, kann man ihn als Modellierer (*supervisor*) schrittweise im Test Client durchführen. In diesem Client wird jeder Schritt vom *Agent(userName='supervisor')* gestartet und erledigt. Damit hat man einen Überblick, ob das Template ohne Probleme einsetzbar ist.

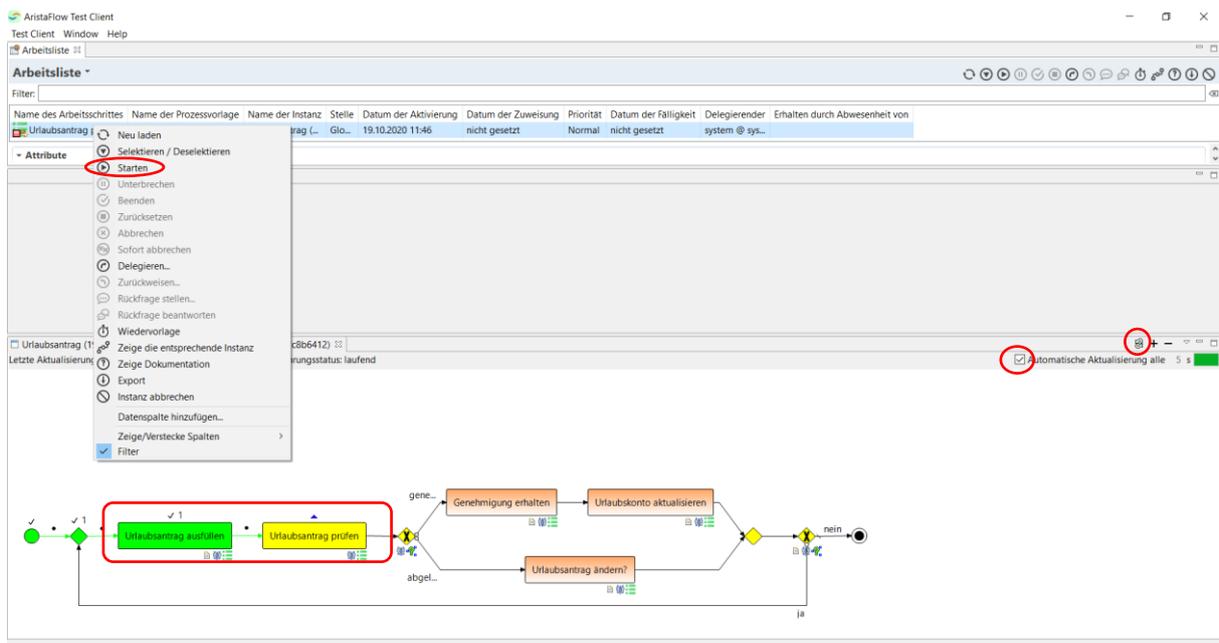
Mit einem Klick auf den Tab *Run Template* erstellen Sie eine *Test-Instance* im Process Template Editor und können das eben erstellte Template im Test-Client testen. Ein Test-Client Fenster mit laufender Instanz öffnet sich. Die erstellte Instanz wird im Process Template Editor unter *Test Instances* (links Mitte) angezeigt.



Der Test-Client ist sehr praktisch für den Modellierer, da man hier das Template ausführen kann ohne es auf den Server hochladen zu müssen. Im Test-Client wird zuerst der Prozessschritt gestartet, dann die auszuführende Tätigkeit erledigt und dabei der Ausführungsstatus des Graphen visualisiert. Das alles erfolgt im gleichen Fenster. Für die Instanzanzeige im Graphen muss man im Rechtsklick Menü des einzelnen Arbeitsschrittes (hier „Urlaubsantrag ausfüllen“ links oben) *Zeige die entsprechende Instanz* wählen oder auf das Graphsymbol rechts oben klicken (siehe Screenshot).



Wir führen die Aktivität aus, indem wir den „Antragssteller“ im Formular eingeben und Daten für Urlaubsbeginn und -ende auswählen. Mit Klick auf *Confirm* wird die Aktivität beendet und in den nächsten Schritt geschaltet. Um den nächsten Arbeitsschritt zu starten, muss man hier „Urlaubsantrag prüfen“ selektieren und im Rechtsklickmenü *Starten* wählen oder die Aufgabe per Doppelklick starten. In diesem Schritt entscheiden wir darüber, ob der Urlaub bewilligt oder abgelehnt wird (*yes/no* wählen). Durch Klick auf *Confirm* wird der Schritt beendet. Der erste Schritt „Urlaubsantrag ausfüllen“ musste wegen Autostart nicht extra gestartet werden. Den Ausführungsstatus des Graphen kann man alle 5 Sekunden entweder automatisch oder selbst mit dem Refresh-Symbol (rechts Mitte) aktualisieren. Die aktuellen und durchgelaufenen Knoten werden grün, der zu startende Knoten gelb angezeigt.





### Hinweis:

Im Test Client müssen auch die automatischen Schritte wie XOR-Verzweigungsaktivitäten explizit gestartet werden

Nach der XOR-Verzweigungsaktivität wird entsprechend der Entscheidung entweder der obere oder untere Zweig genommen.

Bei Genehmigung werden im oberen Zweig die Schritte „Genehmigung erhalten“ und „Urlaubskonto aktualisieren“ ausgeführt. Diese werden zuerst gestartet, dann bestätigt und verlassen. Somit endet der Prozess.

Bei Ablehnung wird im unteren Zweig im Schritt „Urlaubsantrag ändern?“ entschieden, ob ein neuer Antrag gestellt wird.

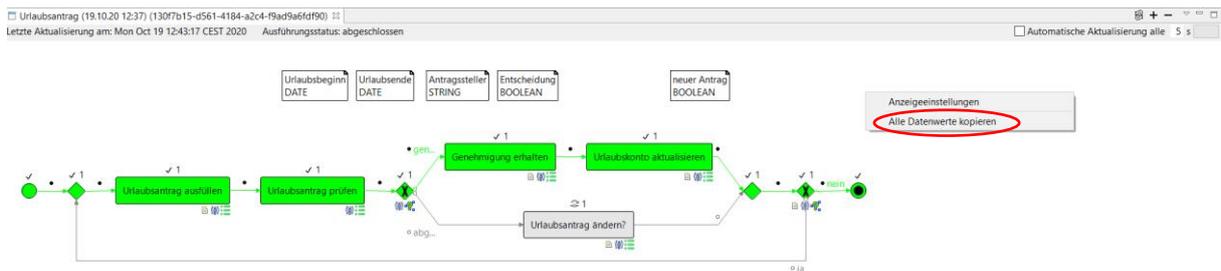
Entscheidet sich der Antragsteller den Urlaub zu ändern kehrt der Graph zum Anfang zurück und er muss einen neuen Urlaubsantrag ausfüllen.

Wenn kein neuer Antrag gestellt wird, dann erreicht der Graph den Endpunkt.



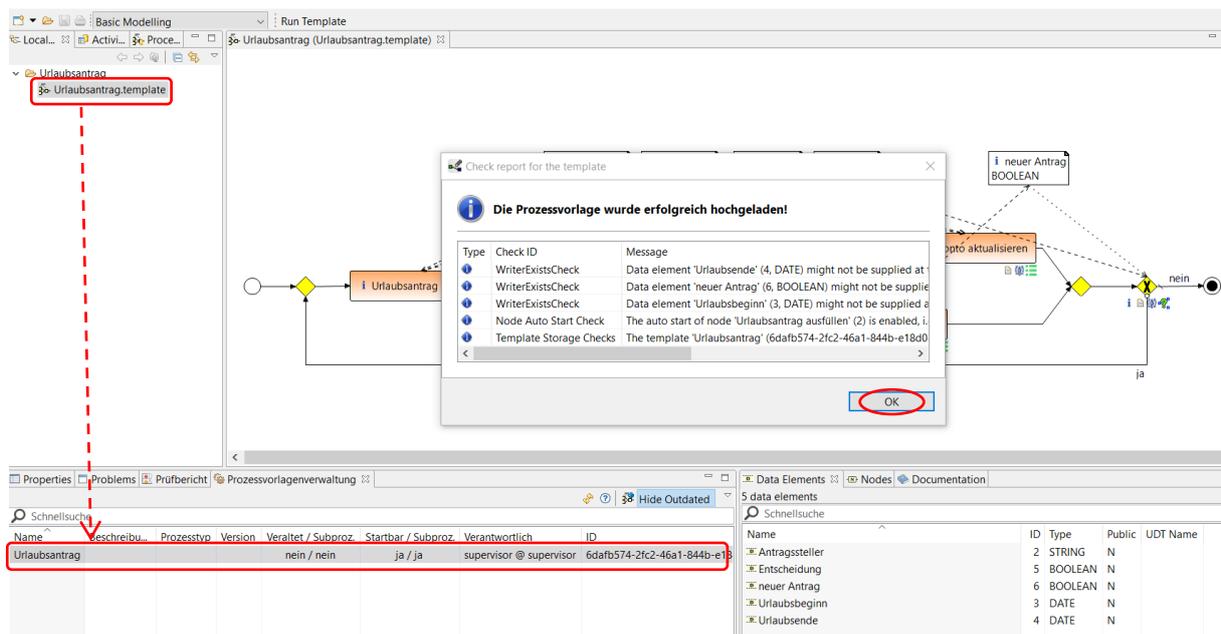
### Tip:

Zugriff auf die schon eingegebenen Daten ist durch Kopieren der Datenwerte im Graphen möglich. Beispielsweise durch Rechtsklick im Ausführungsstatus über *Alle Datenwerte kopieren* und die Daten in einen externen Texteditor exportieren (siehe Screenshot). Einzelne Datenwerte lassen sich anzeigen, indem man das entsprechende Datenelement anklickt. Nach Beendigung oder Abbruch der Instanz ist kein Zugriff auf die Instanzdaten im Test Client mehr möglich.



## Deployment des Prozesses

Vor der Ausführung des Prozesses im *Rich-* oder *WebClient* muss das Template auf den ARISTAFLOW-Server geladen und dabei auf eventuelle Fehler geprüft werden. Für das Deployment des „Urlaubsantrag“-Prozesses wird dieser mit der linken Maustaste im *Navigator* selektiert, unter gedrückt halten der Maustaste auf die View *Prozessvorlagenverwaltung* (links unten) gezogen und abgelegt (Drag & Drop). Der Prozess durchläuft automatisch einen Vollständigkeits- und Korrektheitscheck. Wenn alles in Ordnung ist, wird der Urlaubsantragsprozess auf den ARISTAFLOW-Server geladen und kann nun standardmäßig vom Prozessmodellierer ausgeführt (instanziiert) werden.



The screenshot shows the ARISTAFLOW process editor interface. A dialog box titled "Check report for the template" is open, displaying a successful upload message: "Die Prozessvorlage wurde erfolgreich hochgeladen!". Below the message, a table lists the checks performed:

Type	Check ID	Message
WriterExistsCheck		Data element 'Urlaubsende' (4, DATE) might not be supplied at
WriterExistsCheck		Data element 'neuer Antrag' (6, BOOLEAN) might not be supplied
WriterExistsCheck		Data element 'Urlaubsbeginn' (3, DATE) might not be supplied a
Node Auto Start Check		The auto start of node 'Urlaubsantrag ausfüllen' (2) is enabled, i
Template Storage Checks		The template 'Urlaubsantrag' (6dafb574-2fc2-46a1-844b-e18d0

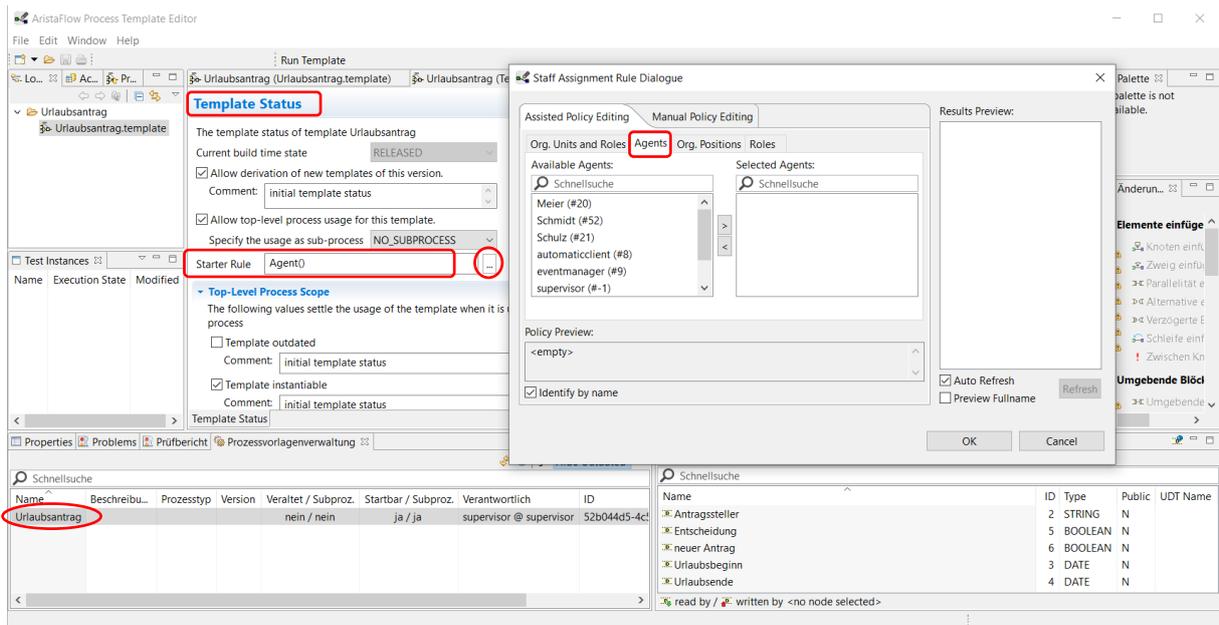
The background shows a process diagram with nodes like "Urlaubsantrag" and "neuer Antrag". Below the diagram, a table lists process templates:

