

ELO **Workflowformulareditor**

<Stand : 08.02.2011 Version : 8.00.000>

Auf den folgenden Seiten wird die Verwendung des ELO Workflowformulareditors beschrieben.
Der Anwender soll danach in der Lage sein, eigene Workflowformulare zu erstellen und zu verwenden.

Inhalt

1	Vorbereitung	3
1.1	Welche Daten werden benötigt	3
1.2	Wo werden die Daten gespeichert	3
1.3	Wie gehören die Daten zusammen	4
1.4	Wie soll das Formular aussehen	5
2	Erstellung des Formulars	7
3	Formular einbinden in einen Workflow.....	13
4	Weitere Optimierung.....	18
4.1	Erstellung von Zwischenüberschriften.....	18
4.2	Motiv farblich hervorheben.....	20
4.3	Schreibschutz für Teilformulare.....	21
5	Mehrfacheingaben.....	25
5.1	Interessentenverwaltung im Bildarchiv.....	25
5.2	Listenlänge begrenzen	27
5.3	Gruppen duplizieren	28
5.4	Automatische Positionszähler	29

6	Markierungsfelder und Auswahlfelder	32
6.1	Radiobuttons (Auswahl)	32
6.2	Checkboxen (Markierung)	33
6.3	Schreibschutz	33
7	Die komplette Werkzeugleiste	34
7.1	Verfügbare Eingabeelemente	34
7.2	Zellen verbinden und trennen	35
8	Validierung	37
9	Eigene Skripte und Styles	39
9.1	Nützliche Hilfsfunktionen	40
9.2	Eigene Styles	42
9.3	Eigene Bilder	42
9.4	Eigene Web-Seiten, Skript- oder Style-Dateien	42
10	ELOWf Struktur im Archiv	43
10.1	Einträge in der config.xml Datei	43
10.2	Struktur im Archiv	44

1 Vorbereitung

Wenn man ein gutes Formular erstellen möchte, kann man nicht einfach am Computer anfangen zu zeichnen und Eingabefelder beliebig hintereinander stellen. Es sind ein paar Vorbereitungen notwendig, die dann aber zu einem besseren Ergebnis führen.

Folgende Schritte sollten ausgeführt werden und die offenen Fragen beantwortet werden:

1. Welche Daten werden benötigt?
2. Wo und wie sollen Sie gespeichert werden? Welche Eingabehilfen sollen verwendet werden (z.B. ELO Stichwortlisten))
3. Wie gehören die Daten logisch zusammen? Wer bearbeitet wann welche Felder?
4. Wie sollte das Formular aussehen?

Im Folgenden soll ein einfaches Beispiel zur Erfassung von Bilderdaten einer Fotodatenbank vorgestellt werden. Dazu wird dann ein zweistufiger Workflow vorbereitet, der die Erfassung der technischen Daten von der Motivbeurteilung trennt.

1.1 Welche Daten werden benötigt

Hier sammelt man erst mal alle Daten, die zu einem Dokument benötigt werden. Dabei ist ein wenig Fingerspitzengefühl gefragt. Wenn man unsinnig viele Daten erfasst, fühlen sich die Anwender gegängelt und werden die Masken vielleicht unvollständig ausfüllen. Wenn man wichtige Daten vergisst, fehlen diese später im Prozess. Deshalb sollte man sich für diesen scheinbar trivialen Punkt etwas Zeit nehmen.

Für das Foto Beispiel werden folgende Daten benötigt (ohne spezielle Reihenfolge):

- Eine Motivbeschreibung.
- Das Kameramodell.
- Die Belichtung.
- Der ISO-Wert.
- Bemerkungen zum Foto.

1.2 Wo werden die Daten gespeichert

In vielen Fällen werden die Workflows gestaltet nachdem bereits Dokumente hierzu im Archiv vorliegen. Dann ist die Verschlagwortungsmaske bereits vorbestimmt und sie muss bei Bedarf um die zusätzlichen Felder ergänzt werden. Andernfalls kann eine neue Verschlagwortungsmaske angelegt werden.

Im aktuellen Beispiel passen die benötigten Daten perfekt zu der bereits vorhandenen Maske „Bild“ aus dem Demo Archiv. Falls Sie diese noch nicht haben, können Sie sie nun anlegen.

Nr	Index Feld	Gruppe	Min	Max	Typ	Zugriffsart
1	Kamera	BIMODELL	0	0	Txt	Normal
2	ISO-Wert	BIISO	0	255	Txt	Normal
3	Belichtung	BILICHT	0	255	Txt	Normal
4	Motiv	BIMOTIV	0	255	Txt	Normal
5	Bemerkung	BIBEM	0	255	Txt	Normal

Da einige Felder, wie z.B. Kamera oder ISO-Wert, häufig mit gleichartigen Daten gefüllt werden, sollten hierfür Stichwortlisten angelegt werden. Diese erleichtern dem Mitarbeiter später die Eingabe. Zudem können Sie zu gewissen Standards bei der Eingabe führen (z.B. durch vordefinierte Begriffe für das Motiv), die dann später die Suche verbessern.



1.3 Wie gehören die Daten zusammen

Ein gutes Formular listet die Eingabefelder nicht einfach in einer willkürlichen Reihenfolge auf. Auch eine alphabetische Sortierung ist im Normalfall nicht angemessen. Das führt dazu, dass jeder Anwender die Liste nach den Eingaben durchsuchen muss, die für ihn relevant sind.

Bei der Strukturierung der Liste sollte man zwei Punkte berücksichtigen:

1. Wer gibt die Daten ein? Das bestimmt die Hauptgruppierung.
2. Wie gehören die Daten einer Gruppe logisch zusammen.

Im Foto Beispiel gibt es zwei Personengruppen, die die Daten erfassen. Zuerst werden vom Fotografen die technischen Daten eingegeben. Später wird vom „Art Director“ eine künstlerische Bewertung vorgenommen.

Zu den technischen Daten gehören das Kameramodell, der ISO-Wert und die Belichtung. Das Motiv und die Bemerkung werden den künstlerischen Daten zugeordnet.

In der Gruppe der technischen Daten wird nun die Kamera ganz nach vorne gestellt, da das eine Basisinformation ist. Anschließend folgt die Belichtung. Als letztes kommt der ISO-Wert, der genau genommen ja eine Detailinformation zur Belichtung ist.

Bei den künstlerischen Daten ergibt sich ein ähnliches Bild: Als Basisinformation zu jedem Foto dient das Motiv, zur weiteren Beurteilung wird dann das Bemerkungsfeld heran gezogen.

Hinweis: falls eine Reihe von Eingabefeldern gleichrangig sind, sollte man sie nach dem Typ und der Größe der Eingabefelder gruppieren. Hiermit erreicht man eine gleichmäßige Eingabe und ein übersichtlicheres Design. Wenn sich drei Textfelder mit drei Datumsfeldern abwechseln, dann gibt das ein sehr unruhiges Aussehen. Besser ist es, wenn erst die drei Datumsfelder und danach die drei Textfelder (möglichst mit gleicher Eingabegröße) angeordnet werden.

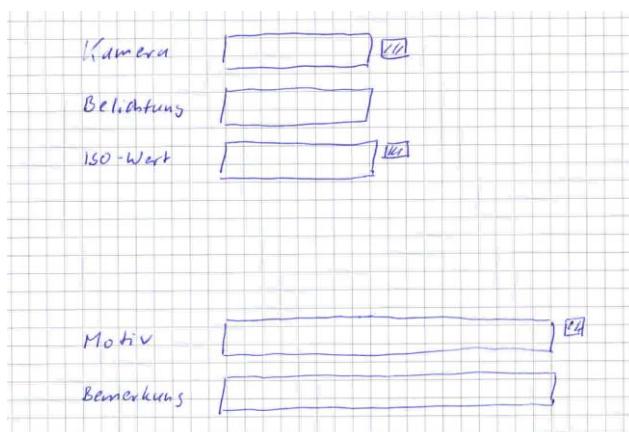
Nicht gut		Besser	
Ablagedatum	<input type="text"/>	Ablagedatum	<input type="text"/>
Bedarf	<input type="text"/>	Druckdatum	<input type="text"/>
Druckdatum	<input type="text"/>	Rückgabedatum	<input type="text"/>
Kostenstelle	<input type="text"/>	Bedarf	<input type="text"/>
Rückgabedatum	<input type="text"/>	Kostenstelle	<input type="text"/>
Zertifikat	<input type="text"/>	Zertifikat	<input type="text"/>

1.4 Wie soll das Formular aussehen

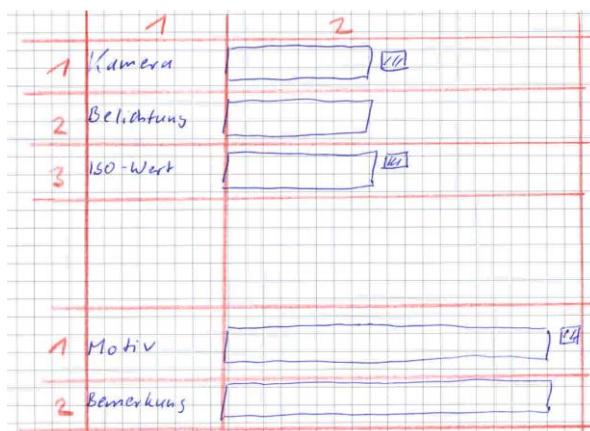
Wenn man nun festgelegt hat, welche Daten erfasst werden sollen und wie diese in Teilgruppen zerlegt und angeordnet werden sollen, geht es an die Gestaltung der Eingabeformulare. Hierbei sollte man sich nicht sofort an den Computer setzen und mal sehen, was so entsteht. Denn dabei besteht das Risiko, dass man sich zu früh in Details verliert (welche Indexzeile, Stichwortliste, Validierung?) und dabei das „Große Ganze“ nicht im Blick behält. Zudem neigt man dazu, „suboptimale“ Lösungen beizubehalten, auch wenn sich während der Gestaltung heraus stellt, dass man es besser machen könnte („Jetzt habe ich mir schon die ganze Arbeit gemacht...“). Letztendlich glaube ich auch, dass man im Umgang mit Bleistift und Papier kreativer entwirft.

