

 **Fix Me!** This page is not fully translated, yet. Please help completing the translation.  
(remove this paragraph once the translation is finished)

# TIM Smartform Suite - DRAFT

## 1. Introduction and differentiation of the TIM Smartform Suite from HTML smartforms

- Configuration instead of programming
- Quick smartform design. Validation, layout and structure
  - [Access to the Tim smartform suite](#)
- Ausbaustufen:
  - [TIM Smartform Designer](#)
  - [TIM Smartform Architect](#)
  - [TIM Smartform Expert](#)
  - Every created configuration can be opened in every version but can only be altered if the used version includes the functionalities.

## 2. Access to the Tim smartform suite

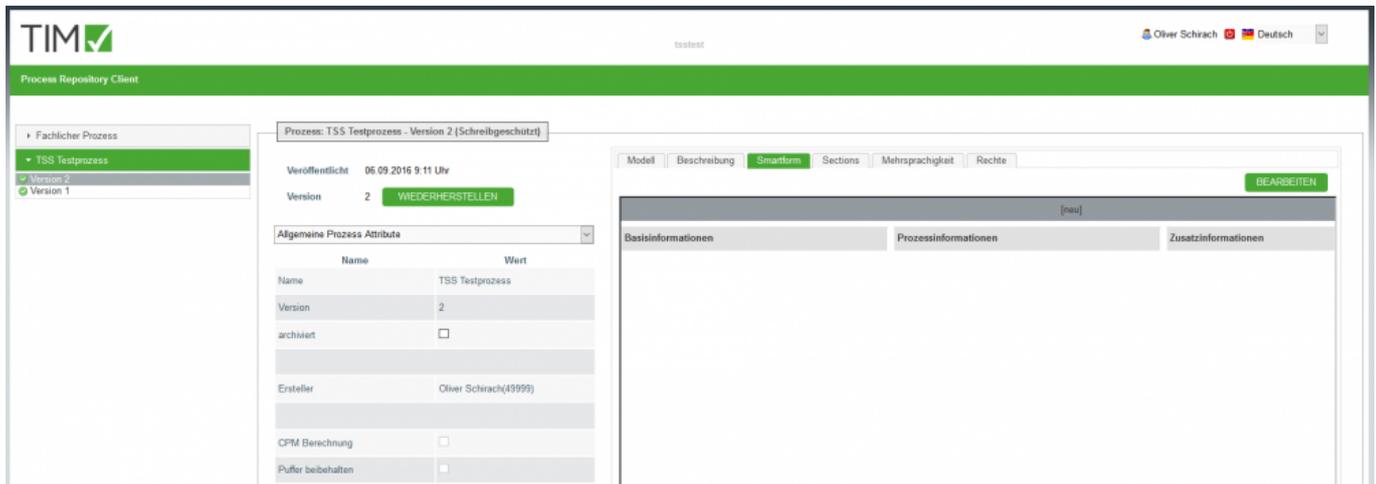
### 2.1 Systemrollen für den Zugang zur TSS

A user has to have the following TODO:roles [Systemrollen](#) to use the corresponding version of the TIM smartform suite:

- „smartform-designer“ to gain excess to TODO [TIM Smartform Designer](#)
- „smartform-architect“ to gain excess to TODO [TIM Smartform Architect](#)
- „smartform-expert“ to gain excess to TODO [TIM Smartform Expert](#)

### 2.2 Excess via repository

The acces to the configurator of the TSS happens via TODO [TIM Repository](#). In the Tab “Smartform” you can find an “EDIT”-Button. The configurator will be opened in a seperate tab upon clicking it.



## 2.3 Systemrolle für die Verwaltung von TSS Ressourcen

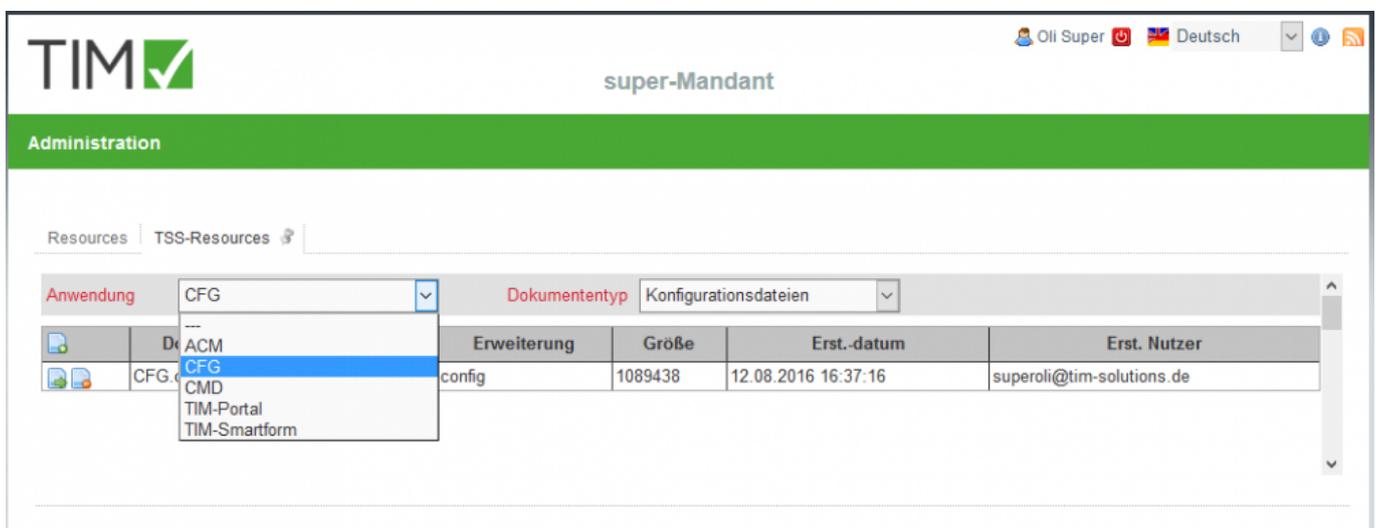
Every data-artifact which is used for the creation of a smartform is saved in a central database. An UI is integrated for the management of the artifacts was integrated in the admin client. A user needs the following role to manage the tss-resources (z.B- configurations, CSV-documents, CSS-documents etc. [TODO Verwaltung von TSS-Ressourcen](#) in his client or super-client:

- „resource-manager“ (If a user is a “resource-manager he doesn't necessarily need the role “admin”)

### 2.3.1 Global resources in the super-client

After the installation/update of the TSS the basic configurations (documenttype CONFIG-document) for the applications CMD and CFG have to be uploaded in the super-client.

- The configurations for ACM, TIM-portal and TIM-smartform are included in the product, but can be individualized.
- With the document-type CSS you can provide a CSS-document in order to implement your corporate identity (Colors, accent etc.) globally for the configurator(CFG) and the representation of the smartform (TIM Smartform)
  - The document will be automatically renamed into css.css upon uploading.



- With the document-type TODO?Logo-Dateien? you can upload a PNG-Document with your corporate logo, which will be automatically renamed into logo.png upon uploading and will be displayed in the top-left corner of the configurator.

### 2.3.2 client-specific resources

It is possible to have artifacts in your client, which are only available and effective in your client.