Name	Beschreibung	Prozesstyp	Version	Veraltet / Subproz.	Startbar / Subproz.	Verantwortlich	ID
Urlaubsantrag				nein / nein	ja / ja	supervisor @ supervisor	6dafb574-2fc2-46a1-844b-e18d0

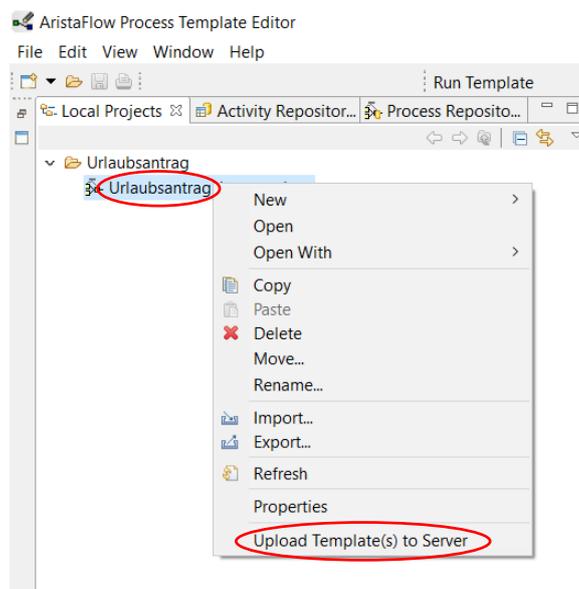
Below the table, a "Schnellsuche" (quick search) section lists data elements:

Name	ID	Type	Public	UDT Name
Antragssteller	2	STRING	N	
Entscheidung	5	BOOLEAN	N	
neuer Antrag	6	BOOLEAN	N	
Urlaubsbeginn	3	DATE	N	
Urlaubsende	4	DATE	N	

Sollte der Prozess hingegen noch nicht vollständig sein oder noch Fehler enthalten, werden diese entsprechend aufgelistet und der Prozess wird nicht auf den Server übertragen. Da es in den seltensten Fällen gewünscht ist, dass ein Prozesstemplate nur vom Modellierer gestartet werden kann, bietet der ARISTAFLOW-Process Template Editor die Möglichkeit für jede Prozessvorlage individuell festzulegen wer diese starten darf. Hierfür wird der entsprechende Prozess in der View *Prozessvorlagenverwaltung* selektiert und im Rechtsklick-Menü *Open Template Status* ausgewählt. Unter dem Punkt *Starter Rule* können nun die Personen, Rollen oder Organisationseinheiten festgelegt werden, die berechtigt sein sollen den Prozess „Urlaubsantrag“ zu starten. Zum Testen und Vorführen wird empfohlen, hier *Agent()* also alle Bearbeiter einzutragen, so dass der Prozess von allen Mitarbeitern gestartet (instanziiert) werden kann. Später wenn Sie einen bestimmten Agent als Starter festlegen möchten, können Sie auf das Kästchen neben *Starter Rule* klicken und den gewünschten Starter aus den Agents des Org.Model Editors wählen (siehe Screenshot).



Wenn Sie Änderungen am Template vornehmen, müssen Sie das aktuelle Template vor der Ausführung nochmal auf den Server hochladen. Das ist über Rechtsklick auf das Template und Wahl *Upload Template(s) to Server* möglich.



Wenn Sie ein Template mehrmals auf den Server laden, dann befinden sich mehrere Versionen eines Templates auf dem Server und der Client kann jedes Template starten und mehrere gleichzeitig ausführen. So verlieren Sie beim Testen schnell den Überblick, welches Template das aktuellere ist. Daher können Sie das ältere Template im Server „outdaten“ und damit in einen nicht startbaren (nicht initiierten) Zustand versetzen. Dann kann das Template vom Anwender nicht mehr gestartet werden. Dies erreichen Sie durch Rechtsklick auf das Template und Wahl -> *Outdate Template Status*. Durch Klick auf *Hide Outdated* in der Prozessvorlagenverwaltung können Sie dann die nicht mehr startbaren Templates ausblenden.

Name	Beschreibu...	Prozesstyp	Version	Veraltet / Subproz.	Startbar / Subproz.	Verantwortlich	ID
Urlaubsantrag				nein / nein	ja / ja	supervisor	3a52c96-7a37-4f10-0443-265bc491
UrlaubsantragNeu				nein / nein	ja / ja	supervisor	-b4d2cc1

- View Template (read only)
- Outdate Template Status**
- Open Template Status
- Verantwortlich kopieren
- Filter >
- Spalten >

Um den Templatenamen zu ändern, klicken Sie im Editor auf die weiße Fläche und gehen Sie im *Properties View* (links unten) auf *Basics* Tab. Hier geben Sie den neuen Namen für das Template ein. Der *Filename* ist nicht gleichzeitig der Templatename, so dass Änderungen des Dateinamens nicht auf den Templatenamen übertragen werden.

Properties	Problems	Prüfbericht	Prozessvorlagenverwaltung
<b>Basics</b>			
Localisation	Name: <input type="text" value="UrlaubsantragNeu"/>		
Notes	Instance Name Template: <input type="text"/>		
Tasks	Description: <input type="text"/>		
Error Handling	Template Supervisor: <input type="text" value="Agent[id='-1', userName='supervisor', orgPositionID='-1', orgPositionName='supervisor']"/> <input type="button" value="Change..."/>		
Advanced			

Nach erfolgreichem Durchlauf des Prozessgraphen im Test Client und Hochladen auf den Server können wir den Urlaubsantrag nun im Rich Client oder im WebClient ausführen. Nun melden sich alle Beteiligten (Agents) des Prozesses auf dem Client an und bearbeiten die Schritte in ihrer Arbeitsliste.

Da im Prozess automatische Prozessschritte (z.B. die XOR-Verzweigungsaktivitäten) vorhanden sind, muss zwingend der *ARISTAFLOW-AutomaticClient* gestartet werden!

Das geht entweder über das Startmenü

**AristaFlow-BPM Suite -> automaticclient**

oder alternativ im Installationsverzeichnis unter folgendem Pfad

**...\AristaFlow-Chronos\automaticclient\bin\automaticclient.bat**

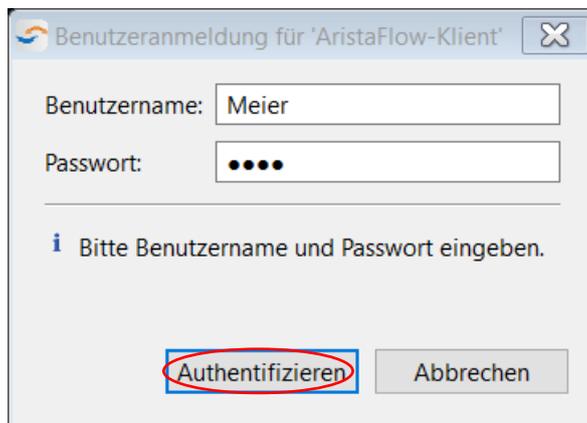
Dann starten Sie für jeden Beteiligten den ARISTAFLOW Client über das Startmenü

**AristaFlow BPM Suite -> AristaFlow Client**

oder alternativ im Installationsverzeichnis über den Pfad

**...\AristaFlow-Chronos\Client\Client.exe**

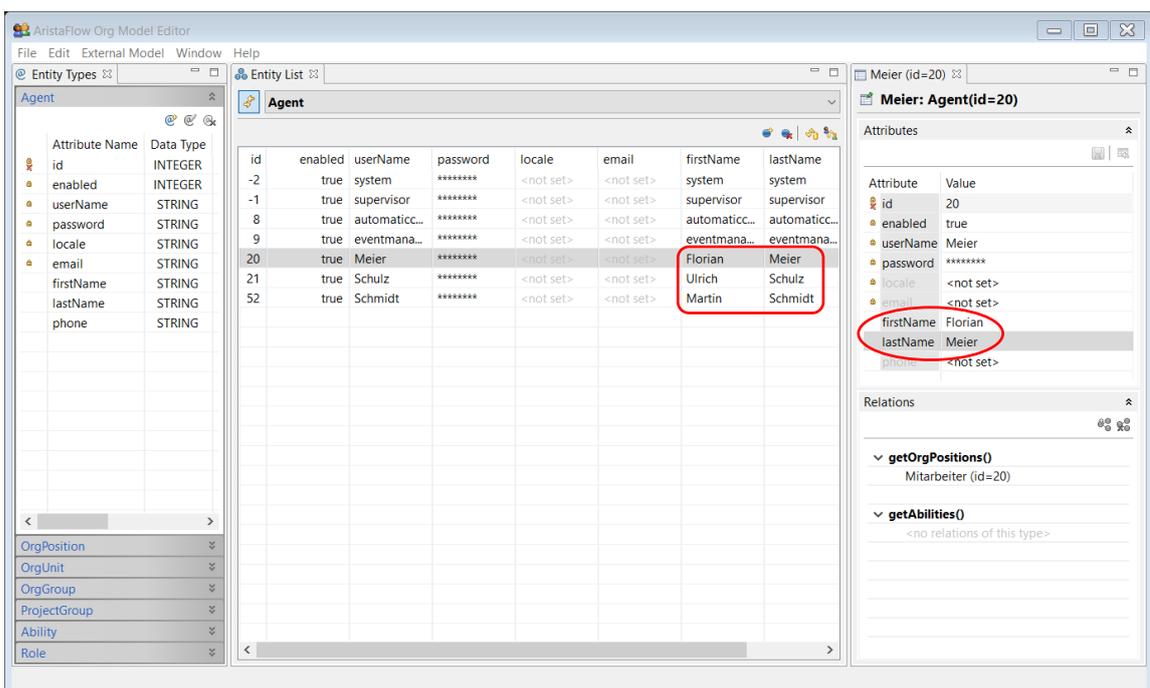
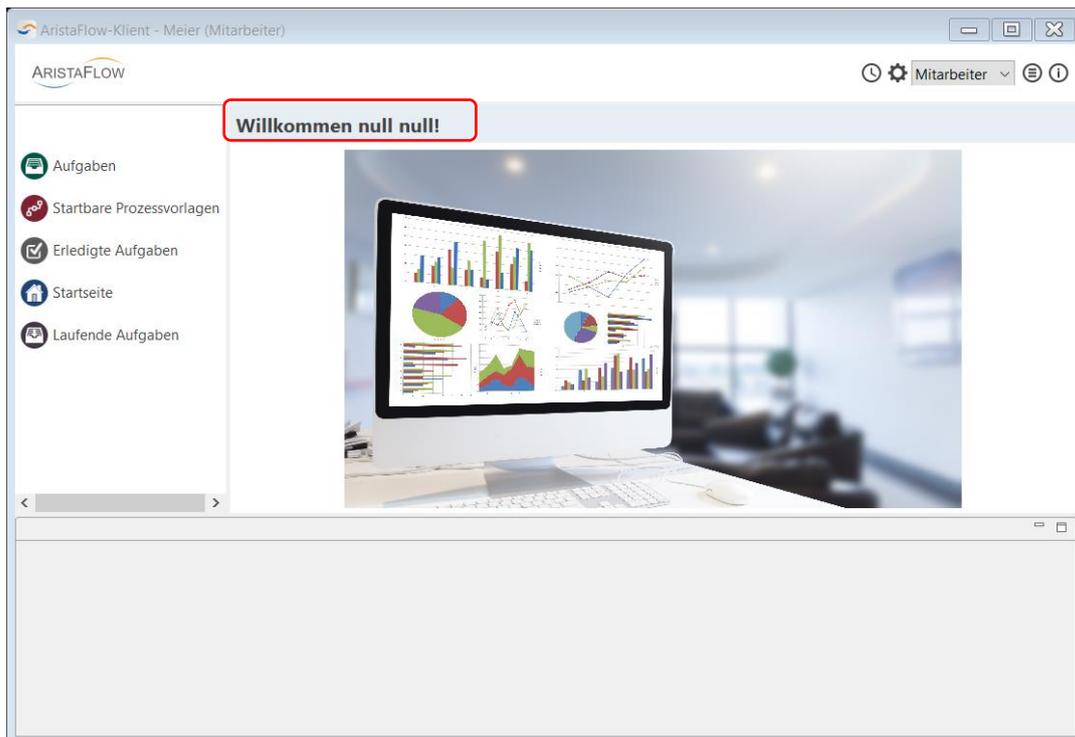
1. Client 1: Anmelden mit User *Name* „Meier“ und dem Passwort, das diesem User im *OrgModelEditor* gegeben wurde (z.B. „test“).



Nach dem Authentifizieren erscheint der *ARISTAFLOW-RichClient*. Dies ist die Arbeitsoberfläche, mit der der Endbenutzer Prozesse starten und Aufgaben abarbeiten kann.

### **Tipp:**

In der Client-Benutzeroberfläche wird man mit Vor- und Nachnamen begrüßt. Da bei unseren Bearbeitern keine Namen hinterlegt wurden, kann bei der Begrüßung nichts angezeigt werden. Daher erscheint die Meldung „Willkommen null null“. Um das zu vermeiden, können Sie im *OrgModelEditor* Vor- und Nachnamen für Agents eingeben (siehe folgenden Screenshots).



2. Client 2: Anmelden mit *User Name* „Schulz“ und dem Passwort das diesem User im *OrgModelEditor* gegeben wurde („test“).
3. Client 3: Anmelden mit *User Name* „Schmidt“ und dem Passwort das diesem User im *OrgModelEditor* gegeben wurde („test“).

Optional:

4. Client 4: Anmelden mit *User Name* „automaticclient“ und dem Passwort „password“. Dies startet die graphische Oberfläche zur Überwachung des automatischen Klienten

5. Client 5: Anmelden mit User *Name* „supervisor“ und dem Passwort „password“. Der Supervisor bekommt alle Prozessschritte in seine Arbeitsliste und kann somit für Überwachungs- oder Demozwecke verwendet werden.

Der Test der Benutzeroberfläche findet in folgendem Teil im WebClient statt. Sie können jedoch die gleichen Schritte auch im RichClient ausführen.

Alternativ zum *Rich Client* gibt es den *ARISTAFLOW-WebClient*. Dieser bietet eine identische Funktionalität und das gleiche *Look-and-Feel* wie der *Rich Client*.