In einem ersten Schritt sollte also eine einfache Papierskizze anfertigen. Dieser Schritt wird oft aus mehreren Iterationen bestehen, da man in der Entstehungsphase eine unübersichtliche oder „unschöne“ Anordnung erkennen und verbessern kann.

In dem einfachen Beispiel kommt man mit einem Entwurf aus.



„Ordentliche“ Formulare verteilen ihre Eingabefelder nicht wild über die Fläche sondern verwenden eine Tabellenstruktur. Diese führt zu klaren Linien und einem übersichtlichen Erscheinungsbild. Deshalb verwendet der ELO Formulareditor auch Tabellen. Im nächsten Schritt wird also bestimmt, wie viele Spalten und Zeilen benötigt werden. Wenn man die Vorlage gut erstellt hat, ist das schnell mit einem Buntstift und einem Lineal durchgeführt.



Alle Formularteile sollten eine gleiche Anzahl von Spalten haben (wir werden später sehen, dass man in Teilbereichen bei Bedarf auch Spalten zusammenfassen kann). Im Beispiel sind es nur zwei Spalten, eine für den beschreibenden Text, eine für die Eingabefelder. Der Formularteil für die technischen Daten besteht aus drei Zeilen, die künstlerische Beurteilung besitzt zwei Zeilen.

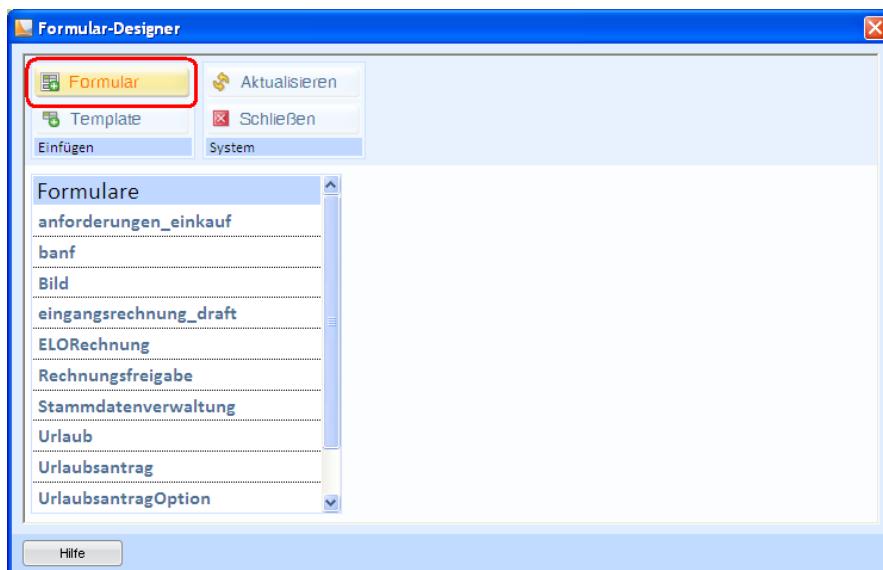
Die vorbereitenden Schritte sind nun abgeschlossen. Als nächstes geht es dann in den Formulareditor.

2 Erstellung des Formulars

Ein ELO Workflowformular kann aus mehreren Teilen bestehen. Der Workflow Administrator hat dann die Möglichkeit, bei jedem Workflowschritt zu bestimmen, welche Formularteile ein Anwender sieht oder nicht sieht. Bei einem Rechnungsworkflow ist der scannenden Poststelle völlig egal, welche Buchhaltungsdaten später eingegeben werden. Auch wenn man diese Felder in der Poststelle ausgrauen und auf ReadOnly setzen kann, führt es doch zu einem unnötig umfangreichen Formular. Besser ist es, wenn die Poststelle in ihrem Eingabeformular nur die Felder sieht, die für sie relevant sind.

Die Möglichkeit zur Aufteilung des Formulars wollen wir auch in dem Foto Beispiel nutzen. Der erfassende Fotograf sieht nur sein Teilformular und kommt so gar nicht erst in die Versuchung, eine künstlerische Bewertung vornehmen zu wollen (was man ja gut verstehen könnte). Der „Art Director“ soll beide Formularteile sehen.

In der Formularliste wird nun zuerst ein neues Formular „Bildworkflow“ angelegt.



Neues Formular anlegen



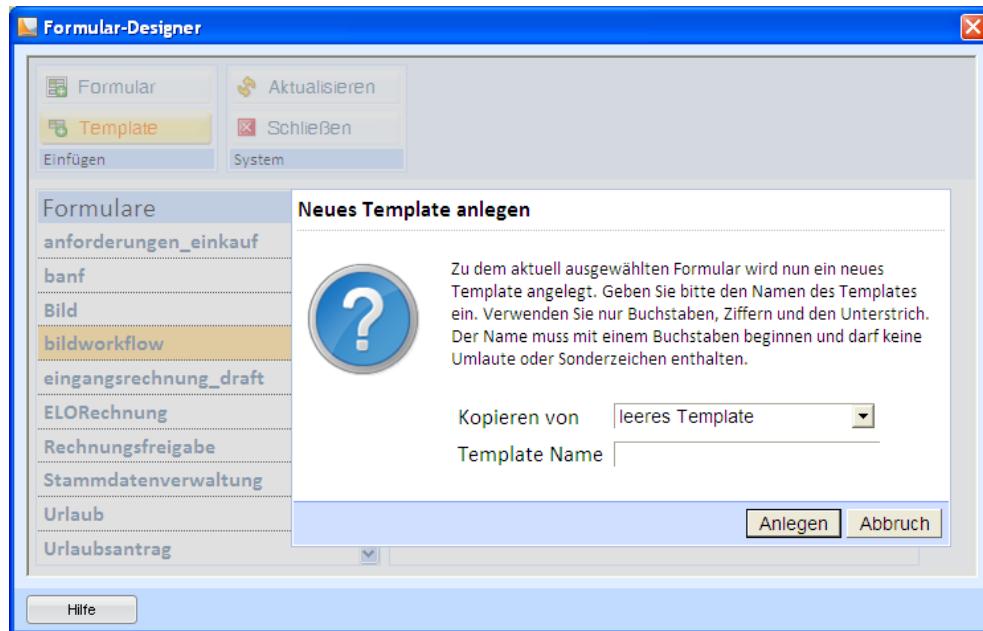
Geben Sie bitte den Namen des neuen Formulars ein. Verwenden Sie nur Buchstaben, Ziffern und den Unterstrich. Der Name muss mit einem Buchstaben beginnen und darf keine Umlaute oder Sonderzeichen enthalten.

Formularname:

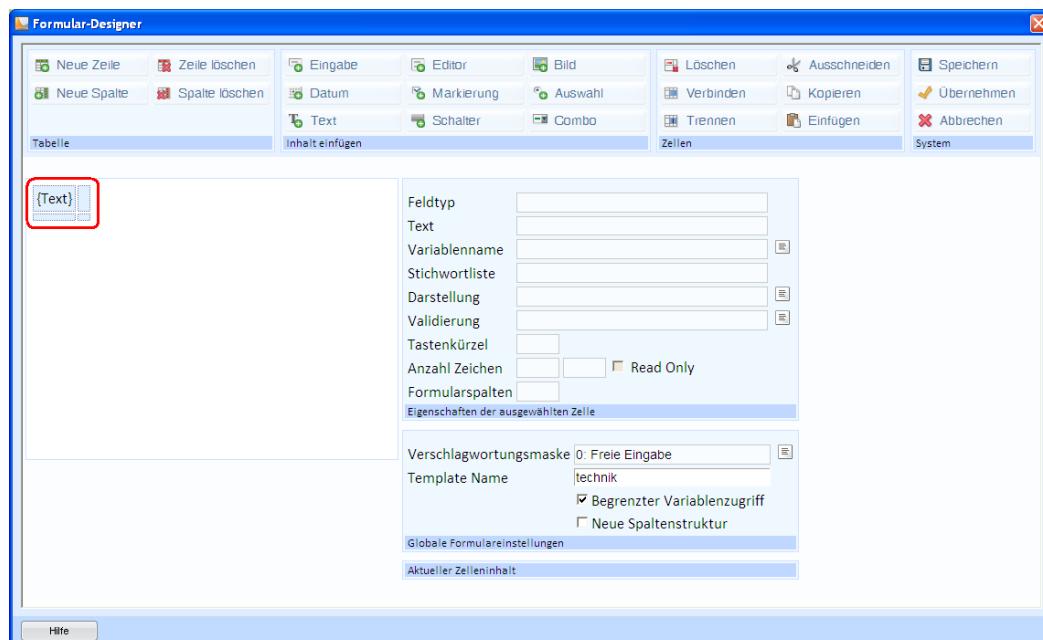
Hinweis: Bitte beachten Sie, dass als Formular oder Templatename nur „Identifier“ zulässig sind. D.h. der Name muss mit einem Buchstaben beginnen und danach dürfen bis zu 20

weitere Buchstaben oder Ziffern folgen. Umlaute, Leerzeichen oder Sonderzeichen sind nicht zulässig.