- For example a CSS-document which implements your corporate CI
- Documents with the types .csv, .txt, .xls, .png etc for integration in smartforms

The screenshot shows the TIM Smartform Designer interface. At the top left is the TIM logo. The user is logged in as Oliver Schirach, and the language is set to Deutsch. The main area displays a table of documents for the process 'tsstest'. A dropdown menu is open for the 'Dokumententyp' column, showing options like 'Konfigurationsdateien', 'Beschreibungen', 'CSS-Dateien', 'Daten-Dateien', 'Tabellendefinitionen', 'Dokumentationen', 'Hilfe-Dateien', 'Bild-Dateien', and 'Logo-Dateien'. The 'CSS-Dateien' option is selected. The table below shows a document named 'CSS.CSS' with a file icon, date name 'css', extension 'css', and size '1210'. The 'Erst. Nutzer' column shows 'oliver.schirach@tim-solutions.de'.

Prozessraum	Dokument	Dateiname	Erweiterung	Größe	Dokumententyp	Erst. Nutzer
tsstest	CSS.CSS	css	css	1210	CSS-Dateien	oliver.schirach@tim-solutions.de

## 3. TIM Smartform Designer

### 3.1 Target group & project-scope

- Professional users who want to create a simple smartform
- Predefined templates
- Intentionally limited choice of possible data-types.
- Creation and arrangement of attributes
- Only simple Smartform, which means without tab-pages, tables or accordions etc. Nur einfache Smartforms, d.h. ohne Tabpages, Tabellen, Accordions, etc.
  - For more complex smartforms you have to use the other versions: TODO [TIM Smartform Architect](#) and [TIM Smartform Expert](#).

### 3.2 Explanation of the UI and function - target group professional users!

#### 3.2.1 UI and operating philosophy

The smartform designer focuses on the smartform.

- It is based on a template and provides the opportunity to arrange attributes per drag and drop.
- With this function you can design you smartform compatible with your process.
- The TSS enables you to use a uniform operating philosophy in your smartforms.
  - Because of the uniform structures of the arrangements of elements, validations, and mandatory fields and editability etc. your end-users will quickly understand all your processes.



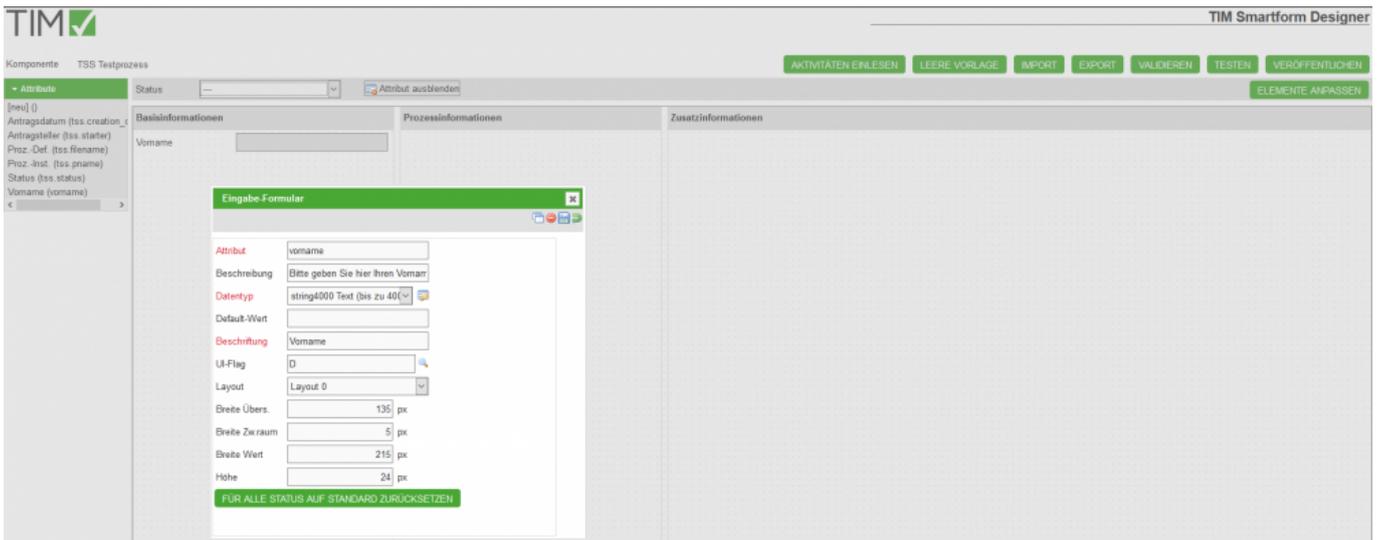
- With the button *Import activities* you can import the activities of your modeled process
- With the *Import* and *Export* button you can import configurations or export them as a backup or for further usage in other processes.
- With the *Validate* button you can validate your configuration.
- With the *Test* button you can test your smartform from user-perspective TODO:(For further details [3.2.3 Testmodus](#))
- With the button *Deploy* you can save your configurations. Everything you change must be saved by pressing the *Deploy* button, afterwards it can be seen in the ToDo client.
- If you refresh your browser without pressing the *Deploy* button your changes will be lost.
  - Before/after doing big changes it is strongly recommended to save your changes by pressing *Deploy*
- The changes on your smartform will be effective for the value "—" in the status-dropdown. In symbolizes the basic configuration.
- In the chapter TODO [3.2.4 Statusabhängige Konfiguration - Aktivitäten einlesen](#) it will be explained, how you can individualize your smartform for specific activities in the process.
- Attributes can be placed in any order, moved to another element or be hidden for a specific status via drag and drop
- For further details: TODO [3.2.2 Attribute](#)

### 3.2.2 Attribute

In the smartform Designer you can create attributes to gather and display process information.

- A new attribute can be created by drag&dropping the *[new] ()* tag into the smartform.
  - This is only possible if the status-dropdown is set to "—" . (For further details to status-specified configurations see also TODO [Statusabhängige Konfiguration - Aktivitäten einlesen](#))
- A new attribute will be saved as a TIM-process-variable.
- For further usage (e.g. in Word templates or Emails etc.) the attributes can be found in the process with `${name_of_the_attribute}`
- Eine spätere Auswertbarkeit im Dashboard zur KVP ist gegeben

- If a new attribute is dragged into the smartform or if an existing element is clicked, a pop-up will appear. There you can change the properties of the element:



Field	Description
Attribute	Identifier of the attribute. Can be referenced with $\${...}$ in emails. ("status" and "name" mustn't be used as attribute-identifier)
Decription	Appears as a tooltip as soon as the user places his mouse on the element
Data-Type	You can select a data-type or create an individual validation (see also <a href="#">TODODatentypen</a> )
Caption	Will be displayed as text in the left side of the element
UI-Flag	Definition of the editability of the element
Layout	Selection of a layout for the display of the attribute. (see also <a href="#">TODODatentypen</a> )
Width label	Width of the space where the description will appear
Width space	Distance between description and attribute
Width value	Width of the attribute
Height	Height of the attribute. Every height that is bigger than 24px will be displayed with the layout-option "big" (see also <a href="#">TODODatentypen</a> )
Button	Description
Reset all status to default	This button will appear, if the status-dropdown is set to "—". It will reset every status-specified versions of the attribute.
Reset actual status to default	This button will appear if the status-dropdown is set to something different then "—". By pressing it you will reset the configurations of this attribute for the selected status.
"Reset a selected attribute-property"-Icon	Every attribute-property that differs from the standart-configuration will be displayed in a bold text. Moreover a button will appear, that will reset the property upon pushing
Copy-Icon	Uses the attribute for the creation of a new one. The properties <i>Attribute</i> and <i>Label</i> will be reseted.
Löschen-Icon	Löscht das Attribut aus der Konfiguration. Dieses Icon ist nur im Status "—" verfügbar. Wenn Sie ein Attribut für einen Status nur ausblenden möchten, dann ziehen Sie es bitte per drag and drop auf die Fläche <i>Attribut ausblenden</i> neben der Statusdropdown. Sobald die Fläche rot wird, lassen Sie die linke Maustaste los. Das Attribut wird für den Status ausgeblendet, es verbleibt aber links in der Attributauswahlliste
Speichern-Icon	Speichert die getätigten Änderungen/ Neuanlage