Um den *WebClient* zu starten, müssen Sie zusätzlich zum AristaFlow-Server den *WebServer* starten. Dies können Sie wie gewohnt über das Startmenü

**AristaFlow BPM Suite -> Web server with console**

oder alternativ im Installationsverzeichnis unter dem Pfad

**...\AristaFlow-Chronos\WebServer\bin\webclient.bat** tun.

Um den Graphverlauf anzeigen zu können, brauchen Sie zusätzlich den Server *Process image renderer with console*. Starten Sie diesen über das Startmenü

**AristaFlow BPM Suite -> Process image renderer with console**

oder alternativ über die Batchdatei im Installationsverzeichnis unter dem Pfad

**...\AristaFlow-Chronos\ProcessImageRenderer\bin\processimagerenderer.bat**

## Hinweis:

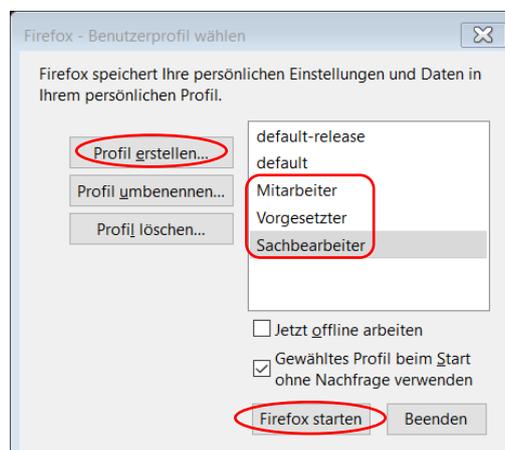
Zum Testen des Urlaubsantrags ist es sinnvoll mehrere Benutzer gleichzeitig im WebClient anzumelden. Da sich aber mit einem Benutzerprofil eines Internet Browsers nur eine Person am Client anmelden kann, braucht man entweder mehrere Browser oder mehrere Profile in einem Browser. Sie können für die drei Beteiligten des Prozesses drei Browser wie z.B. Firefox, Opera und Internet Explorer nehmen.

Zum Erstellen von mehreren Benutzerprofilen beispielsweise im *Firefox* Browser geht man wie folgt vor:

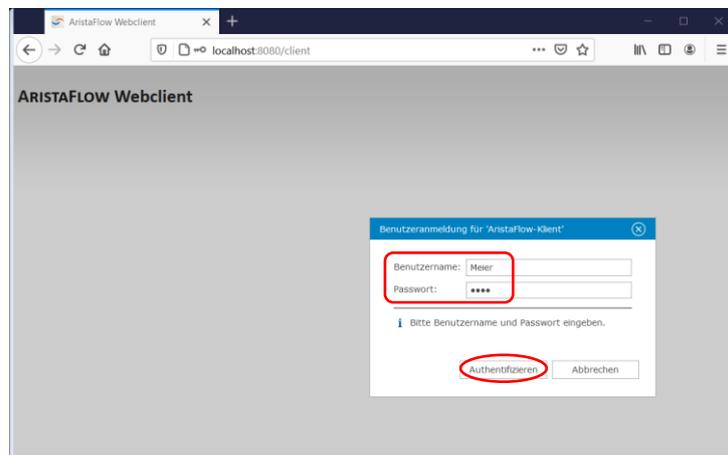
Man gibt den Befehl *firefox.exe -ProfileManager* im Startmenü von Windows an.

Oder alternativ drückt man die Tastenkombination [Windows] + [R]. Dann erscheint das „Ausführen Fenster“. Geben Sie den Befehl *firefox.exe -p* ein und bestätigen Sie mit OK.

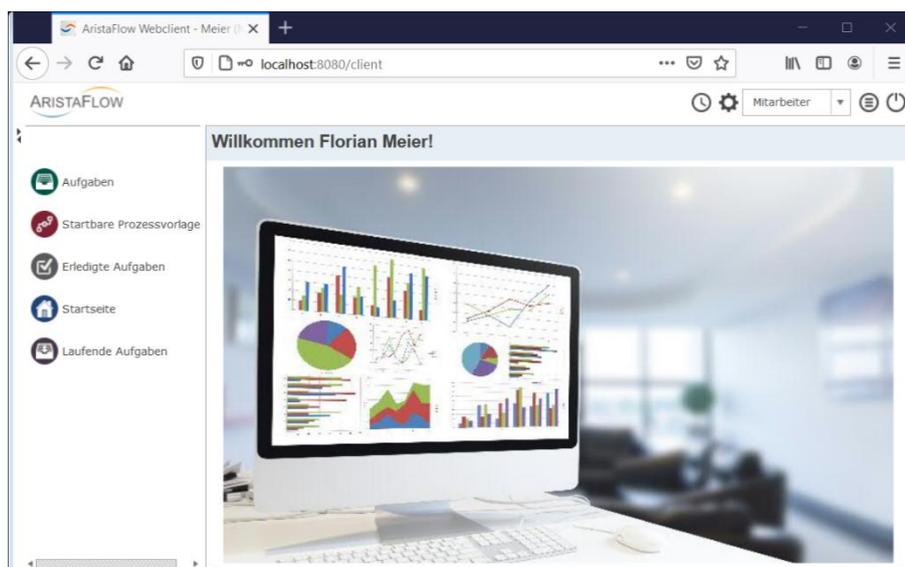
Im nun geöffneten *Profil-Manager* können Sie weitere Profile erstellen bzw. existierende umbenennen, löschen und starten. Normalerweise legt Firefox automatisch ein Profil mit dem Namen „default“ an. Nun legen wir zusätzlich für jeden Bearbeiter (hier Mitarbeiter, Vorgesetzter und Sachbearbeiter) ein eigenes Profil an und melden diese anschließend in einer eigenen Browser-Instanz auf dem *WebClient* an.



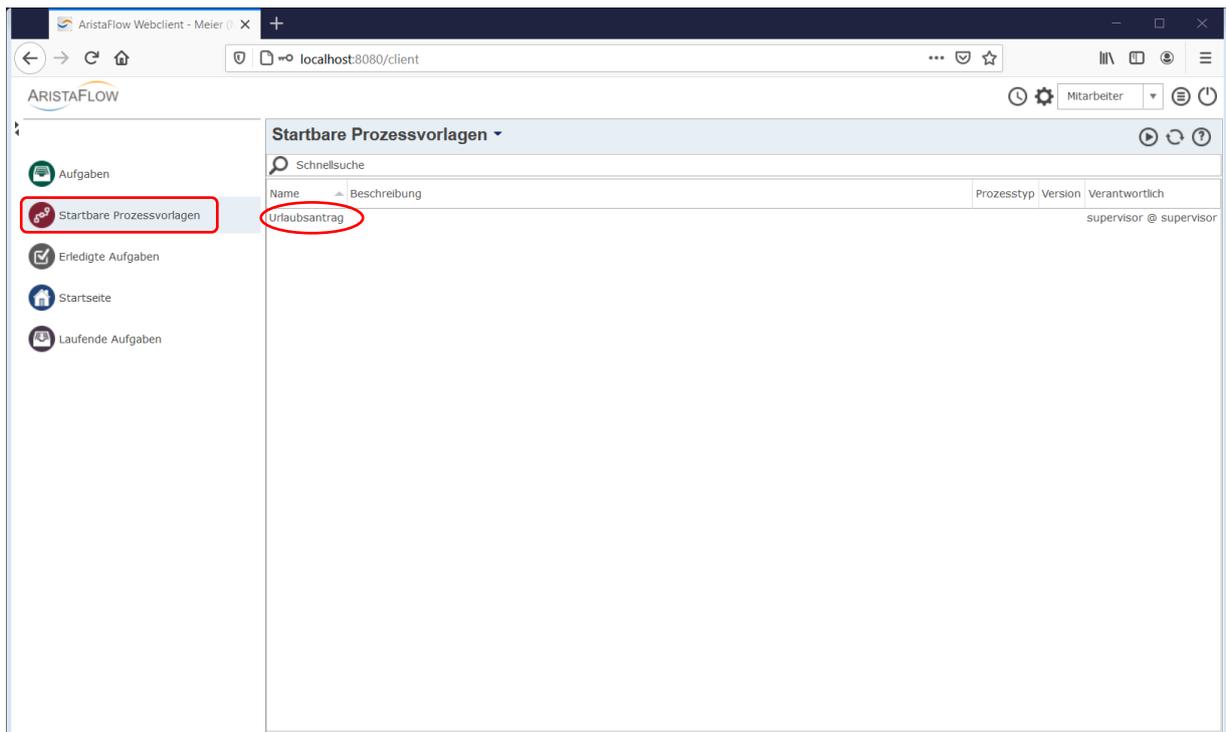
Wir müssen vor jedem Firefox-Start den Profil-Manager wie oben beschrieben öffnen und das geeignete Profil auswählen. Geben Sie dann die Adresse <http://localhost:8080/client> ein und melden Sie den entsprechenden Benutzer mit dem Passwort an.



Nach dem bereits bekannten *Log-in* startet die *WebClient* Oberfläche. Wenn alle Bearbeiter angemeldet sind, können diese nun Prozesse starten und Arbeitsschritte ausführen.



Starten einer Instanz des Prozesses „Urlaubsantrag“:  
Im Client des Mitarbeiters „Meier“ → Link *Startbare Prozessvorlagen* anklicken (links) → Doppelklick auf den Prozess „Urlaubsantrag“.



#### Hinweis:

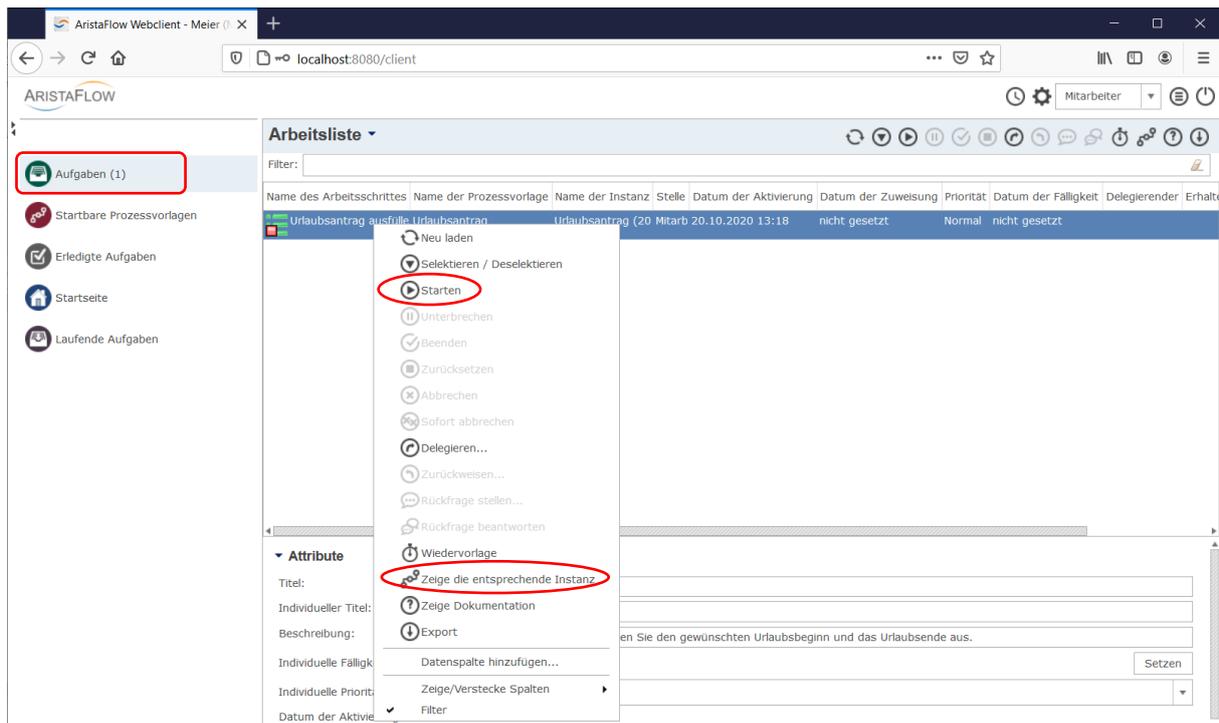
Wenn keine Prozessvorlage unter *Startbare Prozessvorlagen* zu sehen ist, dann kontrollieren Sie die *Starter Rule* (Starterregel) unter *Template Status* im Process Template Editor. Zum Testen tragen Sie als Starter wie im Kapitel *Deployment des Prozesses* beschrieben „Agent()“ ein. Damit kann der Prozess von jedem Bearbeiter gestartet werden.

Im Arbeitsbereich bei „Aufgaben“ wird sofort angezeigt, dass eine neue Aufgabe für den Mitarbeiter „Meier“ vorliegt. Wurde bei der Modellierung die Option *autostart* aktiviert, wird das Formular zur Eingabe der Urlaubsdaten automatisch geöffnet; ansonsten wechseln Sie über den Link „Aufgaben“ zur Arbeitsliste. Hier sehen Sie den neuen Schritt „Urlaubsantrag ausfüllen“.