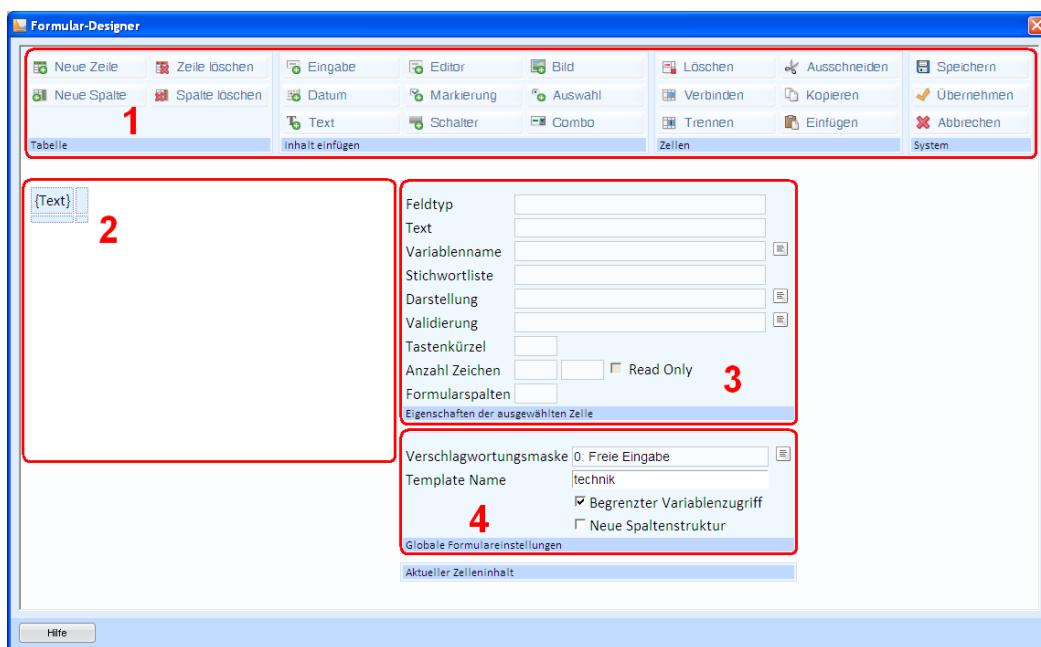
In dem Formular werden nun die Templates „technik“ und „inhalt“ erzeugt. Wenn ein Template erzeugt wird, wechselt die Ansicht automatisch in den Formulareditor.



Das leere Formular enthält als Vorgabe bereits zwei Spalten mit zwei Zeilen, im Bild rot markiert.



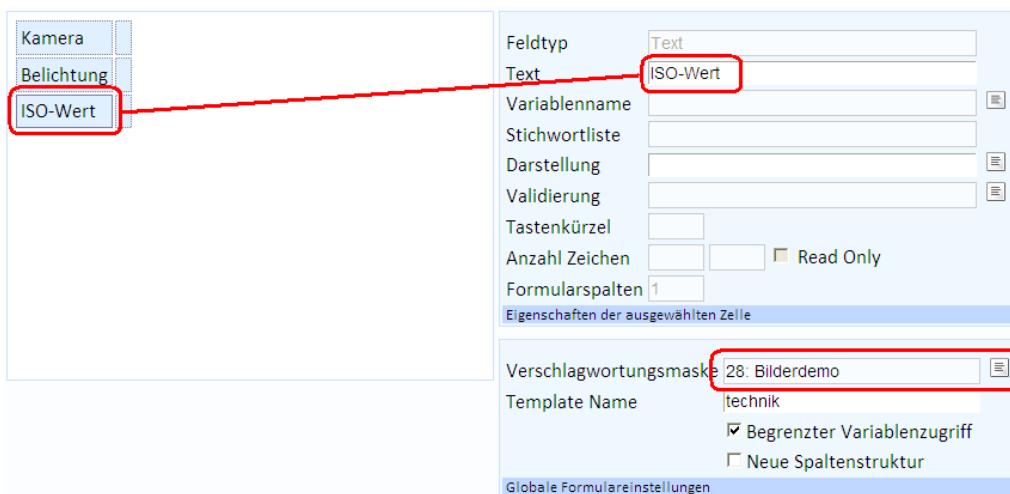
Da wir für den technischen Teil drei Zeilen benötigen, kann die zusätzliche Zeile durch ein Klicken auf die Schaltfläche „Neue Zeile“ aus der Werkzeugeiste erzeugt werden.



Der Formulareditor besteht aus vier Bereichen:

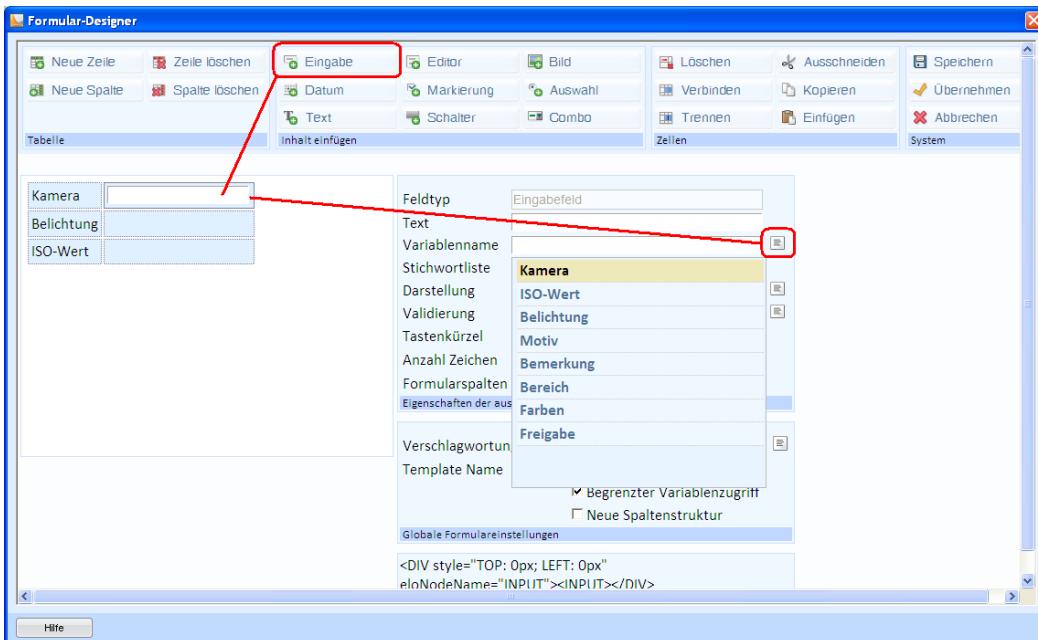
- 1 - Die Werkzeugeleiste: Hier finden Sie Aktionen zum Einfügen neuer Zeilen oder Spalten sowie von Text und Eingabefeldern.
- 2 - Der Formularbereich: Während der gesamten Bearbeitung wird das Formular so angezeigt, wie es der Anwender später auch sehen wird. Hier können Sie Tabellenzellen zur Bearbeitung durch einen Mausklick markieren.
- 3 - Die Zelleneigenschaften: Variablennamen, Texte oder Formatierungen zur aktuell ausgewählten Zelle im Formular werden hier eingegeben.
- 4 - Die globalen Formulareigenschaften: Hier kann man auswählen, für welche Ablagemaske das Formular gestaltet werden soll. Im Beispiel wird die Maske 28: Bilderdemo verwendet.

Zuerst werden die Beschreibungstexte in die erste Spalte eingetragen. Klicken Sie hierzu jeweils auf die entsprechende Zelle und tragen Sie rechts im Eigenschaften Feld unter „Text“ die jeweilige Bezeichnung ein.



Bevor Sie nun als nächstes die Eingabefelder anlegen, sollten Sie darauf achten, dass die Verschlagwortungsmaske richtig eingestellt ist. Andernfalls werden Sie keine Unterstützung für die Eingabe der Indexzeilen erhalten.

Klicken Sie nun in die oberste Zelle der zweiten Spalte hinter „Kamera“ und wählen Sie aus der Werkzeugleiste „Eingabe“ aus. In die Zelle wird nun ein Eingabefeld gelegt. Dieses können Sie nun im Eigenschaften Bereich bearbeiten.



Zuerst wird der Variablename eingetragen. Dieser verbindet ein Eingabefeld mit einer Indexzeile. Dafür gibt es ein komfortables Auswahlmenü, welches über das Stichwortlistensymbol rechts neben dem Eingabefeld aufgerufen werden kann. Wählen Sie hier „Kamera“.

Als nächstes wird die Stichwortliste benötigt, die wir in der Verschlagwortungsmaske für das Feld Kamera definiert haben. Diese können Sie einfach durch ein Doppelklick in dem Feld

Stichwortliste automatisch auf den Variablennamen erzeugen lassen, da wir im Beispiel die Standardliste verwenden.

Verfahren Sie gleichartig für die Felder „Belichtung“ und „ISO-Wert“. Beachten Sie dabei, dass wir für das Feld „Belichtung“ keine Stichwortliste definiert haben. Sie sollten deshalb im Formular auch keine Liste anmelden, der Anwender würde sonst nur eine leere Liste angezeigt bekommen.

Kamera	
Belichtung	
ISO-Wert	

Feldtyp **Eingabefeld**
Text
Variablenname **IX_GRP_BISO**
Stichwortliste **BISO**
Darstellung
Validierung **null**
Tastenkürzel
Anzahl Zeichen **20** Read Only
Formularspalten **1**
Eigenschaften der ausgewählten Zelle

Verschlagwortungsmaske 28: Bilderdemo
Template Name **technik**
 Begrenzter Variablenzugriff
 Neue Spaltenstruktur
Globale Formulareinstellungen

Das Teilformular ist nun fertig und kann über die Schaltfläche „Speichern“ gesichert werden. Legen Sie anschließend noch das zweite Teilformular für den Inhalt an.

Motiv	
Bemerkung	

Feldtyp **Eingabefeld**
Text
Variablenname **IX_GRP_BIBEM**
Stichwortliste
Darstellung
Validierung **null**
Tastenkürzel
Anzahl Zeichen **20** Read Only
Formularspalten **1**
Eigenschaften der ausgewählten Zelle

Verschlagwortungsmaske 28: Bilderdemo
Template Name **inhalt**
 Begrenzter Variablenzugriff
 Neue Spaltenstruktur
Globale Formulareinstellungen

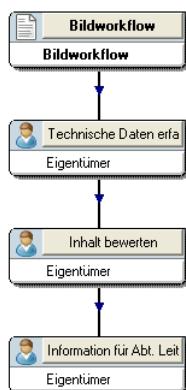
Nachdem Sie es mit „Speichern“ gesichert haben, sehen Sie in der Formularliste dieses neue Formular mit den beiden Templates. Die Funktion „Kopfdaten bearbeiten“ werden wir später betrachten.



Das Workflowformular ist nun fertig. Es kann nun in einem Workflow verwendet werden. Sie können den Dialog nun mit „Schließen“ verlassen.

3 Formular einbinden in einen Workflow

In unserem Beispiel verwenden wir einen ganz einfachen Workflow. In einem ersten Schritt trägt der Fotograf die technischen Daten zum Bild ein. Er leitet den Workflow dann an den Art Director weiter, der die inhaltliche Bewertung vornimmt. Und weil der Abteilungsleiter immer auf dem Laufenden gehalten werden will, bekommt er zum Abschluss das Bild noch mal zur Information vorgelegt.