Button	Description
Rückgängig-Icon	Setzt die getätigten Werte auf den Stand zurück, zu dem das Attribut das letzte mal gespeichert wurde

**Hinweis:** Details zum Statusabhängigen Konfigurieren siehe [Statusabhängige Konfiguration - Aktivitäten einlesen](#)

- Die Neuanlage, sowie alle getätigten Änderungen müssen durch einen Klick auf das Speichernsymbol abgeschlossen werden.
  - Das Popup kann danach geschlossen werden und das Attribut erscheint im gewünschten Element.
- Per drag and drop kann jedes Attribut beliebig in der Reihenfolge innerhalb eines Elementes angeordnet, oder in ein anderes Element verschoben werden
- Wenn ein Attribut ausgeblendet werden soll, dann muss es per drag and drop auf die Fläche *Attribut ausblenden* neben der Statusdropdown gezogen werden. Screenshot siehe [Statusabhängige Konfiguration - Aktivitäten einlesen](#)
  - Das Attribut bleibt auf diesem Weg in der Konfiguration erhalten, wird aber für den ausgewählten Status nicht mehr dargestellt

Die UI-Flags dienen zur Konfiguration der Editierbarkeit und können im Attribute-Popup entweder durch die Lupe in einem Popup ausgewählt, oder direkt eingegeben werden:

The screenshot shows a configuration window for an input form. The main window, titled 'Eingabe-Formular', contains the following fields:

- Attribut:** vorname
- Beschreibung:** Bitte geben Sie hier Ihren Vornam
- Datentyp:** string4000 Text (bis zu 4000)
- Default-Wert:** (empty)
- Beschriftung:** Vorname
- UI-Flag:** D
- Layout:** Layout 0
- Breite Übers.:** 135 px
- Breite Zw.raum:** 5 px
- Breite Wert:** 215 px
- Höhe:** 24 px

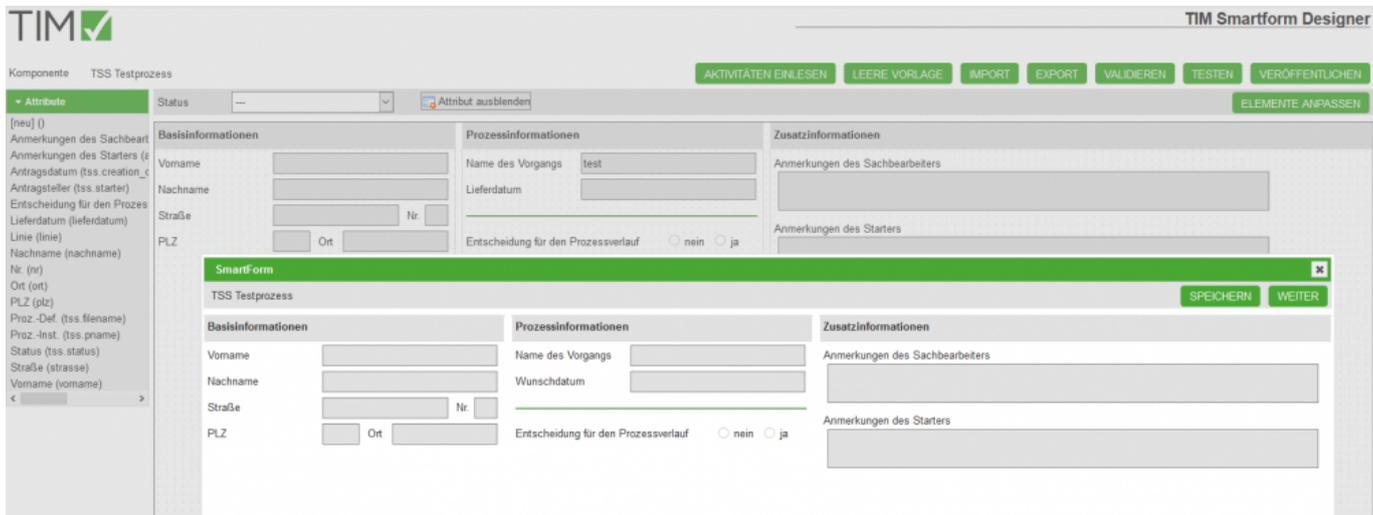
At the bottom of the configuration window is a green button: 'FÜR ALLE STATUS AUF STANDARD ZURÜCKSETZEN'.

The 'UI-Flag' popup shows a table with the following data:

Wert	Identifer	Beschreibung
<input type="checkbox"/>	M	Pflichtfeld
<input type="checkbox"/>	E	editierbar
<input checked="" type="checkbox"/>	D	schreibgeschützt
<input type="checkbox"/>	R	Refresh
<input type="checkbox"/>	h	Hintergrundtext
<input type="checkbox"/>	w	www
<input type="checkbox"/>	e	email-Link
<input type="checkbox"/>	r	aktualisieren

### 3.2.3 Testmodus

Der Testmodus dient dazu, das Smartform für einen ausgewählten Status zu testen, d.h. es öffnet sich nach Klick auf *TESTEN* ein Popup, in dem das Smartform voll funktional dargestellt wird. Dort können Sie prüfen, ob für den jeweiligen Status die konfigurierten Attribute sichtbar/ editierbar sind:



### 3.2.4 Statusabhängige Konfiguration - Aktivitäten einlesen

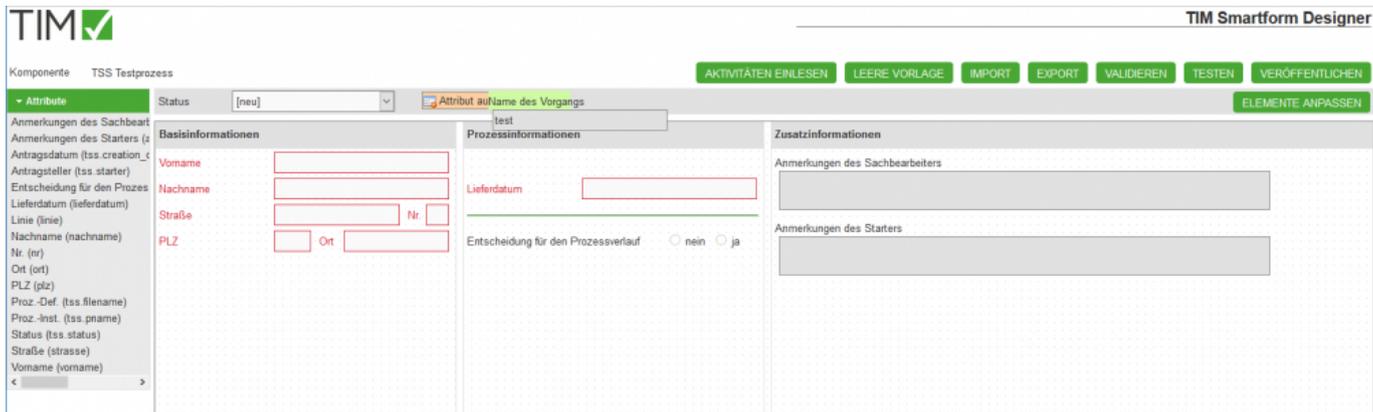
Nachdem Sie Ihr Smartform für den Status "—" konfiguriert haben, können Sie es nun an die Besonderheiten der Aktivitäten im Prozess anpassen. Hierzu klicken Sie bitte auf **AKTIVITÄTEN EINLESEN**. Die Dropdown füllt sich mit den Namen der Aktivitäten.