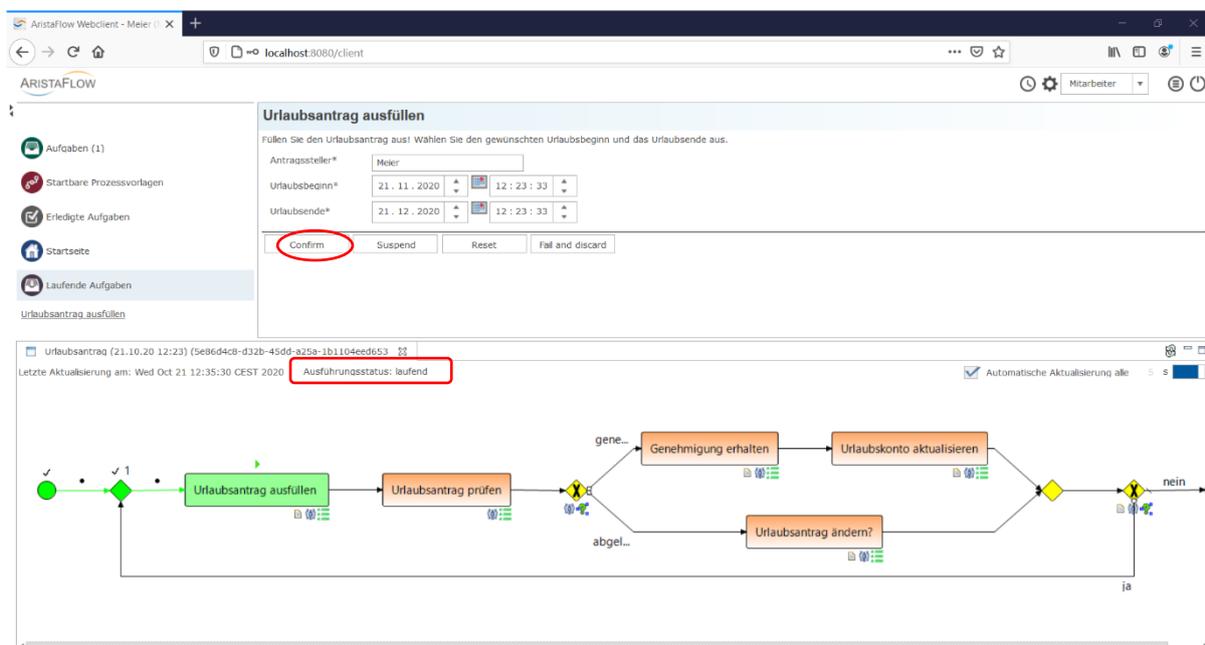
Starten Sie nun den Arbeitsschritt durch Doppelklick oder durch Anwahl im Rechtsklickmenü → Starten. Es öffnet sich das Formular zur Eingabe der Urlaubsdaten.

#### Tipp:

Der *WebClient* bietet genauso wie der *RichClient* eine einfache Möglichkeit des Echtzeitmonitorings. Damit können Sie den Prozessablauf graphisch nachvollziehen. Durch Rechtsklick auf einen Arbeitslisteneintrag und Auswahl von *Zeige die entsprechende Instanz* kann die zugehörige Instanz und deren momentaner Ausführungszustand angezeigt werden. In dieser Ansicht kann auch bei Schritten, bei denen der aktuelle Mitarbeiter nicht beteiligt ist, jederzeit über das Update Symbol der aktuelle Zustand der Instanz abgerufen werden.

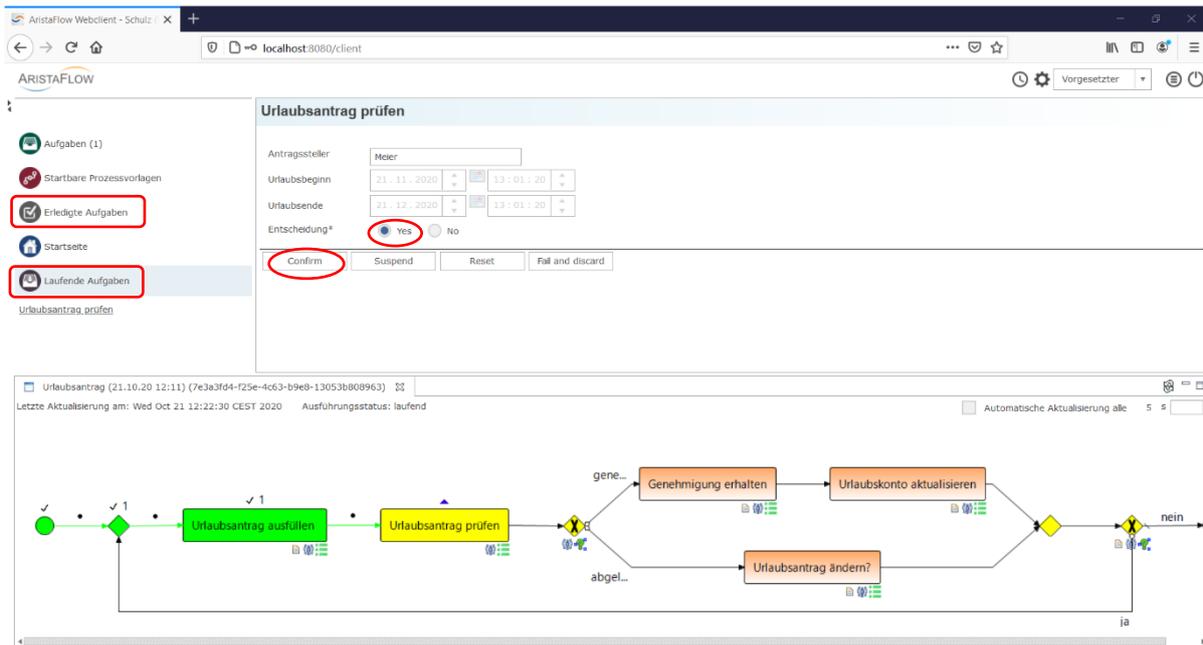


Nach Eingabe der Daten wird das Formular mit *Confirm* bestätigt. Ein Anklicken von *Suspend* bewirkt das Speichern der bereits eingegebenen Daten mit nachfolgender Suspendierung des Prozessschrittes. Dies ermöglicht es Schritte zu einem späteren Zeitpunkt weiterzubearbeiten.

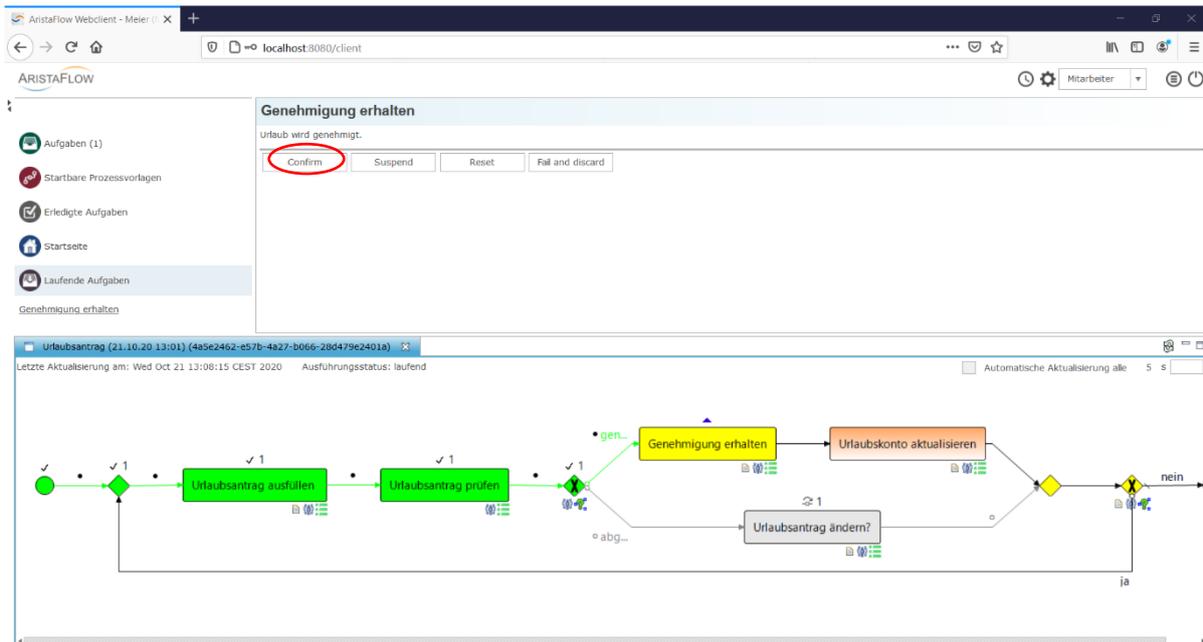


Nach Drücken von *Confirm* ist der Schritt beendet und wird an den Vorgesetzten von „Meier“ (Herr „Schulz“) zur Prüfung weitergeleitet.

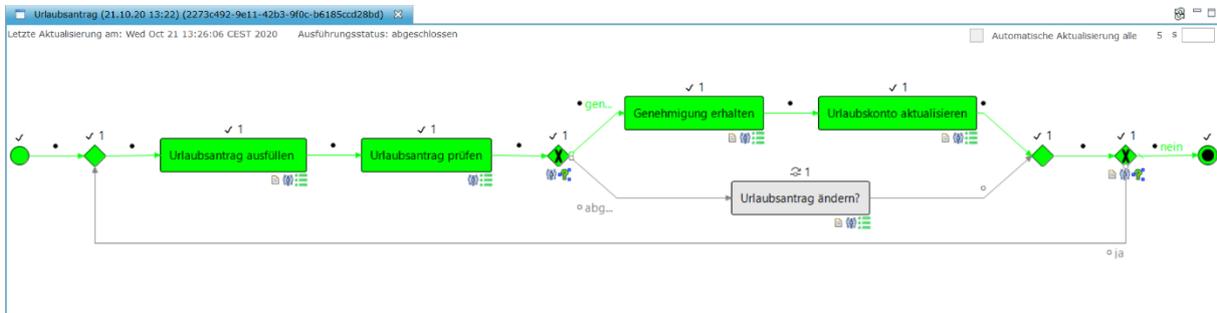
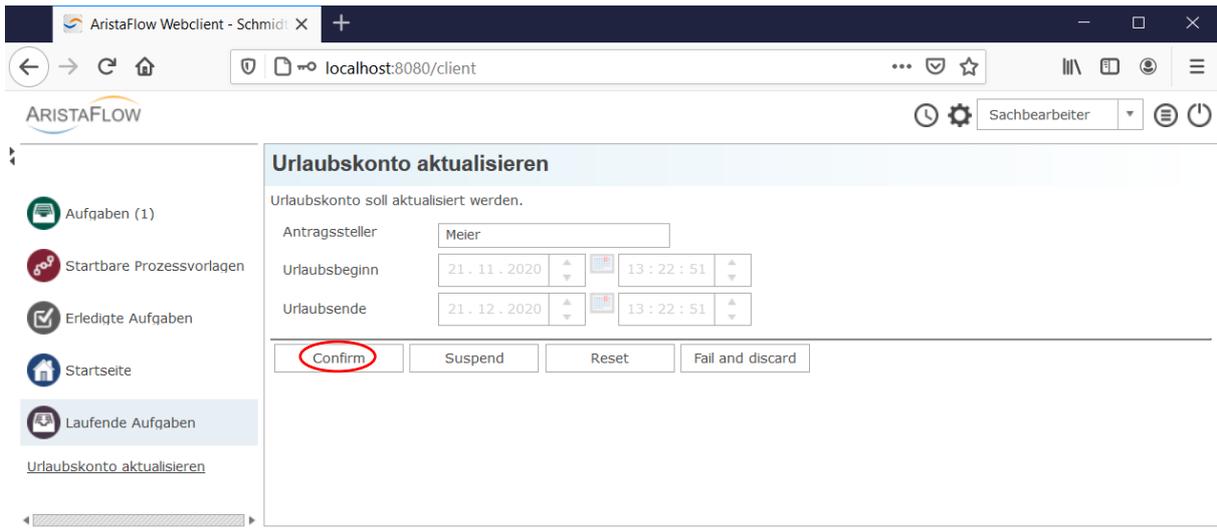
Die Arbeitslisten (Link „Aufgaben“) der angemeldeten Benutzer zeigen im Folgenden exakt diejenigen Prozessschritte an, die von den jeweiligen Benutzern ausgeführt werden sollen. Unter dem Link „Laufende Aufgaben“ bzw. „Erledigte Aufgaben“ finden Sie die aktuell auszuführenden bzw. schon erledigten Tätigkeiten.



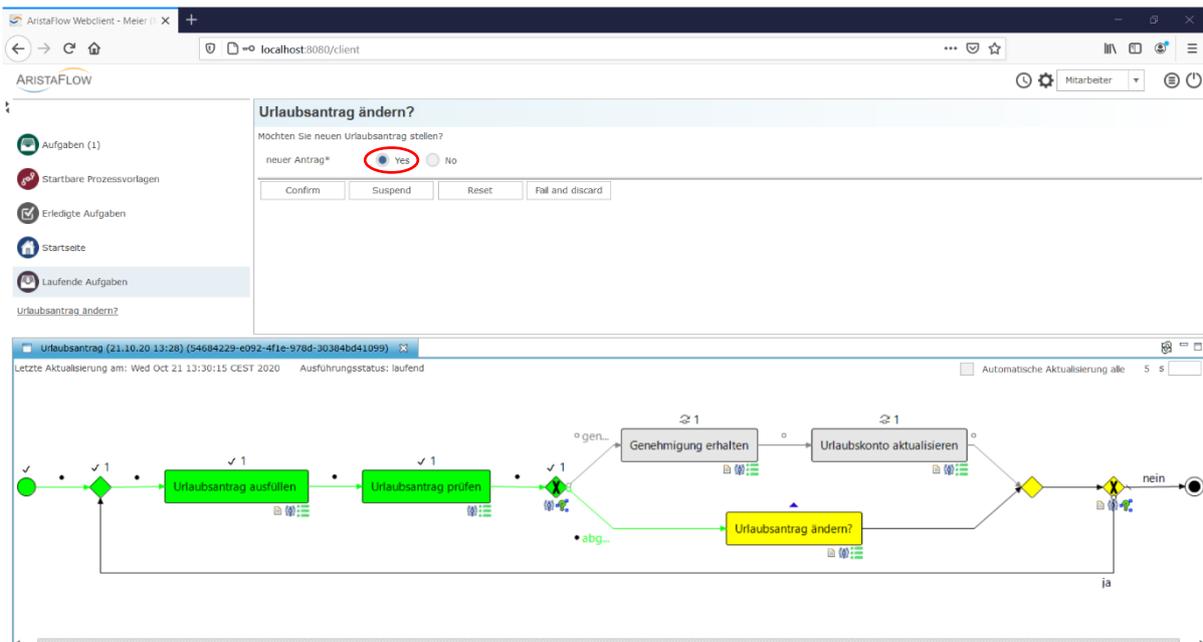
Nun muss der Vorgesetzte „Schulz“ entscheiden, ob der Urlaub bewilligt oder abgelehnt wird. Die Abarbeitung erfolgt analog zum Arbeitsschritt „Urlaubsantrag ausfüllen“. Bei Genehmigung erhält der Antragssteller „Meier“ im nächsten Schritt eine Bestätigung, dass sein Urlaub genehmigt wurde. Mit *Confirm* beendet er den Schritt.