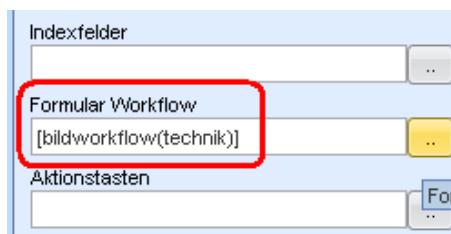
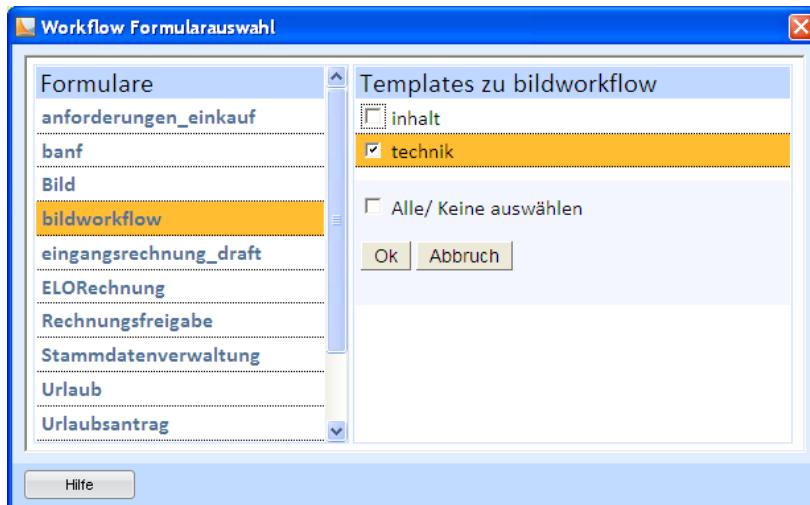
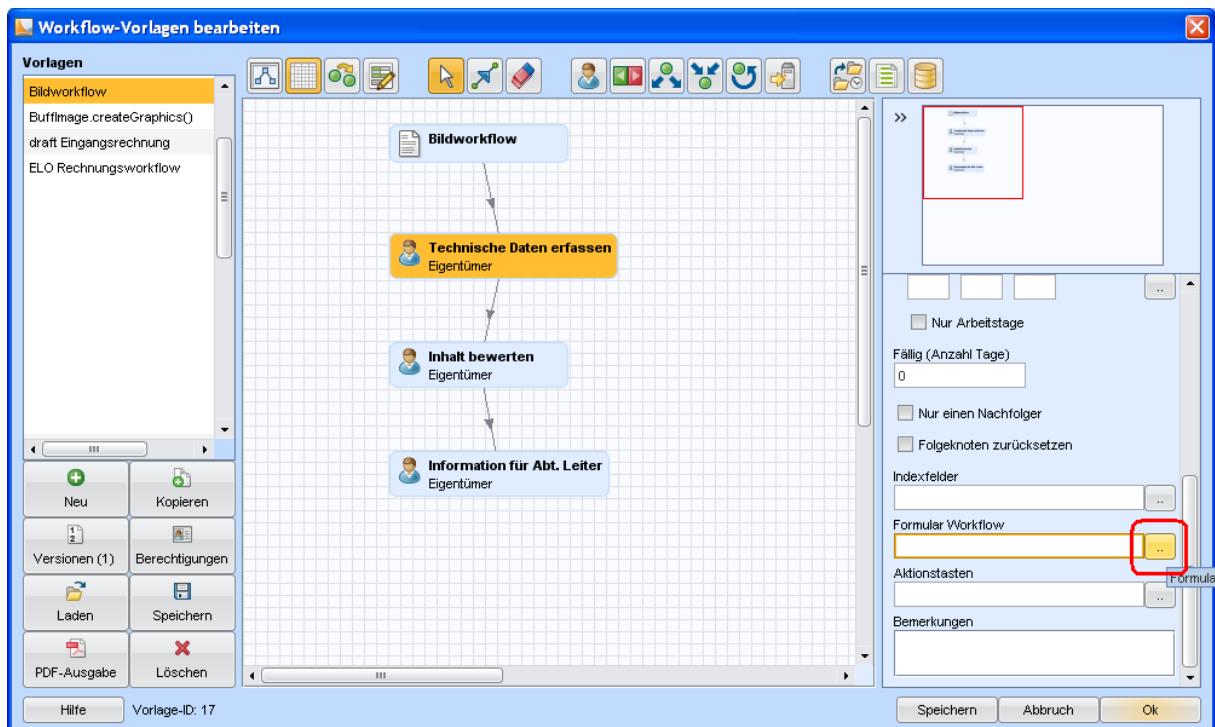


Hinweis: während der Testphase eines Workflows ist es oft praktisch, wenn man sich alle Personenknoten mit dem Anwender „Eigenümer“ belegt. Dann sieht man alle Workflowschritte, ohne dass man sich jeweils unter einem anderen Namen anmelden muss. Falls das nicht möglich ist, weil der Mitarbeiter z.B. durch Skripte ausgewertet oder bearbeitet wird, kann man sich für die jeweiligen Personen im Testsystem eine Vertretung eintragen.

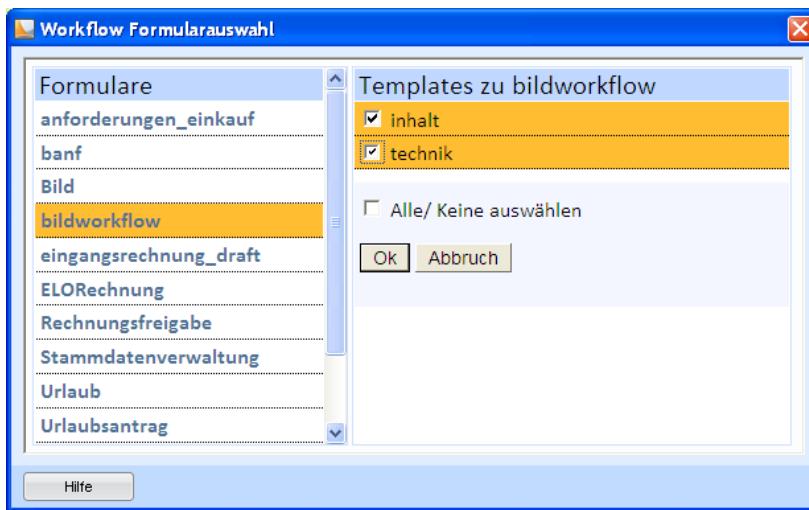
In den einzelnen Workflowknoten muss nun eingetragen werden, welches Formular angezeigt wird und welche Teilformulare der Anwender zu diesem Zeitpunkt sehen soll.

Eingetragen werden muss der Name des Formulars. Danach folgt in runden Klammern eine Liste aller Teilformulare, die an dieser Stelle sichtbar sein sollen. Das Ganze wird dann in eckige Klammern gestellt, damit es von normalen Indexfeld Informationen unterschieden werden kann. Für den ersten Schritt „Technische Daten erfassen“ handelt es sich um das Formular „bildworkflow“ und um das Teilformular „technik“ (alles klein schreiben, ohne Leerzeichen). Daraus ergibt sich dann der Eintrag [bildworkflow(technik)].

Sie können die Auswahl natürlich auch mithilfe eines Auswahldialogs vereinfachen.



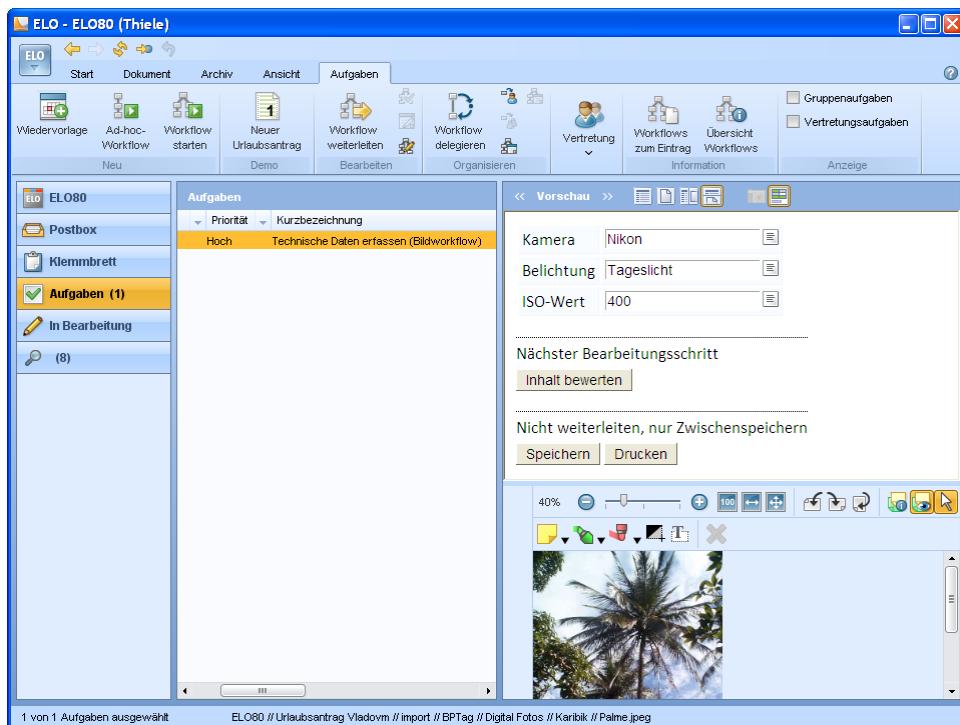
Der Art Director im zweiten Schritt soll sowohl die Technik Felder wie auch die Inhaltsfelder sehen. Er bekommt deshalb den Eintrag [bildworkflow(technik,inhalt)]. Das gleiche gilt für den Abteilungsleiter am Ende der Kette.