- Nun können Sie eine Aktivität, oder einen der Standard-Status [neu, Suche, in Arbeit, Prozess-Manager, beendet, Betrachter] auswählen und das Formular entsprechend anpassen
  - Einfügen bestehender Attribute
- Verschieben oder ausblenden von bereits angeordneten Attributen
- Modifikation der Attributeigenschaften durch einen Klick auf ein Attribut. Details siehe [3.2.2 Attribute](#)

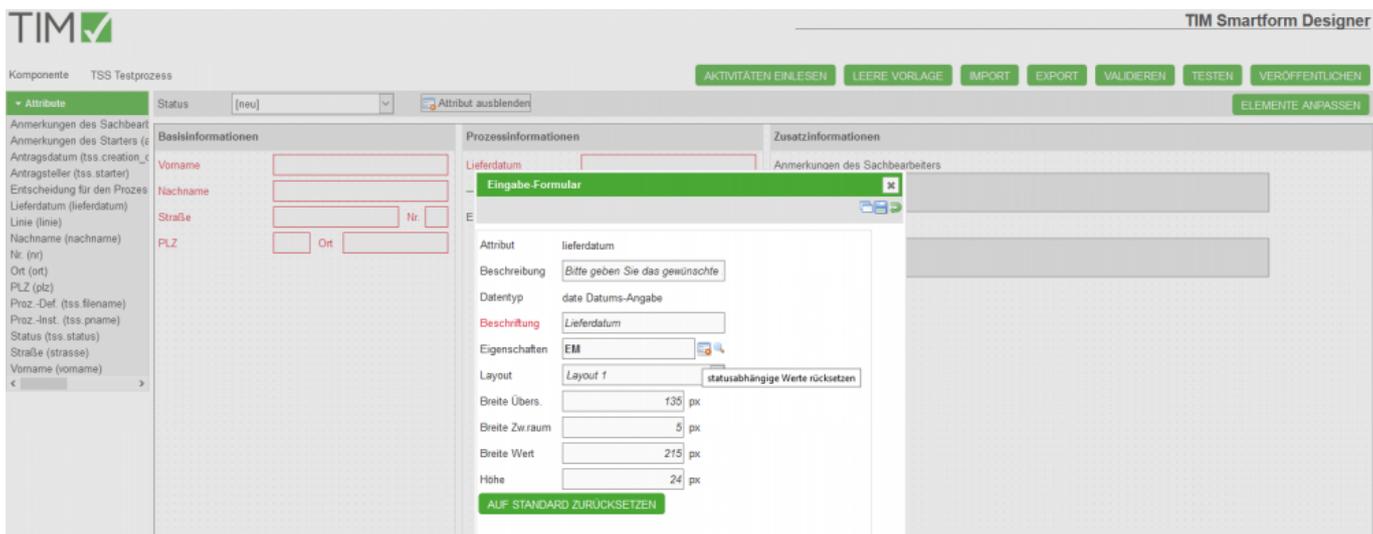
\* z.B. können Sie dadurch einstellen, dass ein Attribut für eine konkrete Aktivität ein Pflichtfeld wird

**Hinweis:** Wenn Sie ein Attribut nur für einen konkreten Status benötigen, dann müssen Sie es zuerst für den Status "—" anlegen, und dort auf *Attribut ausblenden* schieben. Wenn Sie nun den gewünschten Status auswählen, dann können Sie das Attribut links in der Attributliste auswählen und in das gewünschte Element schieben.





- Im Attribute-Popup werden statusspezifische Konfigurationen durch **fett** dargestellt. Konfigurationen, die sich aus dem Status "—" ergeben werden *kursiv* dargestellt

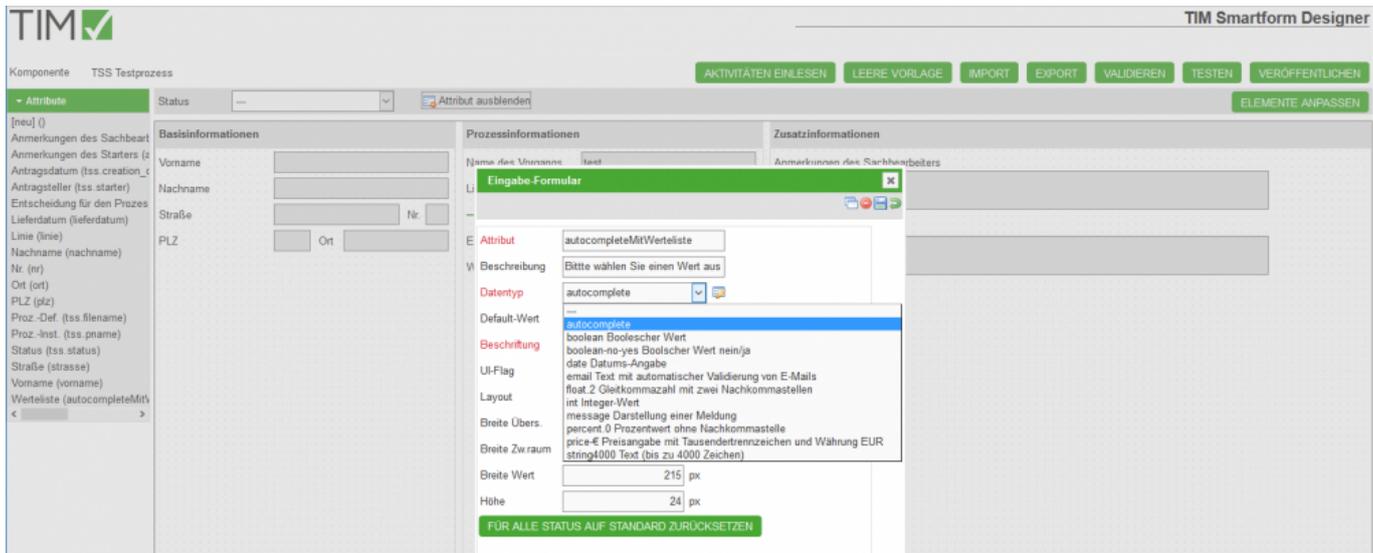


- Beispielhafte Anpassung des oben erstellten Smartforms für den Status *[neu]*, welcher zum Prozessstart gilt