Nun ist der Sachbearbeiter „Schmidt“ dran; er muss das Urlaubskonto durch einen Klick auf *Confirm* aktualisieren. Danach endet der Prozess.



Bei Ablehnung kann der Mitarbeiter „Meier“ einen neuen Urlaubsantrag stellen. Bei erneuter Antragsstellung beginnt der Prozess von vorne.



Wenn kein neuer Antrag gestellt wird, dann erreicht der Graph den Endpunkt und der Prozess ist beendet.

## Überwachen des Prozesses mit dem ARISTAFlow Monitor

Neben der einfachen Möglichkeit des Echtzeitmonitorings im Rich- und WebClient bietet die ARISTAFlow-BPM-Suite mit dem ARISTAFlow-Monitor zusätzlich ein wesentlich mächtigeres Monitoring Tool für Administratoren, Modellierer und Analysten.

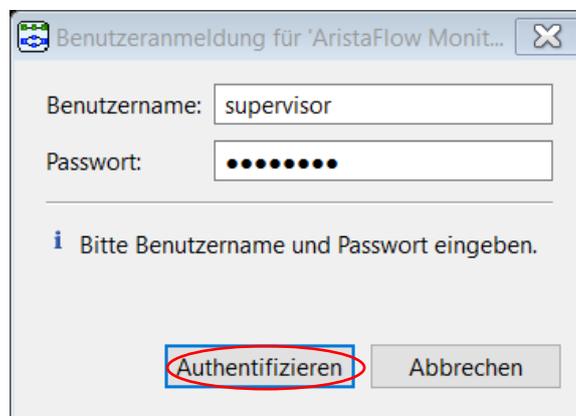
Starten des „Monitors“ über das Startmenü

**AristaFlow BPM Suite -> Monitor**

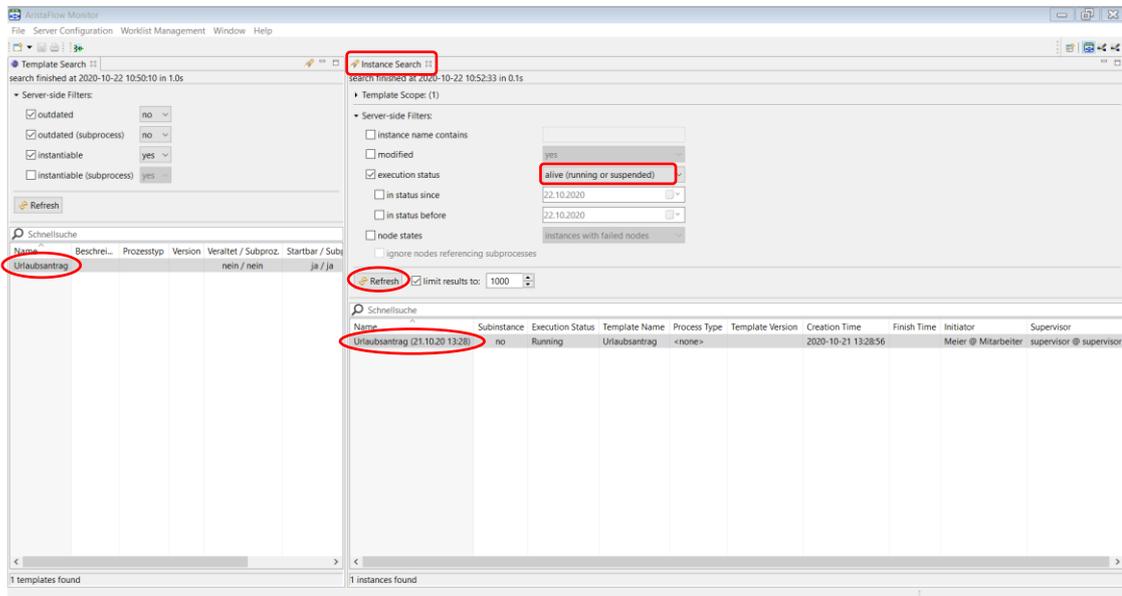
oder alternativ im Installationsverzeichnis unter dem Pfad

**...\AristaFlow-Chronos\monitoring\Monitoring.exe**

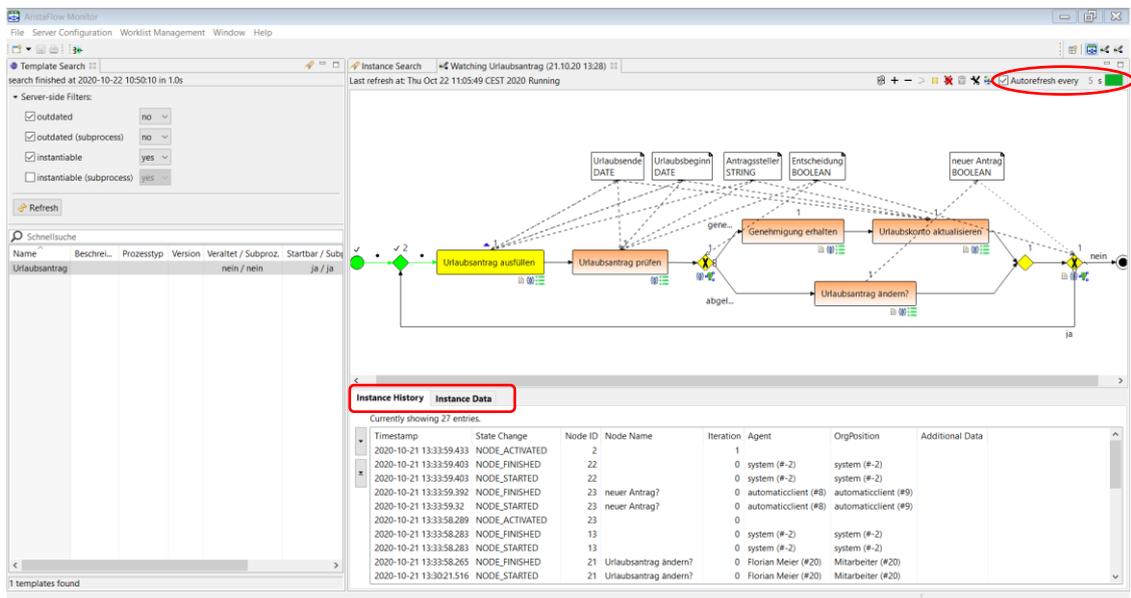
1. Authentifizieren Sie sich im erscheinenden Login-Dialog mit einem gültigen Benutzernamen und zugehörigem Passwort (im Auslieferungszustand mit **supervisor/password**).



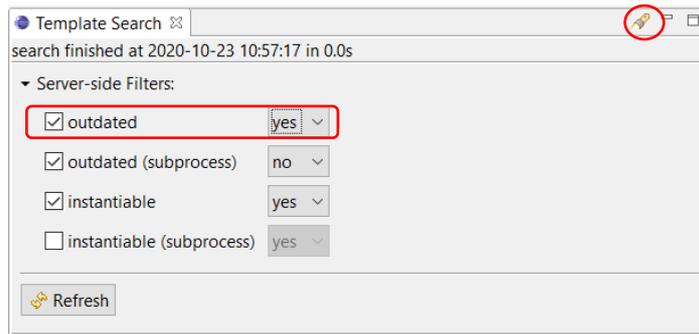
2. Selektieren Sie die Instanz (links unten) und klicken Sie im Rechtsklickmenü auf *Open Instance Search for Template(s)*. Klicken Sie im Reiter *Instance Search* unten auf *Refresh*. Alle laufenden oder suspendierten (running or suspended) Instanzen des Urlaubsantrag-Templates werden aufgelistet. Durch Doppelklick auf eine Instanz in der Liste können Sie die ausgewählte Instanz im Monitor beobachten.



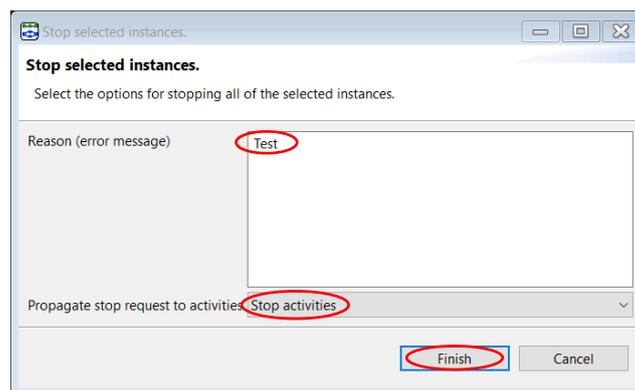
- Es wird die Instanz inklusive ihrem Ausführungszustand, der Ausführungshistorie und Daten (Instance History / Instance Data) angezeigt. Über die Checkbox *Autorefresh* (rechts oben) kann die automatische Aktualisierung aktiviert werden. Dadurch wird die Instanz im angegebenen Zeitintervall automatisch mit dem Server synchronisiert.



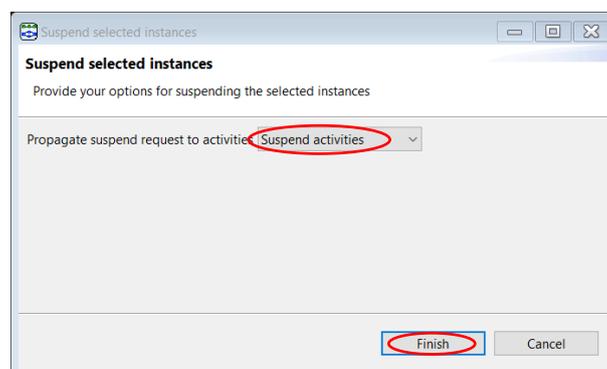
- Wenn Sie unabhängig vom Template alle Instanzen also auch Instanzen von anderen Templates anzeigen möchten, dann klicken Sie im Reiter *Template Search* oben rechts auf das Taschenlampensymbol (siehe Screenshot). Zusätzlich können Sie die veralteten (outdated) Templates wiederfinden, wenn Sie die Filter bei der Templatesuche entsprechend anpassen.



5. Eine Instanz von einer Prozessvorlage wird erstellt, wenn die Prozessvorlage gestartet wird. Dann wird der Startpunkt des Graphen automatisch ausgeführt. Mit dem Erreichen des Endpunkts des Graphen wird die Instanz in den Endzustand (*finished*) versetzt. Um eine laufende Instanz abubrechen (hier *abort selected Instance(s)*), müssen Sie einen Grund für das Abbrechen eingeben. Beim Abbrechen können Sie entscheiden, ob die Aktivitäten der Instanzen weiterlaufen sollen oder sofort gestoppt werden. Dazu führen Sie einen Rechtsklick auf der gewählten Instanz aus und wählen -> *abort selected Instance(s)*. -> Im erscheinenden Fenster den Grund eingeben und unten in der Combo Box auswählen, was mit den Aktivitäten passieren soll.



6. Darüber hinaus können Sie die Instanz auch suspendieren und zu einem späteren Zeitpunkt ausführen. Dabei können Sie analog zum Abbrechen die Aktivitäten der Instanz auch suspendieren (zwischenspeichern) oder laufen lassen. Selektieren Sie die Instanz, wählen Sie im Rechtsklickmenü -> *suspend selected Instance(s)*. und treffen Sie eine Entscheidung was mit den Aktivitäten geschehen soll.



- Durch Setzen des Filters bei der Instanzsuche kann man alle Instanzen, die abgebrochenen oder suspendierten genauso wie die beendeten, anzeigen lassen.

The screenshot shows the 'Instance Search' window with the following filters applied:

- execution status: **aborted** (circled in red)
- limit results to: 1000

The search results table is as follows:

Name	Subinstance	Execution Status	Template Name	Process Type	Template Version	Creation Time	Finish Time	Initiator
Urlaubsantrag (21.10.20 13:28)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-21 13:28:56	2020-10-22 11:55:31	Meier @ Mitarbeiter
Urlaubsantrag (22.10.20 11:56)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-22 11:56:38	2020-10-22 11:58:58	Meier @ Mitarbeiter
Urlaubsantrag (22.10.20 12:00)	no	Aborted (soft)	Urlaubsantrag	<none>		2020-10-22 12:00:55	2020-10-22 12:04:42	Meier @ Mitarbeiter

- Schließlich können Sie abgebrochene oder beendete (*not alive*) Instanzen durch Klick im Rechtsklickmenü -> *Delete selected Instance(s)* löschen.

The screenshot shows the 'Instance Search' window with the filter 'execution status' set to 'not alive (finished or aborted)' (circled in red). A dialog box titled 'Delete selected instances' is open, asking 'Do you really want to delete the selected instances?' with 'OK' and 'Cancel' buttons. The 'OK' button is circled in red.

The search results table is as follows:

Name	Subinstance	Execution Status	Template Name	Process Type	Template Version	Creation Time	Finish Time	Initiator
Urlaubsantrag (20.10.20 12:35)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-19 13:24:42	2020-10-20 13:08:59	supervisor @ supervisor
Urlaubsantrag (20.10.20 12:40)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-19 13:29:04	2020-10-20 13:08:43	supervisor @ supervisor
Urlaubsantrag (20.10.20 12:43)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-20 12:35:30	2020-10-20 13:08:24	Meier @ Mitarbeiter
Urlaubsantrag (20.10.20 12:45)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-20 12:40:33	2020-10-20 13:08:02	Schulz @ Vorgesetzter
Urlaubsantrag (20.10.20 12:47)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-20 12:43:08	2020-10-20 13:07:44	Meier @ Mitarbeiter
Urlaubsantrag (20.10.20 12:45)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-20 12:45:33	2020-10-20 13:07:26	Meier @ Mitarbeiter
Urlaubsantrag (20.10.20 12:47)	no	Aborted (hard)	Urlaubsantrag	<none>		2020-10-20 12:47:00	2020-10-20 13:07:05	Meier @ Mitarbeiter

- Um auch bereits beendete Schritte wie beispielsweise das beendete Urlaubsantragsformular noch einmal anzeigen zu können, klicken Sie mit der rechten Maustaste auf den entsprechenden Arbeitsschritt im Graphen und wählen Sie *Re-open activity (read-only)*. Das Formular wird mit den zum damaligen Zeitpunkt eingegebenen Daten angezeigt. Diese dienen natürlich nur der Anzeige und können nicht mehr verändert werden. Diese Funktion ist nur für beendete also für grüne Knoten im Graph aktiv.