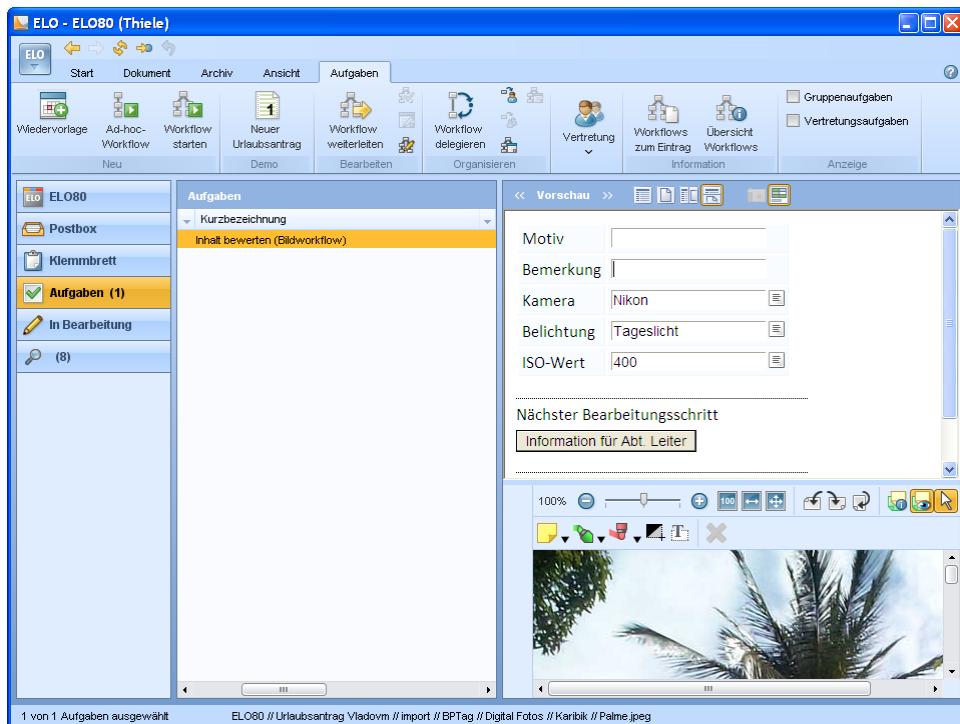
Damit ist der Workflow vorbereitet. Legen Sie nun ein Dokument mit der Verschlagwortungsmaske an und starten Sie den neuen Workflow.

Hinweis: normalerweise werden die Templates in alphabetischer Reihenfolge sortiert angezeigt. Sie können von der Standardreihenfolge abweichen, indem Sie im Workflowknoten die Templatereihenfolge an Ihre speziellen Anforderungen anpassen: [bildworkflow(inhalt,technik)] oder [bildworkflow(technik,inhalt)].

In der Aufgabenliste finden Sie nun die neue Aufgabe vor. Wenn Sie diese auswählen, können Sie das Formular sofort bearbeiten, ohne dass ein spezieller Edit Dialog geöffnet werden muss. Der Fotograf kann nun die technischen Daten einfüllen und den Workflow durch Klicken auf „Inhalt bewerten“ an den Art Director weiter leiten. Falls er das Formular nur teilweise ausgefüllt hat, aber erst noch weitere Details abklären möchte, kann er den Zwischenstand durch Klicken auf „Speichern“ sichern, ohne dass der Workflow weiter geleitet wird.



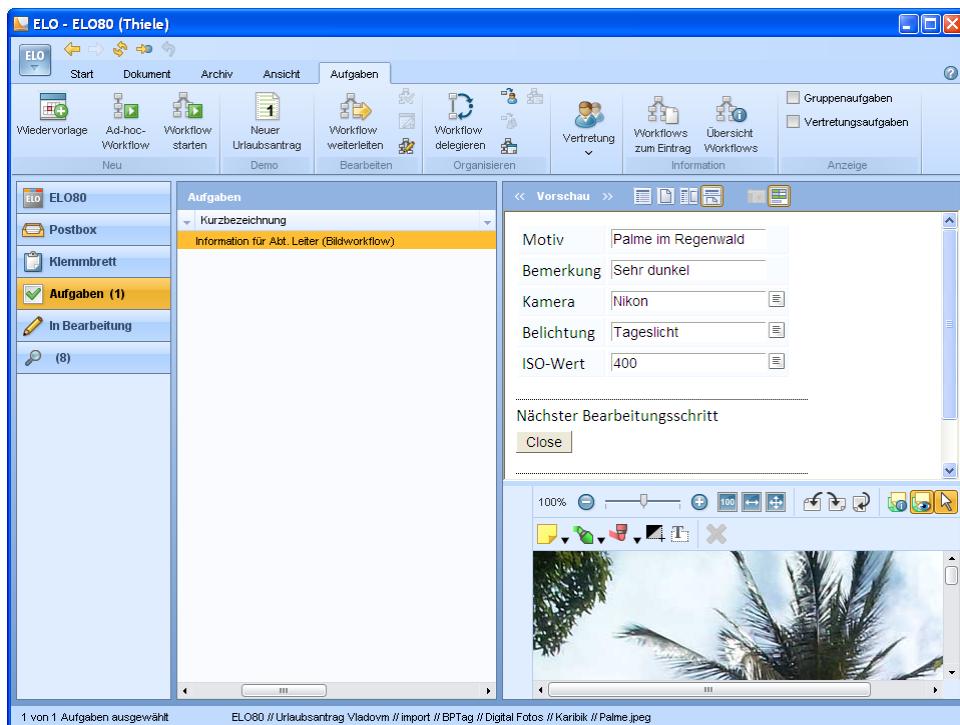
Der Art Director kann dann als nächstes die inhaltliche Bewertung vornehmen. Er sieht dabei die Eingaben des Fotografen, hat seine eigenen Felder aber ganz oben im direkten Zugriff.



Letztendlich landet der Workflow beim Abteilungsleiter. Er sieht alle Eingaben aus dem Workflow und kann ihn anschließend beenden.

Technische Dokumentation

Verwendung des ELO Workflowformulareditors



4 Weitere Optimierung

Dieser erste Workflow hat noch ein paar Schwächen, die im Folgenden verbessert werden sollen.

- In der Gesamtansicht sind die Bereiche für die technischen Daten und für den Inhalt nicht sichtbar getrennt. Es soll jeweils eine Zwischenüberschrift eingefügt werden.
- Das Motiv ist so wichtig, dass es farblich hervorgehoben werden soll.
- Der Art Director sieht die technischen Daten nicht nur, er kann sie auch irrtümlich verändern. Das ist nicht gewollt. Für ihn sollen diese Felder komplett schreibgeschützt sein.
- Der Abteilungsleiter will das Bild nur zur Kenntnis nehmen und gar nichts verändern. Für ihn sollten alle Felder schreibgeschützt sein.

4.1 Erstellung von Zwischenüberschriften

Für die Überschrift wird in jedem Template eine zusätzliche Zeile benötigt. Diese muss ganz oben angelegt werden. Da der Editor nur das Anfügen von Zeilen am Ende oder nach der selektierten Zeile anbietet, wird die neue Zeile erst mal als zweite Zeile angelegt und anschließend der Inhalt aus der ersten Zeile nach unten verschoben. Die nachfolgenden Schritte führen Sie bitte für beide Teilformulare „technik“ und „inhalt“ aus.

Selektieren Sie dafür eine Zelle in der ersten Zeile und klicken Sie auf „Insert Row“.



Sie können nun mit der Maus den Text „Motiv“ und das Eingabefeld dahinter per Drag&Drop in die neue zweite Zeile ziehen.

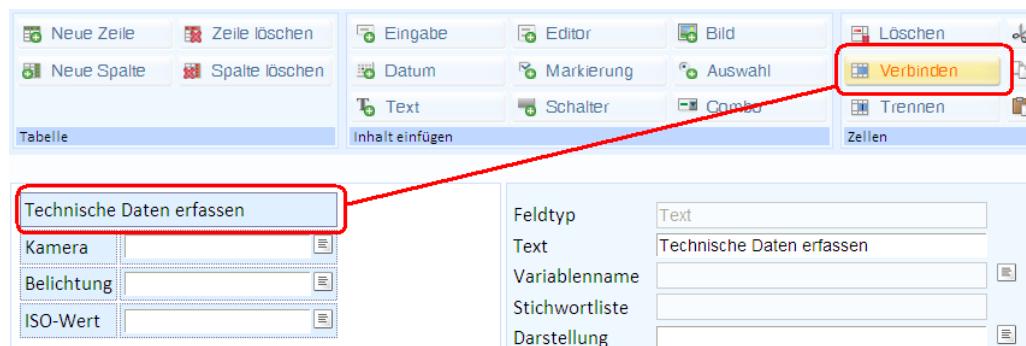


Während des Drag&Drop Vorgangs werden beim Ziehen der Maus alle Zellen, die als Ziel dienen können während der Überfahrt grün angezeigt. Eine rote Markierung zeigt, dass die Zelle bereits mit einem Inhalt belegt ist und keine zusätzlichen Werte annehmen kann.

Als nächstes wird die Überschrift links oben in die erste Zelle eingetragen.