### 3.2.5 Datentypen

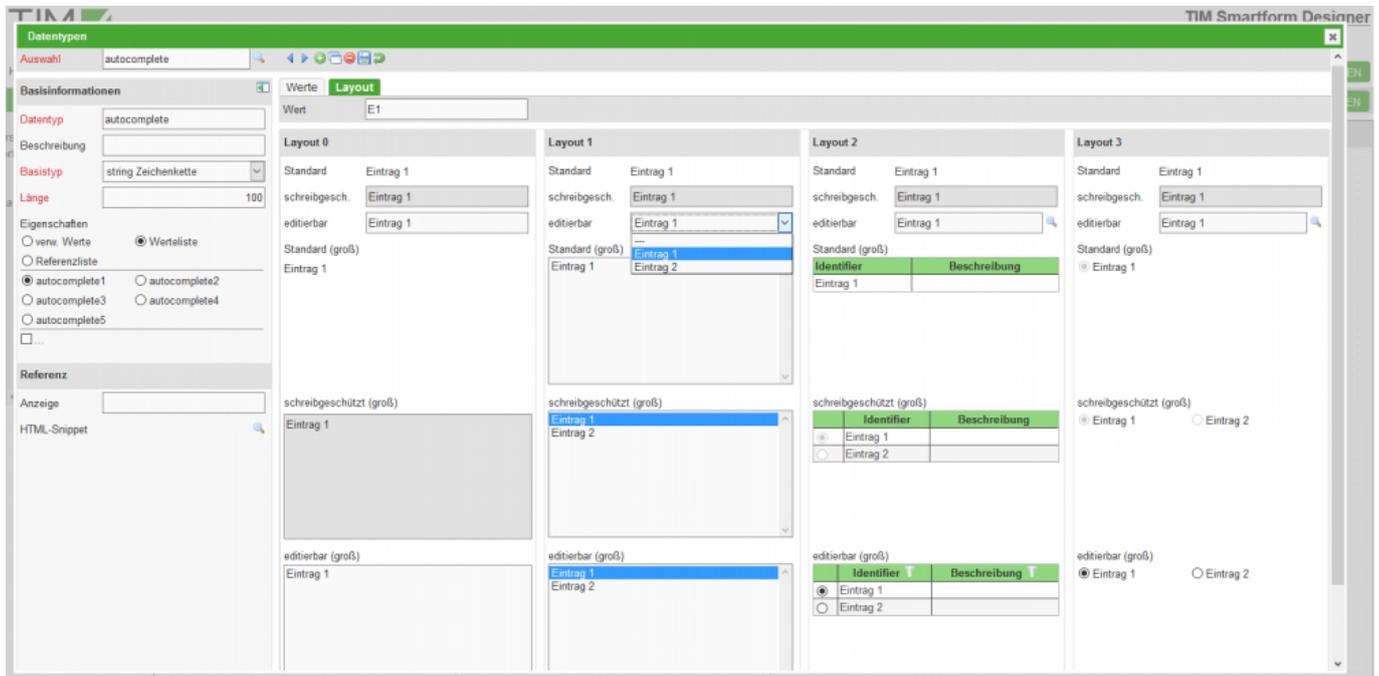
Im Attribute-Popup können Sie aus einer Liste vordefinierter Datentypen auswählen:



- Im Smartform Designer wurde diese Liste bewusst auf die in der folgenden Tabelle dargestellten Typen eingeschränkt

Datentyp	Beschreibung	Beispielhafte Verwendung
boolean	Boolescher Wert	Als Checkbox
decision-no-yes	Boolescher Wert nein/ja	Als Radio Button mit Ausprägung Ja/ Nein
date	Datums-Angabe	Als Textfeld mit Date Picker
email	Text mit automatischer Validierung von E-Mails	Als Textfeld
float.2	Gleitkommazahl mit zwei Nachkommastellen	Als Textfeld, welches den Input versucht, zu einer Zahl mit 2 Nachkommastellen umzuwandeln. Alle anderen Eingaben werden verweigert
int	Ganze Zahl	Als Textfeld, welches den Input versucht, zu einer Zahl umzuwandeln. Alle anderen Eingaben werden verweigert
message	Darstellung einer Meldung	Zur Darstellung von Hinweistexten
percent.0	Prozentwert ohne Nachkommastelle	Als Textfeld mit fester Einheit % und Validierung auf Zahlen
price-€	Preisangabe mit Tausendertrennzeichen und Währung EUR	Als Textfeld mit fester Einheit EUR und Validierung auf Zahlen mit 2 Nachkommastellen
string4000	Zeichenkette mit einer Maximallänge von 4000 Zeichen	Als ein- oder mehrzeiliges Textfeld

- Über einen Klick auf das Symbol rechts neben der Datentyp-Dropdownliste im Attribute-Popup können Sie durch die Layoutvarianten der Datentypen blättern



- zum anderen können Sie im Popup *Datentypen* einen eigenen Datentyp durch Klick auf das PLUS-Icon anlegen
  - Eigene Datentypen basieren dabei auf den vorhandenen Grunddatentypen
  - Durch die Angabe von Längenbeschränkungen oder Eingabemasken kann somit eine Validierung, z.B. auf Eingabe eines syntaktisch gültigen Buchungskreis XX\_098\_zt, konfiguriert werden
- Details zur Anlage von Datentypen siehe [6.4 Modul Datentypen](#)
  - Komplexe Datentypen z.B. für die Integration von CSV Listen als Autocomplete oder Dropdowns, etc. können im [TIM Smartform Architect](#) definiert werden
  - Angelegte Datentypen tauchen dann in der Dropdownliste *Datentyp* im Attribute-Popup mit auf und können durch den Designer verwendet werden

Anlegen von komplexeren Datentypen ist aber auch im Designer möglich:

- Mehrere zusammenhängende Radiobuttons:

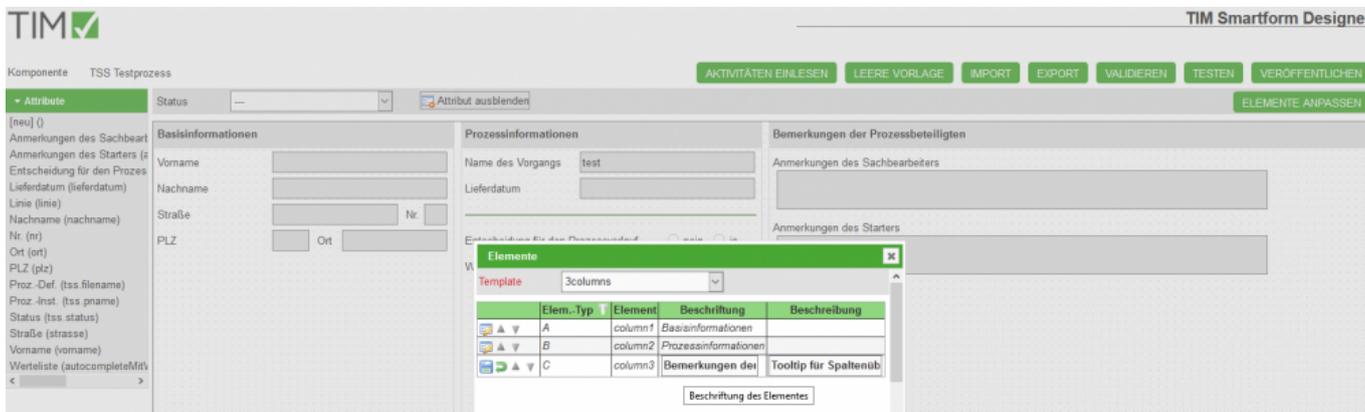
Dafür ist es notwendig einen neuen, eigenen, Datentypen zu erstellen, wie oben bereits beschrieben. Nachdem man auf das PLUS-Icon geklickt hat, muss man als Basistyp "String" auswählen, die rot hinterlegten Felder ausfüllen und bei "Eigenschaften" "Werteliste" auswählen. Der Wert "Länge" gibt hierbei an, wie viele Zeichen das Label (Text vor dem jeweiligen Radiobutton) eines Radiobuttons enthalten darf. Wenn man nun auf Speichern (Diskette) klickt, erscheinen 2 Reiter: "Werte" und "Layout". Sie müssen in dem Reiter "Werte" jeden gewünschten Radiobutton als einzelne Zeile hinterlegen, wobei "Wert" den Wert angibt, den der Radiobutton bei Auswahl senden wird. Nun müssen Sie alles speichern und das Pop-Up schließen. Im übrig bleibenden Pop-Up müssen Sie nun als Layout das "Layout 3" auswählen und für "Höhe" "25" oder mehr Pixel angeben.