AristaFlow Monitor  
File Server Configuration Worklist Management Window Help

Template Search search finished at 2020-10-23 09:32:56 in 0.0s

Server-side Filters:

- outdated no
- outdated (subprocess) no
- instantiable yes
- instantiable (subprocess) yes

Refresh

Schnellsuche

Name	Beschreibu...	Prozesstyp	Version	Veraltet / Subproz.	Startbar / Subpr...
Urlaubsantrag				nein / nein	ja / ja

Instance Search Watching Urlaubsantrag (23.10.20 09:31) \*Urlaubsantrag ausfüllen

### Urlaubsantrag ausfüllen

Fillen Sie den Urlaubsantrag aus! Wählen Sie den gewünschten Urlaubsbeginn und das Urlaubsende aus.

Input Values

Antragssteller: Meier

Urlaubsbeginn: 25.10.2020 09:31:48

Urlaubsende: 30.10.2020 09:31:48

Close

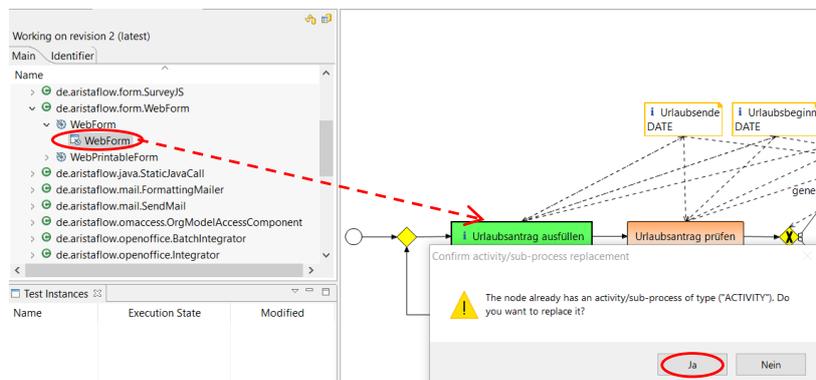
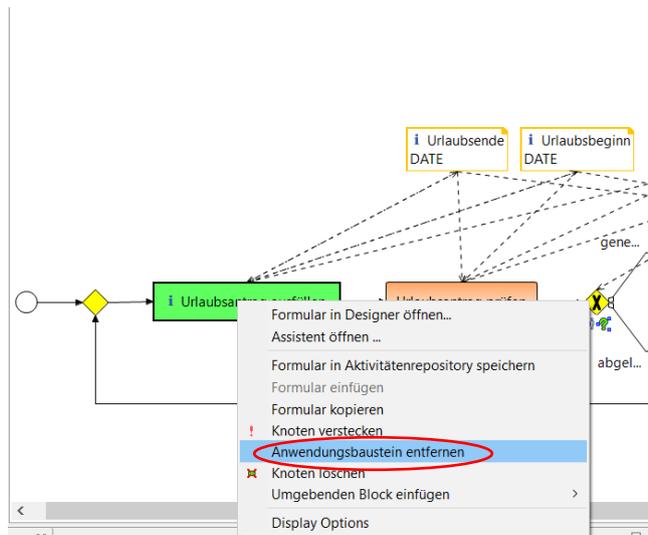
Zusätzlich bietet der Monitor auch administrative Funktionen, wie das Zurücksetzen von fehlgeschlagenen Arbeitsschritten und die Möglichkeit laufende Instanzen einfach und mit Korrektheitsgarantie an neue Gegebenheiten anzupassen.

# Erstellung eines Webformulars aus dem Activity Repository

ARISTAFLOW bietet mit dem *ARISTAFLOW-ActivityRepository* ein mächtiges Tool um nahezu beliebige Aktivitäten, Funktionen und Services in einen Prozess zu integrieren. Für den hier vorgestellten Urlaubsantragsprozess genügen einige wenige Standardaktivitäten, die wir bereits im Process Template Editor eingebunden haben, wie z.B. User Form des Arbeitsschritts „Urlaubsantrag ausfüllen“ oder die XOR-Verzweigung.

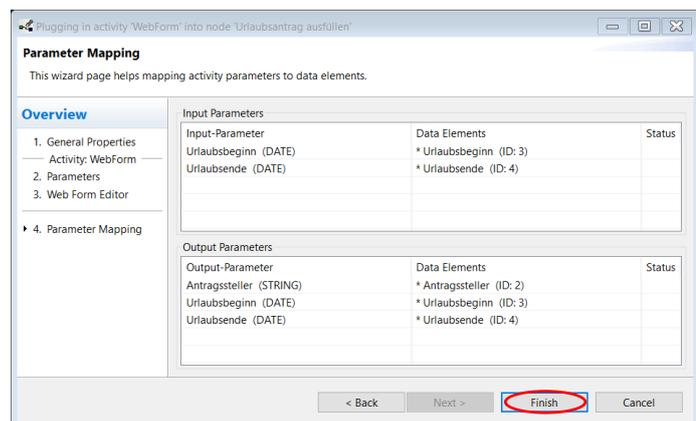
Wenn Sie anstelle des automatisch generierten Userformulars ein ansprechenderes Webformular möchten, können Sie dies über die Aktivität *Webform* gestalten.

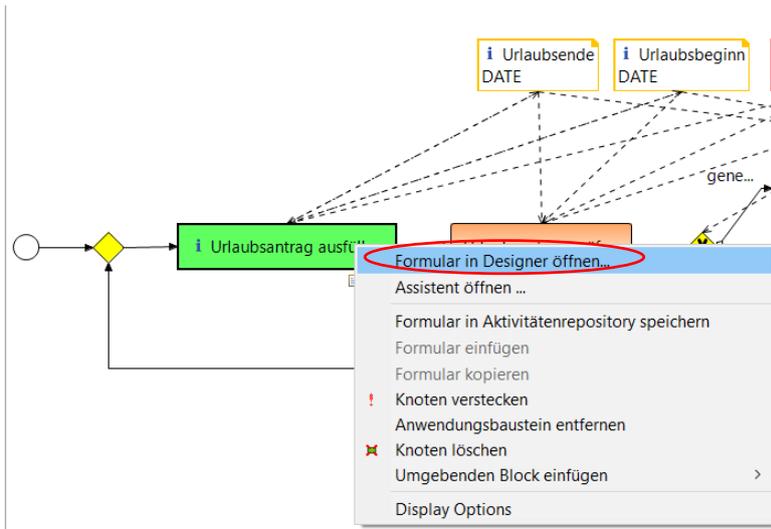
Dazu müssen Sie zuerst das Userform aus dem Knoten „Urlaubsantrag ausfüllen“ über Rechtsklickmenü -> *Anwendungsbaustein entfernen* löschen.



Oder Sie können das Userform per Drag & Drop gleich durch ein neues Webform ersetzen.

Nach dem Einbinden des Webforms öffnet sich gleich das bekannte Fenster zur Versorgung der Aktivität mit Input und Output Parametern. Durch wiederholtes Klicken von *Next* wird das Webform automatisch mit benötigten Parametern verbunden.





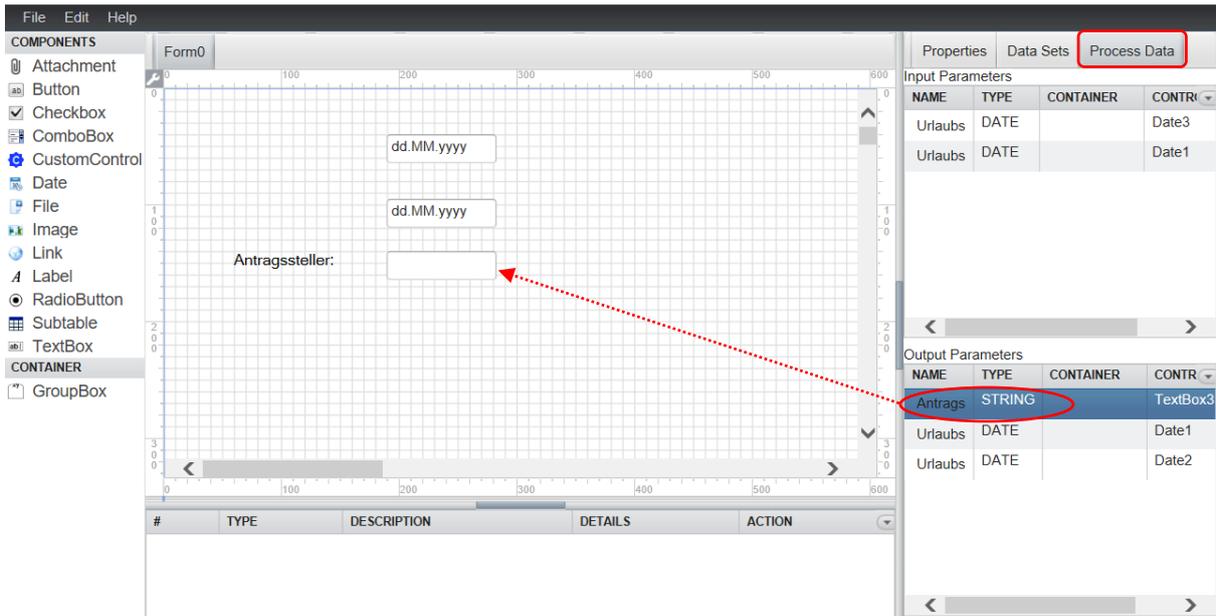
Nun können Sie das Formular per Rechtsklick auf den Knoten gestalten. ->Formular in Designer öffnen...

Im Designer Fenster befinden sich links die Komponenten und rechts die Eigenschaften, wie Datenquellen oder Design, die auf diese Komponenten angewendet werden können. In unserem Beispielsprozess müssen in diesem Schritt die drei Parameter (Urlaubsbeginn, Urlaubsende und Antragssteller) vom Benutzer eingegeben werden. Für Urlaubsbeginn und -ende brauchen Sie jeweils ein *Date*-Komponente als Eingabefeld. Das Einfügen von Komponenten erfolgt hier per Drag&Drop. Ziehen Sie zwei Datenfelder auf das Formular und binden Sie die Kalender-Funktion in diesen Feldern ein. Dazu selektieren Sie das Datumsfeld und setzen Sie *calendar* auf *true* im Reiter *Properties*.

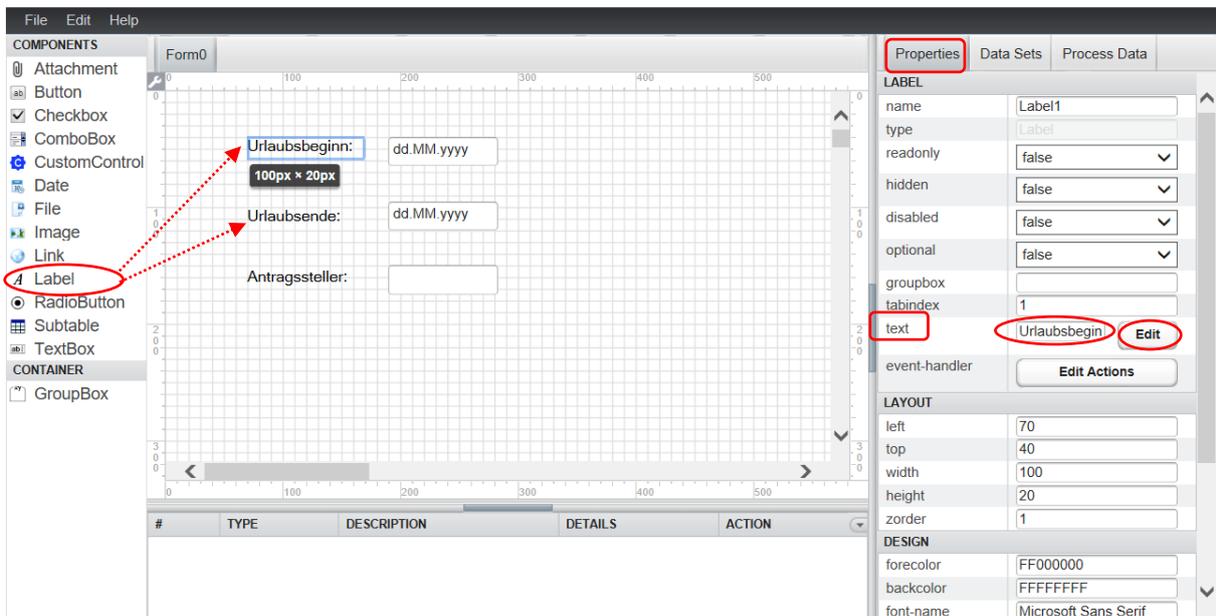
#	TYPE	DESCRIPTION	DETAILS	ACTION
1	DATE	dd.MM.yyyy	100px x 25px	
2	DATE	dd.MM.yyyy		

PROPERTY	VALUE
name	Date1
type	Date
readonly	false
hidden	false
disabled	false
optional	false
groupbox	GroupBox1
tabindex	1
dateformat	dd.MM.yyyy
timeformat	
calendar	true
value	
event-handler	Edit Actions