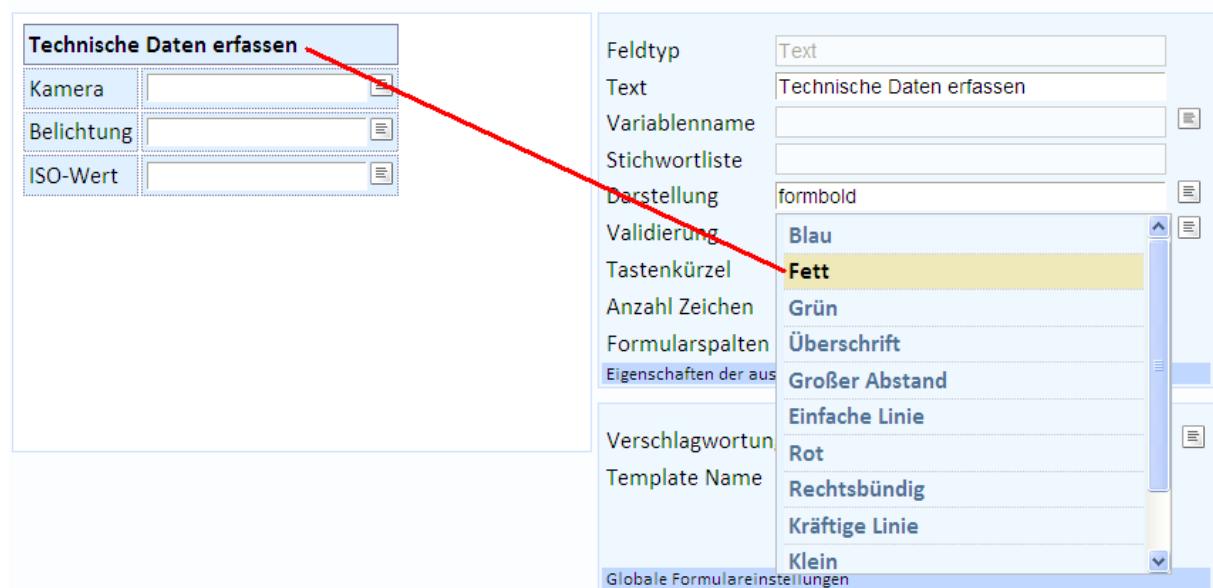
Technische Daten erfassen	
Kamera	
Belichtung	
ISO-Wert	

Das ist noch nicht ganz das gewünschte Ergebnis. Diese Überschrift sollte nicht auf die linke Spalte begrenzt sein und diese unnötig verbreitern. Das kann man erreichen, indem man die Zelle links mit der Zelle rechts verbindet. Dazu klicken Sie einfach auf „Verbinden“.



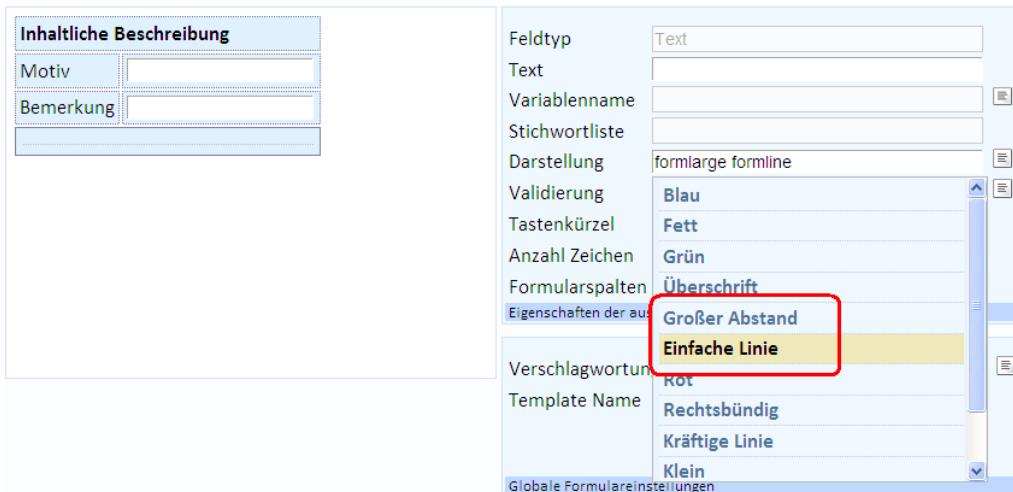
The screenshot shows a table editor interface. On the left, there's a table with four rows: 'Technische Daten erfassen' (header), 'Kamera', 'Belichtung', and 'ISO-Wert'. The first column has a single cell for each row. On the right, there's a properties panel for the first row. A red box highlights the first cell of the first row. A red arrow points from this cell to a toolbar at the top of the editor, where the 'Verbinden' (Connect) button is highlighted with a yellow box. The toolbar also includes other buttons like 'Neue Zeile' (New Row), 'Zeile löschen' (Delete Row), 'Eingabe' (Input), 'Editor', 'Bild' (Image), 'Löschen' (Delete), 'Neue Spalte' (New Column), 'Spalte löschen' (Delete Column), 'Datum' (Date), 'Markierung' (Marker), 'Auswahl' (Selection), 'Trennen' (Separate), 'Text' (Text), 'Schalter' (Switch), 'Content', and 'Zellen' (Cells).

Nun soll die Überschrift noch hervorgehoben werden, das kann man über eine Einstellung in der Eigenschaft „Darstellung“ erreichen.

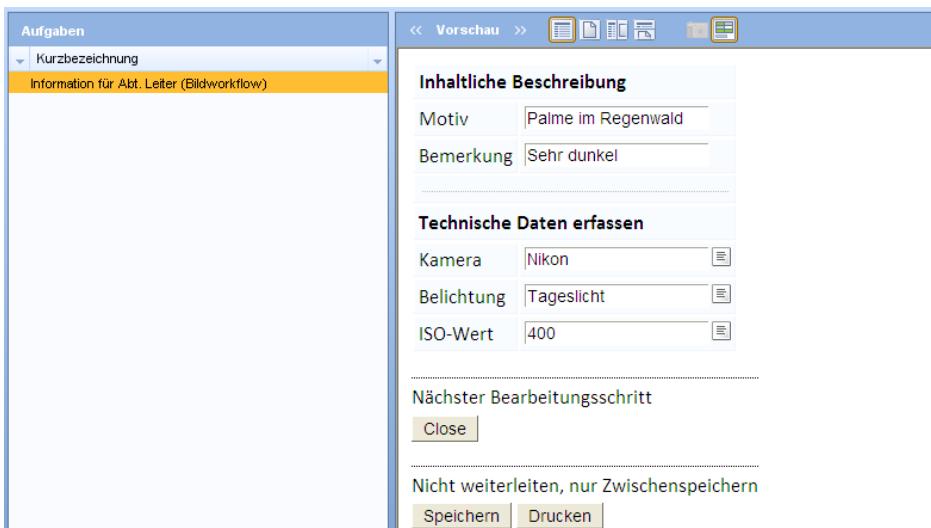


The screenshot shows the same table editor interface. The first cell of the first row now has a bold font style applied. A red arrow points from this bolded cell to a properties panel on the right. The properties panel lists several properties: 'Feldtyp' (Field type) set to 'Text', 'Text' set to 'Technische Daten erfassen', 'Variablenname' (Variable name), 'Stichwortliste' (Keyword list), 'Darstellung' (Presentation) set to 'formbold', 'Validierung' (Validation), 'Tastenkürzel' (Keyboard shortcut), 'Anzahl Zeichen' (Number of characters), 'Formularspalten' (Formular columns), 'Eigenschaften der aus' (Properties of the out), 'Verschlagwortung' (Indexing), 'Template Name', and 'Globale Formulareinstellungen' (Global form settings). The 'Fett' (Bold) option under 'Darstellung' is highlighted with a yellow box.

Nach dem Speichern werden die gleichen Operationen für das Teilformular „inhalt“ durchgeführt. Damit die beiden Bereiche deutlicher getrennt werden, erhält das Teilformular auch eine (fast) leere Zeile am Ende. In der Grundeinstellung wird so eine leere Tabellenspalte nur sehr klein dargestellt. Damit sie formatiert werden kann, muss sie einen Inhalt erhalten. Durch ein Klick auf das Werkzeug „Text“ kann man ein (leeres) Textfeld einfügen und in der Darstellung nacheinander die Optionen „Großer Abstand“ und „Einfache Linie“ auswählen.



Das gesamte Formular aus der Sicht des Abteilungsleiters stellt sich nun so dar:

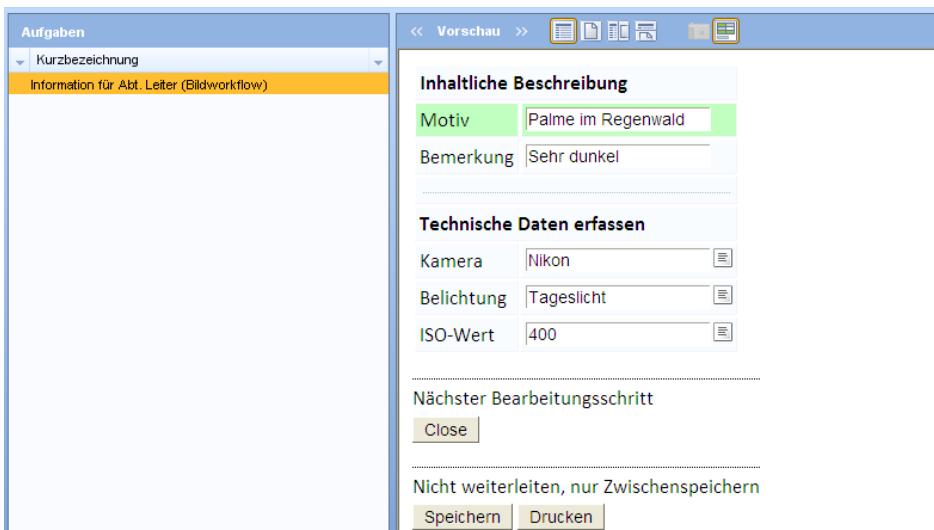


4.2 Motiv farblich hervorheben

Die einzelnen Zellen können im Workfloweditor mit einer Farbe hinterlegt werden. Hierzu rufen Sie das Teilformular „inhalt“ im Editor auf und selektieren die Zelle „Motiv“. In den Eigenschaften auf der rechten Seite wählen Sie dann unter „Darstellung“ in der Auswahlliste „Grün“. Das gleiche machen Sie dann mit dem Eingabefeld rechts daneben.