### 3.2.6 Templates

Grundlegend basiert das Smartform auf einem Template, welches das grobe Layout vorgibt. Serienmäßig wird das Template *3columns* vorgegeben. Wenn Sie ein individuelles Template für Ihren Prozess benötigen, so kann dies im Smartform Architect oder Expert erfolgen. Hierzu öffnet ein Nutzer mit der Rolle *smartform-architect* oder *smartform-expert* Ihre Konfiguration und erstellt ein Template.

Anschließend können Sie das Template via *ELEMENTE ANPASSEN* auswählen und Ihre Attribute darin anordnen **Hinweis:** Der Smartform Architect richtet sich an Fachuser. Deswegen ist eine Pflege von komplexen Layoutelementen, wie z.B. Tabs und Platzhalter für die Darstellung von Tabellen, nicht im Designer pflegbar

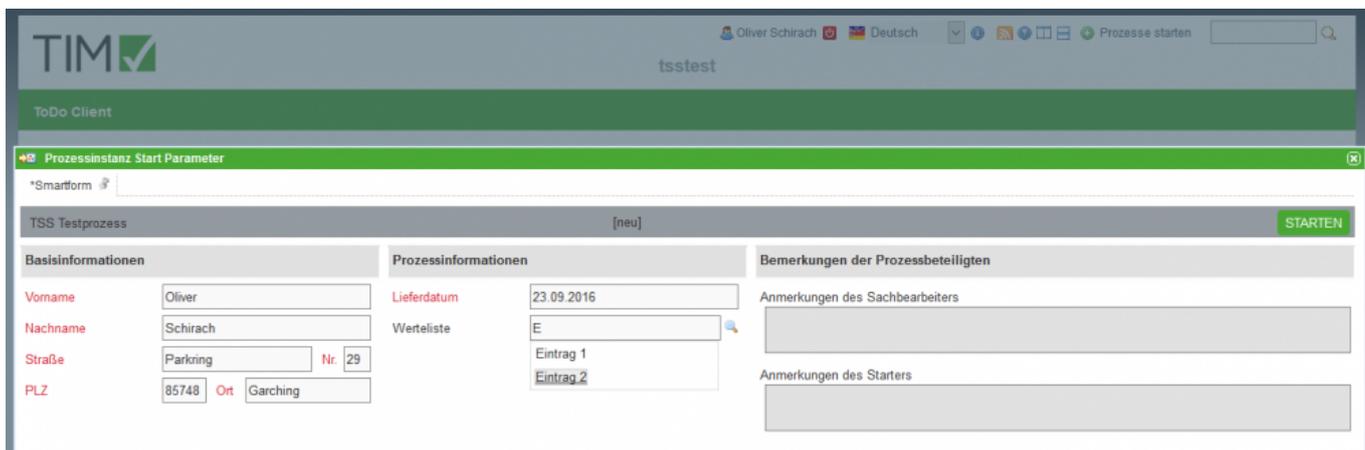
- Via *ELEMENTE ANPASSEN* können Sie zudem die Überschriften der Template-Elemente anpassen:



### 3.2.7 Ergebnis im ToDo Client

Der folgende Screenshot zeigt das konfigurierte Smartform einmal für den Prozessstart und einmal für die erste Aufgabe. Sie können sehen, dass teilweise Attribute ausgeblendet, oder rot als Pflichtfelder hervorgehoben sind:

- Smartform für den Start eines Vorgangs "TSS Testprozess":



- Smartform für die Erledigung der ersten Aufgabe "Erste Aufgabe im Prozess" Vorgangs "TSS Testprozess":

The screenshot shows a web form for a task in a process. The form is titled 'Aufgabe' and is part of the 'TSS Testprozess' for '- olsch - 20160001'. It is the first task in the process. The form is divided into three main sections: 'Basisinformationen', 'Prozessinformationen', and 'Bemerkungen der Prozessbeteiligten'. The 'Basisinformationen' section contains fields for 'Vorname' (Oliver), 'Nachname' (Schirach), 'Straße' (Parking), 'PLZ' (85748), and 'Ort' (Garching). The 'Prozessinformationen' section contains fields for 'Name des Vorgangs' (- olsch - 20160001), 'Lieferdatum' (23.09.2016), and 'Entscheidung für den Prozessverlauf' (radio buttons for 'nein' and 'ja'). The 'Bemerkungen der Prozessbeteiligten' section contains two text areas for 'Anmerkungen des Sachbearbeiters' and 'Anmerkungen des Starters'. There are also buttons for 'SPEICHERN' and 'WEITER'.

### 3.2.8 Config zum Selbertesten

Die Konfiguration, anhand der die Screenshots erstellt wurden, können Sie hier einsehen und für Ihre Tests verwenden: [Beispielkonfiguration](#)

## 4. TIM Smartform Architect

### 4.1 Zielgruppe & Projektscope

- IT Berater
- Erstellung komplexer Smartforms mit Datenstrukturen, Tabpages, etc.
- Integration von strukturierten Daten aus csv, txt, xls oder Datenbanktabellen
- Erstellung von Übersetzungen für mehrsprachige Smartforms in DE und EN
- Erstellung von Datentypen, Templates und Meldungen für einen Prozess
- Einfacher Zugang zu Groovy für showInterface zum modifizieren der Benutzeroberfläche für Sonderkonstellationen oder Feldabhängigkeiten
- ALLES, was für einen Prozess definiert wird, d.h. Templates, Datentypen, etc. ist nur innerhalb dieses Prozesses gültig und existent!

### 4.2 Erklärung der Oberfläche und Funktionen

- Überblick: Menü; Sidebar; WYSIWYG
- Details siehe [6. Module des TIM Smartform Architect und Expert](#)

## 5. TIM Smartform Expert

### 5.1 Zielgruppe & Projektscope

- IT Berater und Entwickler
- Tiefe Systemintegration
- Komplexe Konfigurationsmöglichkeiten
- Groovy extended. Auch zur Verwendung im Prozess via CBAGenericActionHandler möglich
- ALLES, was für einen Prozess definiert wird, d.h. Templates, Datentypen, etc. ist nur innerhalb dieses Prozesses gültig und existent!