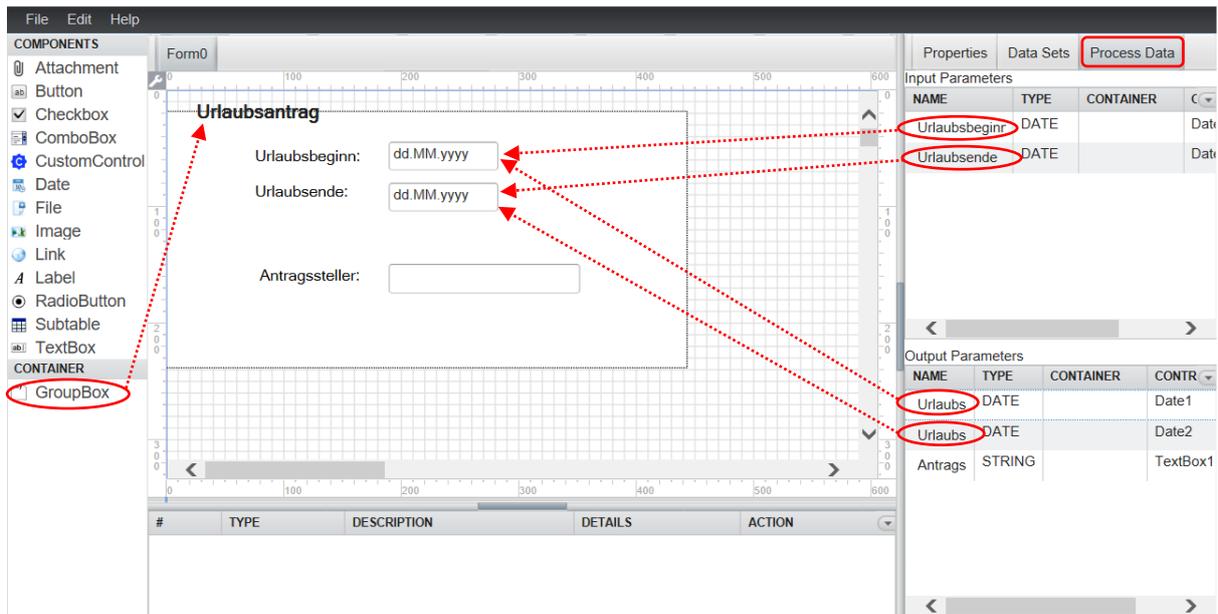
Alternativ können Sie die Eingabefelder von rechts im Reiter *Process Data* mit Hilfe von Input- und Outputparametern per Drag&Drop erstellen. Für den Antragssteller beispielsweise markieren Sie zuerst die erste Zeile „Antragsteller“ unter *Output Parameters* (blaue Markierung) und legen sie auf das Formular. So wird auch das Label mit dem Eingabefeld erstellt. Ansonsten müssen Sie das Label explizit hinzufügen.



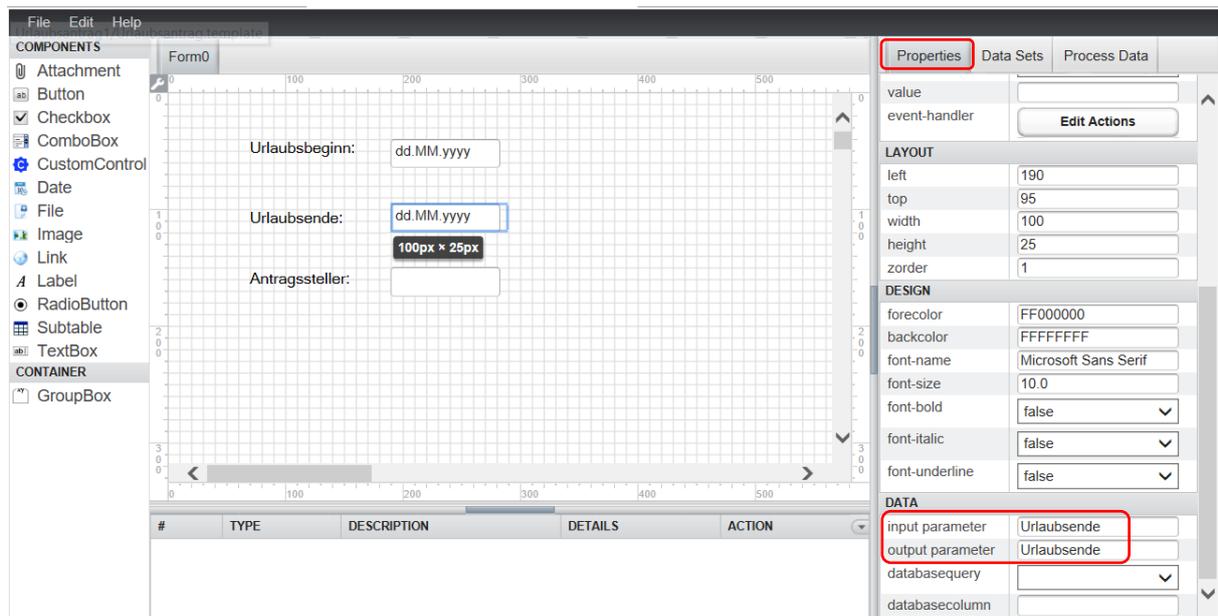
Nun setzen Sie wieder per Drag&Drop die Labels für die Komponenten. Die Labels benennen Sie entsprechend den Output-Parametern unter dem Reiter *Properties* (rechts oben). -> bei *text*, den Namen für Label eingeben -> *Edit* klicken. Das geöffnete Fenster *Translations* mit -> *Ok* bestätigen (siehe Screenshot).



Sie können die einzelnen Komponenten mit einem GroupBox „Urlaubsantrag“ zu einem Formular gruppieren. Nach der Erstellung der Eingabefelder müssen Sie diese mit Parametern verbinden (wenn diese nicht per Drag&Drop von rechts also über Process Data erstellt wurden). Dazu verknüpfen Sie die Input- und Output-Parameter wieder per Drag&Drop mit den entsprechenden Eingabefeldern.

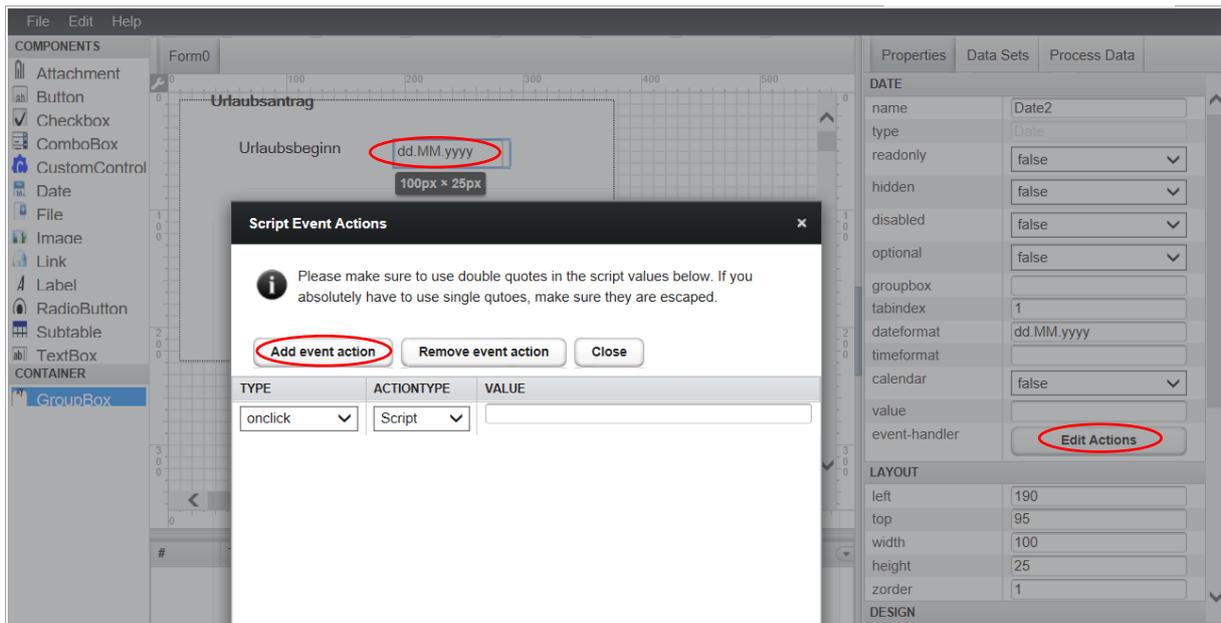


Es ist sehr empfehlenswert, vor dem Einsatz des Formulars die richtige Zuordnung der Parameter zu überprüfen. Dafür nach Selektieren des einzelnen Feldes unter *Properties* -> *DATA* nachschauen, ob Input und Output Parameter mit dem entsprechenden Feld übereinstimmen.

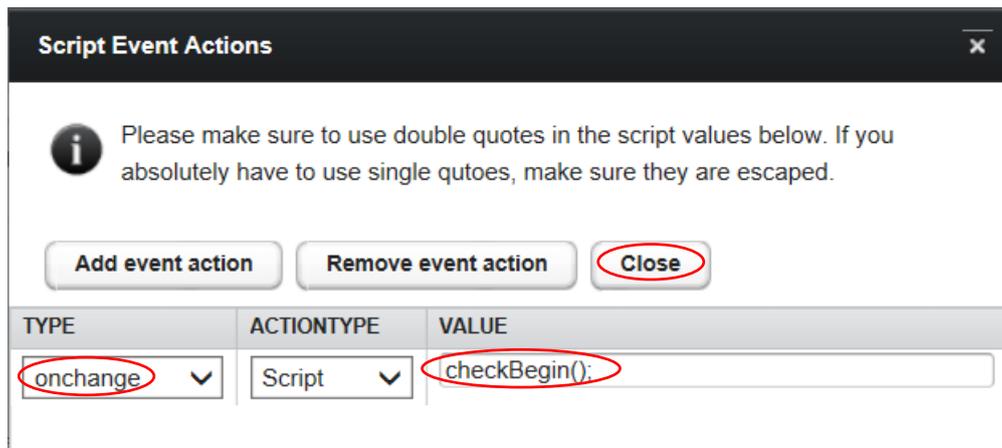


Als nächstes können Sie Ereignisse (*events*) in den Eingabefeldern auslösen und die Benutzereingabe dadurch steuern. Beispielsweise können Sie durch *Events* überprüfen, ob der eingegebene Urlaubsbeginn zeitlich vor dem aktuellen Datum oder nach dem Urlaubsende liegt. Dafür müssen Sie kleine Javascript-Funktionen in *Events* hinterlegen.

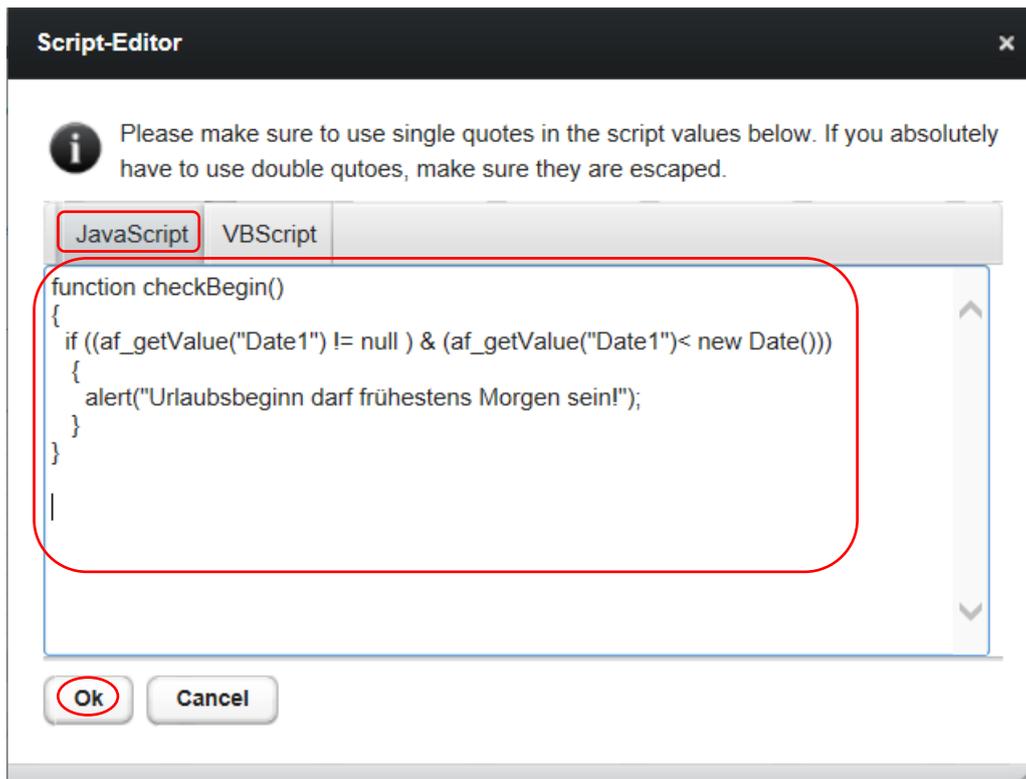
Um *Events* einzufügen, gehen Sie auf das Eingabefeld „Urlaubsbeginn“ und klicken Sie *Edit Actions* (rechts Mitte). Im geöffneten Fenster *Script Event Actions* fügen Sie ein *Event* ein. -> *Add event action*.



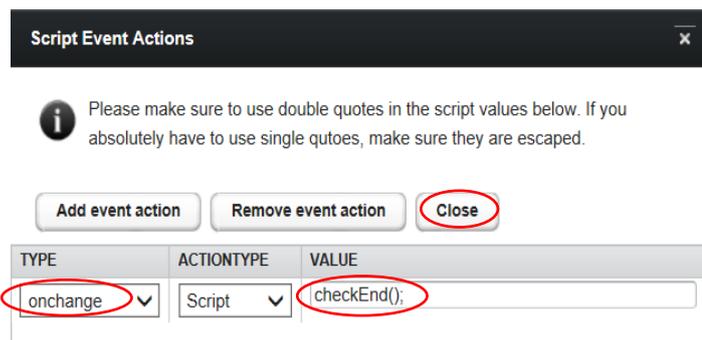
Die Überprüfung, ob der Urlaubsbeginn vor dem morgigen Datum liegt, soll bei jeder Änderung des ersten Feldes stattfinden. Daher wählen sie als Eventstyp *onchange* und lassen Actiontype auf *Script*. Nun geben Sie den Namen für die Javascript Funktion mit Klammern und Bindestrich wie z.B. „checkBegin();“ ein. Schließen Sie das Fenster mit einem Klick auf *Close*.