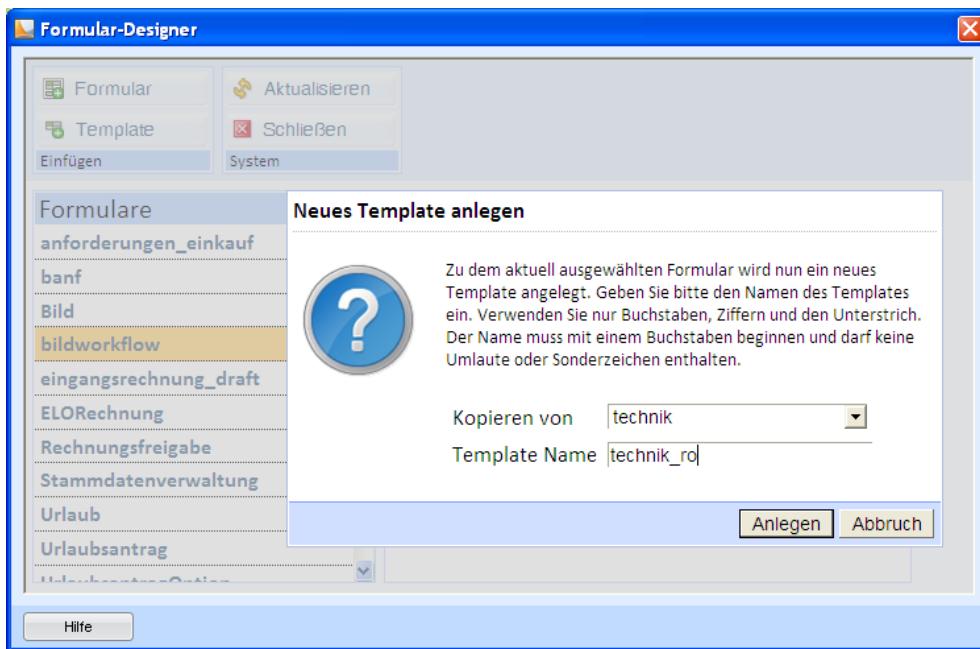


Der Abteilungsleiter sieht nun das hervorgehobene Motiv.

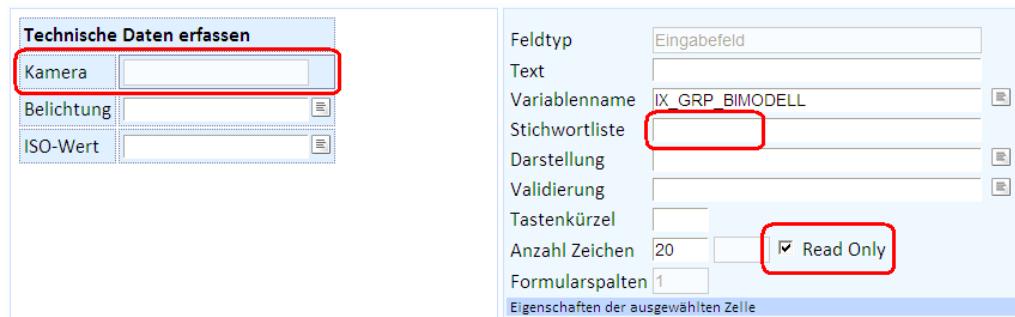


4.3 Schreibschutz für Teilformulare

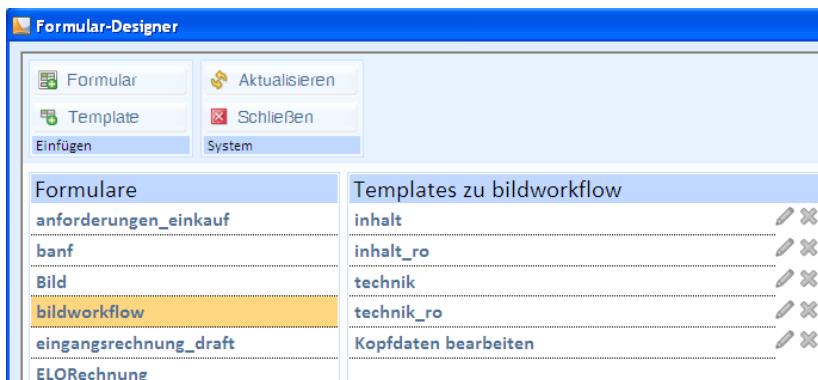
Als nächstes sollen die geschützten Teilformulare erstellt werden. Diese sollte man immer erst dann erzeugen, wenn die Gestaltung des Formulars ansonsten bereits vollständig fertig gestellt ist. Gehen Sie dafür in die Formular Auswahlliste und erzeugen Sie ein neues Teilformular „technik_ro“. Auch wenn der Name im Prinzip frei wählbar ist, so ist es in diesem Fall sinnvoll, dass er mit „technik“ beginnt. In diesem Fall erkennen auch spätere Bearbeiter, dass Sie eine Variante eines bereits bestehenden Formulars erstellt haben. Wählen Sie unter „Kopieren von“ den Eintrag „technik“ aus und klicken auf „Anlegen“. Das System legt nun ein neues Template an und kopiert den Inhalt des Teilformulars „technik“ in das neue Teilformular „technik_ro“. Das erspart Ihnen die Arbeit, den Inhalt erneut erstellen zu müssen.



Klicken Sie nun das Eingabefeld hinter „Kamera“ an und aktivieren Sie im Eigenschaften Bereich „Read Only“. Zusätzlich sollten Sie noch die Stichwortliste entfernen, andernfalls kann der Anwender zwar keine direkten Eingaben mehr vornehmen – aber über die Stichwortliste geht es dann trotzdem (das kann durchaus gewünscht sein, z.B. wenn man sicher stellen will, dass nur vordefinierte Werte eingetragen werden).

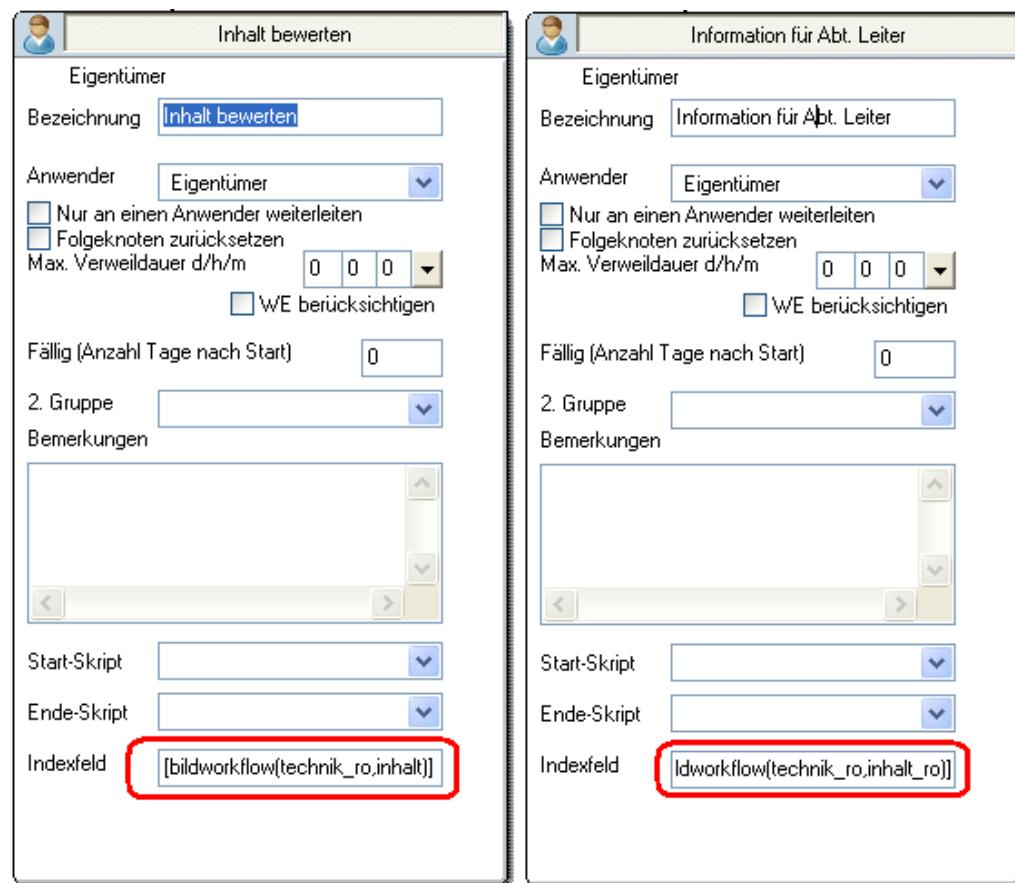


Die gleichen Operationen führen Sie nun für die Eingabefelder zur Belichtung und zum ISO-Wert aus. Speichern Sie nun das Teilformular und erzeugen auf die gleiche Art und Weise ein „inhalt_ro“ Teilformular.



Nun haben wir alle benötigten Teilformulare, es muss nur noch dafür gesorgt werden, dass sie auch vom Workflow verwendet werden. Hierzu rufen Sie wieder den Workflow Designer auf und wählen den Bild Workflow aus.

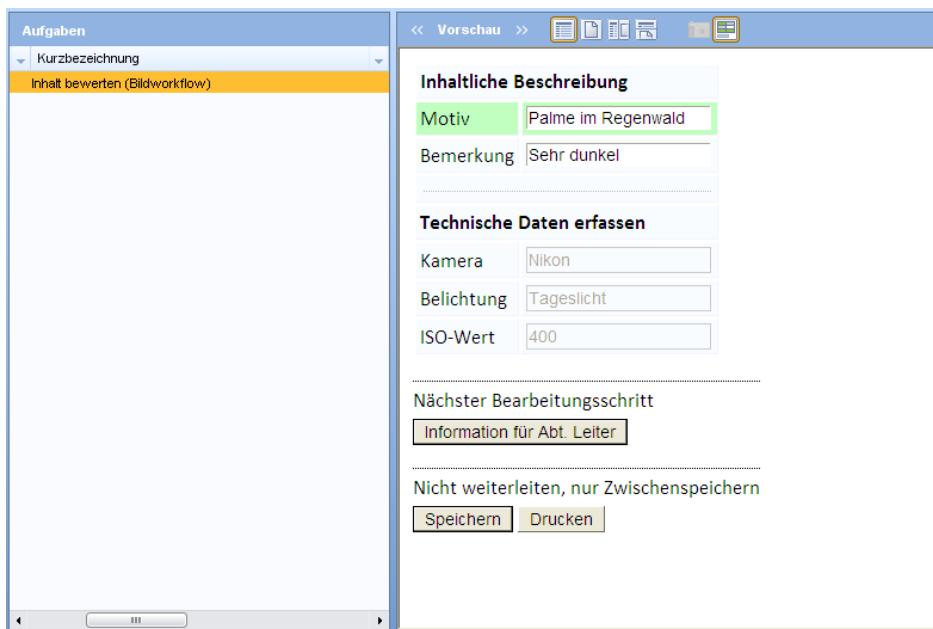
Öffnen Sie zuerst den Knoten für den Inhalt und ändern Sie den Namen für das Teilformular „technik“ in „technik_ro“. Das gleiche machen Sie beim Knoten des Abteilungsleiters, dort werden beide Teilformulare in der _ro Varianten verwendet.