## 5.2 Erklärung der Oberfläche und Funktionen

- Überblick: Menü; Sidebar; WYSIWYG
  - Details siehe [6. Module des TIM Smartform Architect und Expert](#)
- 

# 6. Module des TIM Smartform Architect und Expert

## 6.1 Modul Bedienoberfläche

- Eigenes Template erstellen:

Sie brauchen Architect- oder Expertrechte. Klicken Sie oben rechts auf den Reiter "Templates", anschließend auf das grüne "PLUS-Icon". Benennen Sie ihr neues Template in dem Feld "Template" und speichern Sie alles (Diskette). Anschließend gehen Sie auf den Reiter Elemente und klicken auf das grüne PLUS-Icon am rechten Bildschirmrand. Benennen Sie ihr Element. (Folgendes ist nicht der einzig richtige Weg sondern nur ein Vorschlag wie man ein blankes Template in passender Größe designen kann). Wählen Sie als Position 1 aus, falls Sie noch keine anderen Elemente eingefügt haben, geben Sie als Element-Typ "A-Element" und als "UIE" "uie-details Formular-Element" an. In dem Abschnitt "Positionierung" können Sie die gewünschte Größe sowie den Abstand zu anderen Elementen in Pixeln angeben. Anschließend drücken Sie auf Speichern (Diskette) und auf "Veröffentlichen".

## 6.2 Modul Templates

## 6.3 Modul Tabellen

- dau- immer entsprechend des Dateityps, oder Datenbankzugriff
- Bei Ablage als Prozessvariable (BL-Flag P) immer dau-tim verwenden. Achtung: Max. 4000 Zeichen in einer Prozessvariable erlaubt. Dies schließt Steuerzeichen der Tabelle mit ein. Bitte nur zum fixen Testen verwenden

## 6.4 Modul Datentypen

Auflistung aller Basistypen und Details dazu

Reguläre Ausdrücke

- Stellen Pattern erlaubter Zeichenketten dar.
- Ist im Feld für den regulären Ausdruck ein Pattern eingegeben, ist der Attributwert nur erlaubt, wenn er dem Pattern entspricht.
- Syntax von regulären Ausdrücken ist nicht einfach – bitte Literatur verwenden, z.B. <http://www.regular-expressions.info/quickstart.html>.

Masken

- Geben dem Nutzer definierte Pattern zur Eingabe vor und lehnen Falscheingaben ab:
  - 0 .. Vorzeichen und Ziffern
  - 9 .. nur Ziffern
  - a .. Buchstaben (A-Z,a-z) sowie '-' und '\_'
  - z .. nur Buchstaben (A-Z,a-z)
  - \* .. Buchstaben (A-Z,a-z) und Ziffern sowie '-' und '\_'

## 6.5 Modul Meldungen

## 6.6 Modul Übersetzung

## 6.7 Modul funktionale Struktur

## 6.8 Verwaltung von TSS-Ressourcen

---

# 7. praktische Tipps für die gängigsten Smartformanforderungen

## 7.1 Attribute in Taskliste darstellen

- Attribute, die in der Taskliste angezeigt werden sollen, sind sog. Indexfelder
- Es gibt 10 Felder, die mit beliebigen Attributen belegt werden können
- Modul Tabellen - tim\_process auswählen - Tab Attribute - Attribut auswählen - rechts in der Sidebar unter Geschäftslogik das BL-Flag auf 1 oder 2 ... oder 0 setzen - speichern - Veröffentlichen - Echten Prozess zum Testen starten
- Die Spaltenüberschriften können mittels dem folgenden Eintrag in der custom.js für jeden Mandanten angepasst werden:
  - `gadget.languages.de.standard_smartformindex1= 'Überschrift für Index 1 in DE';`
  - `gadget.languages.en.standard_smartformindex2= 'Überschrift für Index 1 in EN';`

## 7.2 Besondere Eigenschaften von Attributen

- Zusätzliche Flags zu "E" oder "D" für:
  - Mail (e): Es wird automatisch rechts neben dem Eingabefeld ein kleines Icon gerendert. Wenn der Nutzer auf das Icon klickt, dann öffnet sich das Systemstandardprogramm für den Versand einer Mail. Die Empfängeradresse wird dabei vorbelegt
  - HTML-Link (w): Es wird automatisch rechts neben dem Eingabefeld ein kleines Icon gerendert. Wenn der Nutzer auf das Icon klickt, dann öffnet sich der Systemstandardbrowser und die eingegebene Webseite wird geladen.

- Zusätzliche Flags zu "E":
  - Placeholder (h): Der eingegebene Text aus dem Feld Beschreibung wird nicht nur als Tooltip verwendet, sondern auch als Hintergrundtext in editierbaren einzeiligen Textfeldern. Sobald der Nutzer etwas eintippt, verschwindet der Hintergrundtext
  - refresh (r): Dies führt dazu, dass nach Eingabe durch einen Nutzer in das Feld, das Smartform neu geladen wird. Alle Daten bleiben erhalten. Dadurch wird aber die Groovy Methode showInterface erneut ausgeführt. Dort abgelegte Funktionen können nun basierend auf der Eingabe des Nutzers z.B. Attribute ein-/ ausblenden oder Pflichtfeldbedingungen individualisieren
  - Werden diese beiden Flags in Kombination mit "D" und/oder ohne "E" genutzt, dann haben sie in dem Moment keine Wirkung
- Umbruch Labelbreite zu Feldbreite:
  - Per default wird ein Attribut immer wie folgt gerendert: Label + Zwischenabstand + Eingabefeld
  - Wenn nun das Eingabefeld einmal unter dem Label erscheinen soll, dann muss die Breite des Labels, d.h. Breite Übers., einfach breiter eingestellt werden. Wenn nun Breite Übers. + Breite Zw.raum + Breite Wert > Breite des Elements, in dem das Attribut platziert wird, dann bricht das Eingabefeld automatisch nach unten um

### 7.3 Einbinden einer csv-Datei für ein Autocomplete-/Dropdownfeld

- Zuerst muss im Modul Tabellen eine Tabelle angelegt werden, die genau so heißt, wie die vorliegende csv, txt oder xls
  - Die Tabelle bekommt die entsprechend ihrer Endung passende DAU zugeordnet.
  - Die BLU muss immer auf blu-basic gesetzt werden
  - Nun müssen in dieser Tabelle alle Spalten, die in der zugrundeliegenden Datei existieren, als Attribute angelegt werden
    - Dies ist analog zu den Attributen im Smartform auszufüllen
    - Wichtig: DA-Flag immer auf "SIU" setzen
    - Indizes: Hier geben Sie an, welche Spalten zur eindeutigen Referenz (= Index 1) eines Datensatzes zusammengezogen werden müssen
    - Ref.-Typ:
      - Hier entscheiden Sie, welches Attribut mit "D" automatisch als Bezeichnung, z.B. später in einer Dropdown verwendet wird
      - Hier entscheiden Sie, welches Attribut mit "I" automatisch als fachlicher Kurzidentifizier verwendet wird
      - Hier entscheiden Sie, welches Attribut mit "V" automatisch verwendet wird, wenn ein ausgewählter Eintrag gespeichert wird
      - Details siehe unten (Exemplarisches Ergebnis eines Dropdowneintrags im HTML: <option value="Attribut mit V">"Attribut mit I" "Attribut mit D"</option>)
- Nun muss die Datei via Tab Ressources im Admin Client hochgeladen werden. Hierbei ist der Datentyp Datendatei auszuwählen
  - Nach dem Upload bitte den Button Cache löschen klicken, damit die neueste hochgeladene Version verwendet wird
- Datentyp für Dropdown
  - Damit die Tabelle nun als Grundlage für eine Dropdown oder ein Autocompletefeld verwendet werden kann, muss hierfür ein datentyp angelegt werden