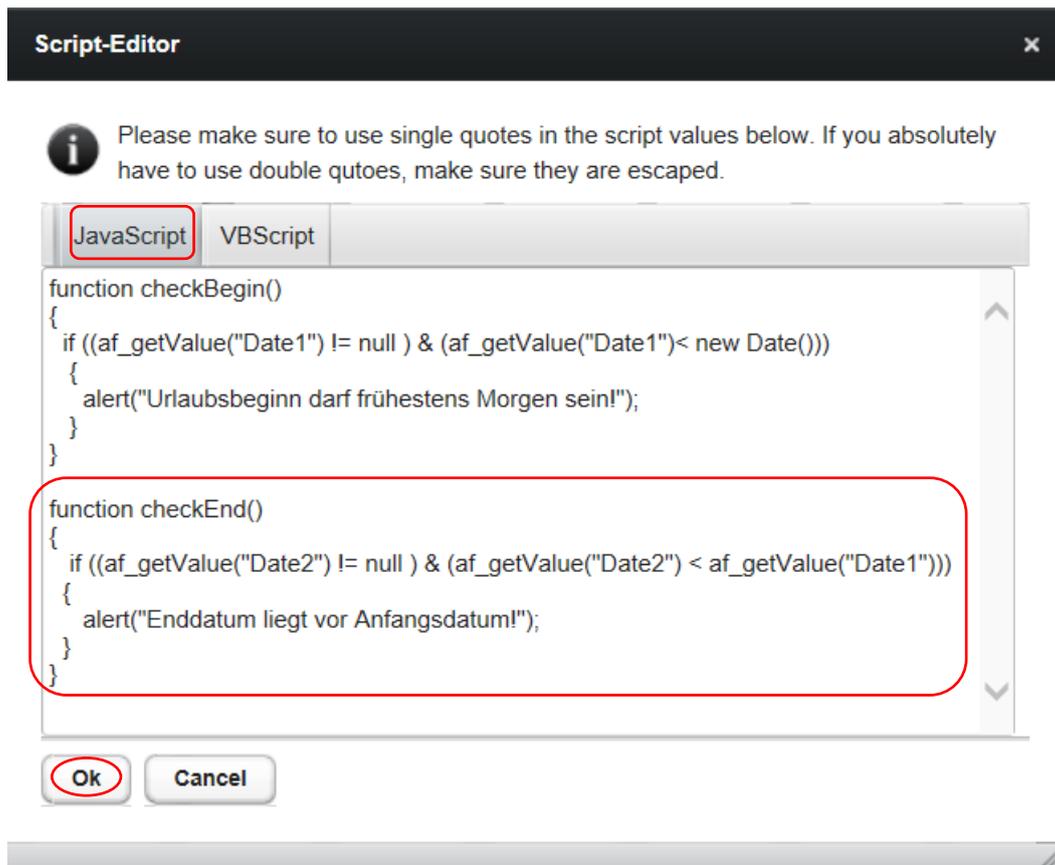
Die Funktion `checkBegin()` muss jetzt im Script Editor implementiert werden. Den Script Editor öffnen Sie durch Klick auf *Edit* (links oben im Menü des Designer Editors) und *->Scripts...* Im Script Editor geben Sie den Code unter *JavaScript* wie im Screenshot ein und speichern mit Klick auf *OK*.



Zur Überprüfung des Urlaubsendes auf Gültigkeit lösen Sie wieder ein *onchange* Event aus und geben den Namen für die JavaScript Funktion „checkEnd();“ ein. Danach schließen Sie das Fenster.

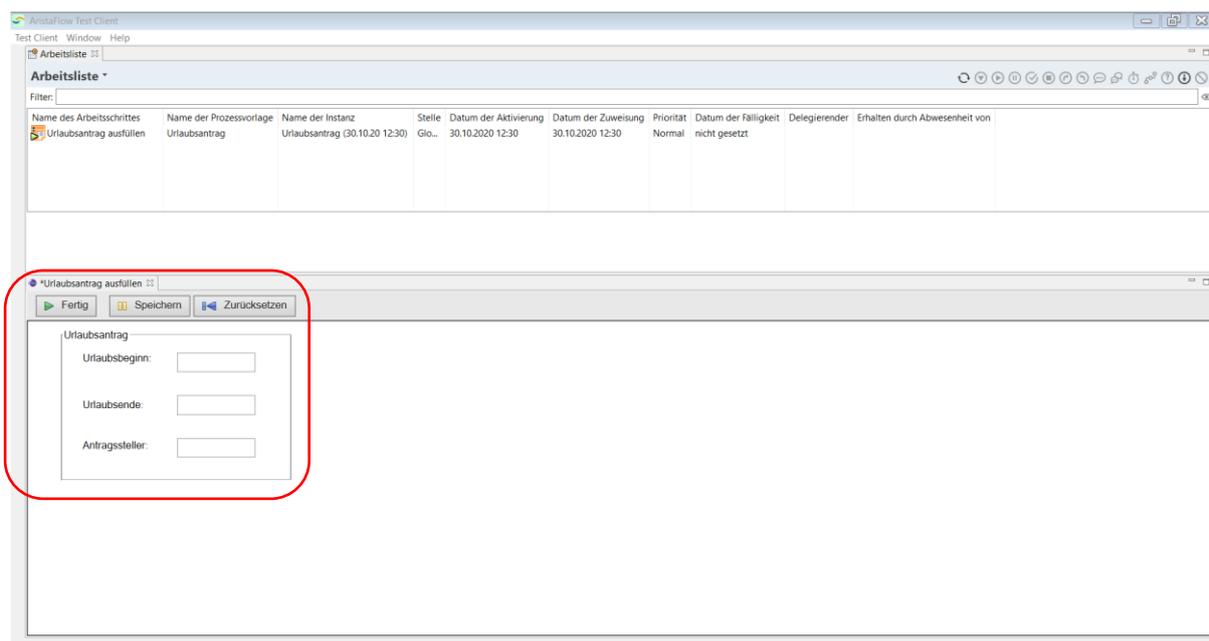


Tippen Sie den Code im JavaScript Editor für die Funktion „checkEnd();“ wie im folgenden Screenshot ein. Achten Sie dabei besonders auf die Klammern und Bindestriche.

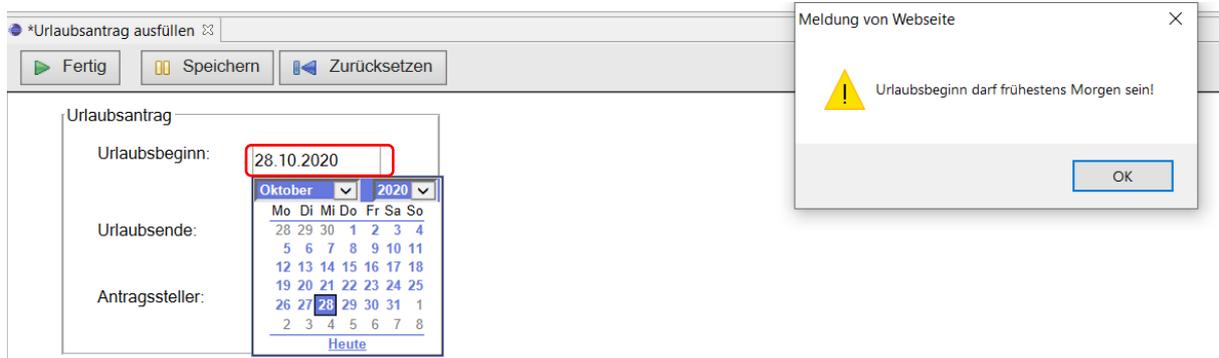


## Test des Webforms im Test-Client

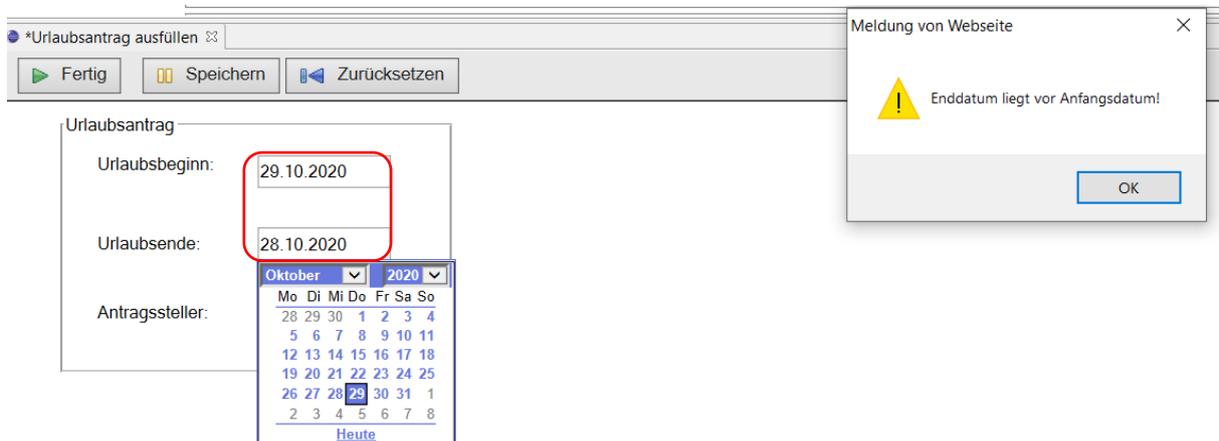
Nach der Speicherung der Änderungen können Sie das Webform im Test-Client auf Funktion testen. Selektieren Sie das Template und klicken auf *Run Template* im Process Template Editor. Der Test-Client öffnet sich mit dem neu erstellten Webform. Das Formular enthält schon vorimplementierte Buttons zum Senden, Suspendieren und Zurücksetzen (*Fertig*, *Speichern*, *Zurücksetzen*).



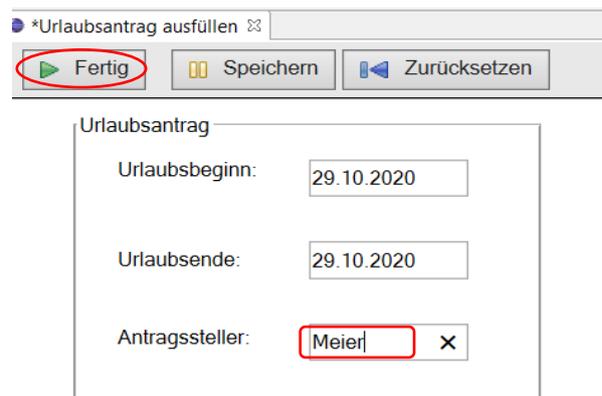
Wählen Sie ein Datum aus dem Kalender. Wird als Datum heute oder ein früheres Datum gewählt, erscheint die implementierte Fehlermeldung, die den Anwender auf eine falsche Eingabe hinweist.



Genauso kommt bei Urlaubsende eine Fehlermeldung, wenn das Enddatum vor dem Anfangsdatum liegt.



Geben Sie den Namen für den Antragssteller ein und senden Sie das Formular durch den *Fertig* – Button ab.



Im nächsten Schritt „Urlaubsantrag prüfen“ finden Sie die durch Webform gesendten Daten wieder.

AristaFlow Test Client  
 Test Client Window Help

Arbeitsliste ✕

**Arbeitsliste** ▾

Filter:

Name des Arbeitsschrittes	Name der Prozessvorlage	Name der Instanz	Stelle	Datum der Aktivierung
 Urlaubsantrag prüfen	Urlaubsantrag	Urlaubsantrag (...)	Glo...	28.10.2020 10:44

▾ **Attribute**

Titel:

Individueller Titel:

\*Urlaubsantrag prüfen ✕

**Urlaubsantrag prüfen**

Antragssteller

Urlaubsbeginn

Urlaubsende

Entscheidung\*  Yes  No

Nach Bedarf können Sie das Userform für den Vorgesetzten auch durch ein Webform ersetzen.

## Epilog

Mit AristaFlow BPM Suite haben Sie die Möglichkeit Ihre Geschäftsprozesse mit Bearbeitern, Datenflüssen und Aktivitäten zu modellieren, auszuführen und zu überwachen. Mit den benutzerfreundlichen Editoren und Tools von AristaFlow ist dies für Fachanwender und Prozessmodellierer kein Problem und erfordert keine/kaum Programmierkenntnisse. Mit dem Korrektheitsprinzip während der Template-Erstellung stellen Sie sicher, dass der modellierte Prozess auch korrekt umgesetzt und durchgeführt wird. Zusätzlich gibt es Tests, die gleich nach der Modellierung den Prozessablauf untersuchen; oder Tests, die die Durchführung des Prozesses mit Clients und ihren Aktivitäten komplett visualisieren.

Dabei sind spätere Veränderungen an Prozessen und ihre direkte Übernahme in das Template in jeder Phase, z.B. beim Testen oder auch beim Überwachen möglich.

In der Regel reichen die standardmäßig gelieferten Aktivitäten aus, um nahezu alle notwendigen Funktionen oder Services in den Prozessen einzubinden. Bei anspruchsvolleren Aktivitäten jedoch, können diese nach Bedarf weiterentwickelt und angepasst werden.

AristaFlow BPM Suite ist ein Muss für jedes Unternehmen, das seine komplexen Geschäftsaufgaben optimieren oder automatisieren und dadurch die Zusammenarbeit ihrer zahlreichen, verteilten Mitarbeiter vereinfachen will.

Diese Dokumentation soll Ihnen einen ersten Überblick über die Funktionalitäten unserer AristaFlow BPM Suite verschaffen. Dabei kann das Tutorial nur einen kleinen Bruchteil der gesamten Funktionalität der AristaFlow BPM Suite behandeln. Für ausführlichere Informationen oder Anleitungen nehmen Sie bitte Kontakt mit uns auf!