- Basistyp: = Datentyp der Spalte aus der CSV, die als eindeutige Referenz dient
  - Eigenschaften: Referenzliste
  - Tabelle: vorher angelegte Tabelle
  - Attribut: Das Attribut aus der Tabelle, das in der DB gespeichert werden soll. Es muss eindeutig sein. Entspricht dem Attribut mit dem "V"
  - Filter: optional - hier kann mit der Syntax ATTRIBUT= die Ergebnismenge der Tabelle auf ein Attribut eingeschränkt werden. Mittels ATTRIBUT={ATTRIBUT\_IN\_TIM\_PROCESS} kann auch auf den Inhalt eines anderen Attributs gefiltert werden, um somit eine Filterung zu ermöglichen, die vom Inhalt eines anderen Attributs abhängig ist. WICHTIG: Das Attribut, wonach gefiltert werden soll, muss das Flag "r" haben, damit bei einem Wechsel des Wertes auch die Tabelle mit dem neuen Filter geladen wird
  - Anzeige:
    - leer lassen, dann zieht der Default, bestehend aus I und D
    - oder mit {Attribut} die Attribute auswählen, die angezeigt werden sollen. z.B.: {nachname}, {vorname} ergibt: "Schirach, Oliver"
- Alternativ kann die Tabelle auch als Grid eingebunden werden, um Datensätze tabellarisch darzustellen oder bearbeitbar zu machen

## 7.4 Erstellung von Tabpages

- Im Smartform muss ein Element mit der uie-tabpages-interfaces eingebunden werden.
  - Soll das erste enthaltene Tab direkt geladen werden, dann muss unter Eigenschaften das Flag "t" ausgewählt werden
- Nun muss für jedes zu platzierende Tab ein Template erstellt werden
  - Achtung: Bitte vermeiden Sie eine Überschneidung der verwendeten Elementbuchstaben, da sonst beim drag&drop die platzierten Attribute in allen Tabs auftauchen, in denen der selbe Buchstabe existiert
  - Sollte eine Überschneidung von Buchstaben unumgänglich sein, so dürfen die Attribute, die in den betroffenen Buchstaben nur in einem Tab auftauchen, nicht per drag&drop platziert werden
  - Die Attribute müssten dann via Bedienoberflächen - Attribute in der tabellarischen Ansicht den jeweiligen Buchstaben zugeordnet werden. Achten Sie hierbei darauf, dass Sie sich im richtigen Tab befinden. Wenn in der Tabelle ein Elementbuchstabe eingegeben wird, dann erscheint das Attribut exakt nur in dem gewünschten Tab.
- Zur Kombination des Tabs mit dem Element im Smartform, muss aus dem Accordion in Bedienoberfläche - Editor der Eintrag "[neu] ()" aus dem Untermenü "Inhalte" auf das Element mit der uie-tabpages-interfaces gezogen werden
  - Hierbei erscheint links eine leere Sidebar, in der das einzufügende Tab konfiguriert wird:
    - Bedienoberfläche: technischer Name für das Tab
    - Beschriftung: im Smartform verwendeter Bezeichner für das Tab
    - übergeordnet: Smartform
    - Template: Das zu verwendende Template für das Tab auswählen
    - Menü / Inhalt: Das Element mit der uie-tabpages-interfaces auswählen
    - UIU: Wenn der Tab auch einfach nur Attribute enthält, die als Prozessvariablen in tim\_process abgebildet werden, dann: uiu-selection
      - Wenn das Tab auf eine Tabelle zeigt, die mit dem Prozess in Verbindung steht, dann uiu-details
    - Eigenschaften: E, damit der Tab generell editierbar ist

- zusätzlich n und a, wenn der Tab auf eine Tabelle zeigt, die mit dem Prozess in Verbindung steht und die darin befindlichen Attribute unmittelbar im Tab angeordnet werden. Sobald eine Grid in dem Tab verwendet wird, so sind n und a nicht zu verwenden
- Icons: keine notwendig. Sie kommen zum Tragen wenn über diesen Weg ein Grid zur tabellarischen Ansicht/ Bearbeitung von Datensätzen eingebunden wird
- Tabelle: tim\_process, wenn einfach nur Prozessvariablen dargestellt werden sollen, oder eine andere fachliche Tabelle, wie gewünscht
- Filter: Einschränkung einer fachlichen Tabelle auf gewisse Einträge. So könnten z.B. aus einer Stammdatentabelle Informationen angezeigt werden, die entsprechend nach einem oder mehreren Attributen gefiltert sind

## 7.5 Integration einer fachlichen Tabelle als Grid

- Achtung: verwendetes Template grid-Template unter Bedienoberfläche - Elemente - G - auf uie-grid-edit umstellen, sonst tauchen die Icons nicht auf

## 7.6 Erstellung eines Accordions

- Subelemente

## 7.7 Erstellung eines Groovy Codes, der beim Laden des Smartforms gezogen werden soll

- Im Modul Bedienoberfläche gibt es für jede Bedienoberfläche links in der Sidebar den Button "showInterface"
  - Dort kann Groovy Code hinterlegt werden, der im Moment des Seitenaufbaus ausgeführt wird
- Mit dem Flag r kann das Neuladen des Smartforms nach Eingabe eines Wertes in ein Feld aktiviert werden
  - Dadurch wird das Smartform neu geladen und showInterface erneut ausgeführt
  - Ebenso werden dynamische Filter in Datentypen oder Tabellen, die auf andere Attribute zugreifen, neu ausgeführt
  - Eingegebene Daten bleiben bei diesem Neuladen erhalten
- Beispiele:
  - Auslesen der Benutzersprache und setzen in das Attribut "sprache":

```
def locale = FRAME.getProperty("locale_id").toUpperCase();
if (locale != null && locale.length() > 1 && locale.substring(0,2) !=
"DE") {
    DATA_ROW.setValue("mm_sprache", "EN");
}
```

- Setzen des UI-Flags für ein Attribut:

```
ICbaRow attribut_xxxx = FRAME.searchRow("dd_intattr", "interf_recid="
+ CONFIG_ROW.getRecid() + "&attr_id=TECHNISCHER_ATTRIBUTNAME");
attribut_xxxx.setValue("ui_flag", "D");
```

- Lesen eines Wertes für ein Attribut:

```
String wert_des_attributs_im_moment_des_seitenaufbaus =  
DATA_ROW.getValue("TECHNISCHER_ATTRIBUTNAME");
```

- Setzen eines Wertes für ein Attribut:

```
DATA_ROW.setValue("TECHNISCHER_ATTRIBUTNAME",  
"neuer_Wert_für_das_Attribut");
```

## 7.8 Konfiguration von statusabhängigen Attributeigenschaften

- Zurücksetzen einzelner Attributvarianten auf Standard
- „Für alle Status auf Standard zurücksetzen“ bzw. „Auf Standard zurücksetzen“ Schaltflächen
- Statusübersicht. Kopieren, Löschen, \* **Schritt, der entfallen ist** \*

## 8. Installation

- siehe [Installation](#)

## 9. Altes CBA Wiki

- [16.0 Übersicht](#)
- [16.1 Einführung](#)
- [16.2 Tabellen](#)
- [16.3 Datentypen](#)
- [16.4 Templates](#)
- [16.5 Bedienoberflächen](#)
- [16.6 Meldungen](#)
- [16.7 TIM-Anbindung](#)

From:  
<https://wiki.tim-solutions.de/> - **TIM Wiki** / [NEW TIM 6 Documentation](#)

Permanent link:  
<https://wiki.tim-solutions.de/doku.php?id=en:software:tss:start&rev=1483115674>

Last update: **2021/07/01 09:55**