The 'Inhalt bewerten' panel has an 'Indexfeld' field containing `[bildworkflow(technik_ro,inhalt)]`. The 'Information für Abt. Leiter' panel has an 'Indexfeld' field containing `[ldworkflow(technik_ro,inhalt_ro)]`.

Speichern Sie nun die Workflowvorlage und starten Sie einen neuen Workflow.

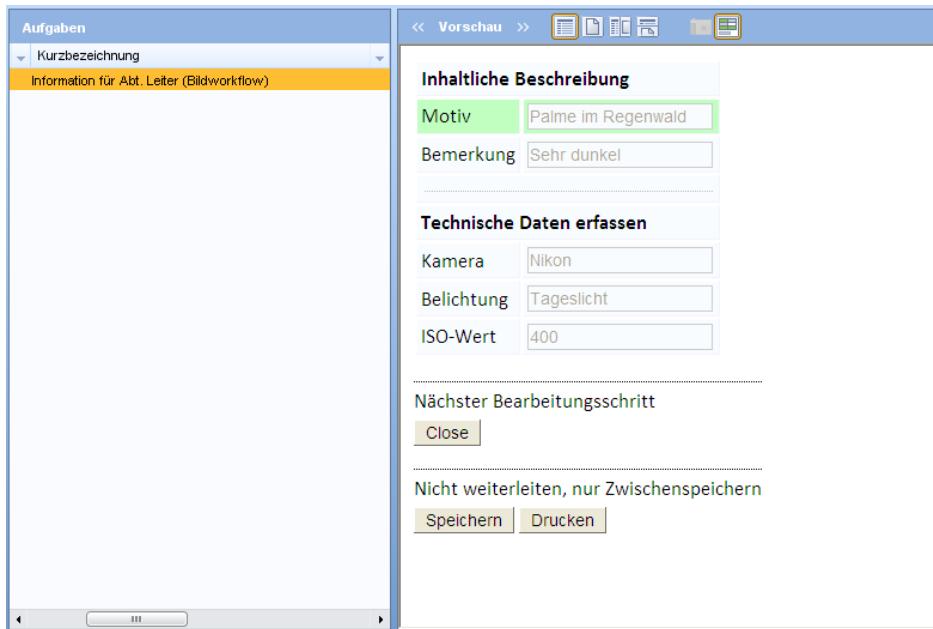
Der Art Director kann nun nur noch die inhaltlichen Felder bearbeiten, die technischen Felder sind für ihn schreibgeschützt (ausgegraut).



The screenshot shows a workflow editor interface. On the left, a sidebar titled "Aufgaben" lists a task named "Inhalt bewerten (Bildworkflow)". The main area is a form titled "Inhaltliche Beschreibung". It contains several input fields:

- Inhaltliche Beschreibung**
 - Motiv: Palme im Regenwald
 - Bemerkung: Sehr dunkel
- Technische Daten erfassen**
 - Kamera: Nikon
 - Belichtung: Tageslicht
 - ISO-Wert: 400
- Nächster Bearbeitungsschritt:
 - Information für Abt. Leiter (highlighted in orange)
- Nicht weiterleiten, nur Zwischenspeichern
 - Speichern
 - Drucken

Für den Abteilungsleiter sind alle Felder ausgegraut, er kann den Vorgang nur noch zur Kenntnis nehmen aber keine Änderungen mehr durchführen.



This screenshot shows the same workflow editor interface, but the task in the sidebar is now "Information für Abt. Leiter (Bildworkflow)". The main form area is identical to the previous screenshot, but the "Close" button in the "Nächster Bearbeitungsschritt" section is highlighted in orange.

5 Mehrfacheingaben

In manchen Situationen benötigt man mehr als ein Eingabefeld für Informationen, die mehrfach vorkommen können. Das können in einem Rechnungsworkflow die Rechnungspositionen sein. Oder in einem Urlaubsworkflow die Termine (falls in einem Formular gleich mehrere Termine beantragt werden dürfen).

Zur Realisierung dieser Anforderung sind zwei Hürden zu nehmen: zuerst stellt sich die Frage, wie diese Daten gespeichert werden. Und als nächstes die Frage, wie sie im Formular aufgeführt werden sollen.

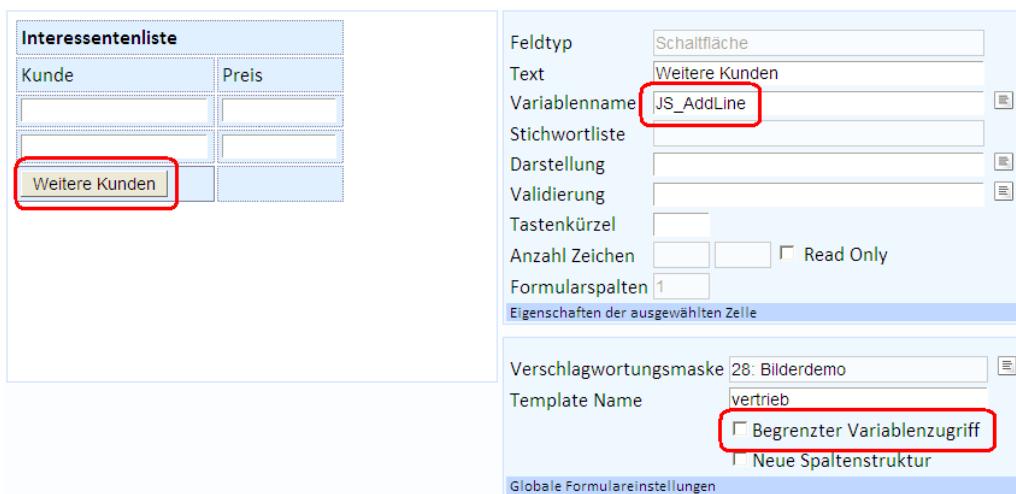
Zur Speicherung gibt es die Möglichkeit, eine Feldstruktur zu verwenden. Auch wenn ELO bei den Indexzeilen kein Array zur Verfügung stellt, kann man es über eine passende Benennung simulieren. Wenn das Rechnungsformular für jede Rechnungsposition zwei Felder besitzt (z.B. ARTIKEL für die Artikelbezeichnung und PREIS für den Preis dieser Rechnungsposition), dann kann man zum Speichern die Indexzeilen IX_GRP_ARTIKEL1, IX_GRP_ARTIKEL2... sowie IX_GRP_PREIS1, IX_GRP_PREIS2... verwenden. Der Nachteil bei der Verwendung von Indexzeilen liegt darin, dass diese in der Zahl begrenzt sind. Und, da sie im Maskeneditor vorab angelegt werden müssen, ist die Zeilenzahl im Formular ebenfalls begrenzt.

Unter ELOprofessional 8.0 gibt es deshalb einen zusätzlichen Datenspeicher. Der muss nicht konfiguriert werden sondern stellt einfach beliebig viele Indexzeilen mit einem frei wählbaren Namen zur Verfügung. Angesprochen wird dieser Speicher über Variablennamen IX_MAP_..., also im Beispiel IX_MAP_ARTIKEL1, IX_MAP_ARTIKEL2... . Hier könnte das Formular prinzipiell beliebig viele Daten ablegen, man muss nur die Performance im Auge behalten – wenn hier tausende von Werten verwendet werden, wird der Dokumentenzugriff entsprechend langsam.

Das zweite Problem war die Anzeige im Formular. Ein Papierformular hat an dieser Stelle im Allgemeinen eine feste Anzahl von Zeilen. Diese können verwendet werden und wenn alle voll sind, wird ein weiteres Formular benötigt. Dieses Verfahren kann man natürlich auch am Computer abbilden, es geht aber auch besser: es wird nur eine (oder einige wenige Zeilen) vorab angezeigt. Wenn der Anwender weitere Einträge benötigt, dann erhält er am Ende der Liste eine „Weitere“ Schaltfläche, die auf Anforderung eine zusätzliche Zeile erstellt.

5.1 Interessentenverwaltung im Bildarchiv

Im Folgenden wollen wir nun den Abteilungsleiter aus seiner passiven Rolle entlassen. Er bekommt nun die Aufgabe, zu einem neuen Bild eine Liste der möglichen Interessenten zu benennen und einen Soll-Preis festzulegen.



Dafür wird ein drittes Teilformular „vertrieb“ angelegt. Dieses enthält eine Interessentenliste mit zwei Spalten: dem Kunden und einen Preis. Vorbereitet werden zwei Zeilen um dem Anwender zu signalisieren: hier kann man mehrere Zeilen eingeben. Am Ende des Formulars wird ein Button „Weitere Kunden“ angelegt. Dieser hat als Variablennamen „JS_AddLine“ – das ist ein vordefinierter Name, der die Zeile über dem Button in eine neue, leere Zeile kopiert.

Hinweis: ein Formular hat in der Voreinstellung die Eigenschaft „Begrenzter Variablenzugriff“ aktiviert. Damit wird sichergestellt, dass im Formular nur die Indexzeilenvariablen übertragen werden, die auch angezeigt werden. Bei einem dynamischen Formular geht das aber nicht. Da ist beim Aufruf der Anzeige noch nicht klar, welche Variablen tatsächlich verwendet werden. Deshalb muss man in diesem Fall sicherheitshalber alle Variablen übertragen. Aus diesem Grund muss für dieses Beispiel die Eigenschaft „Begrenzter Variablenzugriff“ abgeschaltet werden.

In der Workflow Vorlage muss nun noch im Knoten des Abteilungsleiters das zusätzliche Formular eingetragen werden.



Bei der Anzeige der Aufgabe stellt sich die Ansicht für den Abteilungsleiter nun so dar. Er kann die beiden Zeilen der Interessentenliste ausfüllen.

Aufgaben

Kurzbezeichnung

Information für Abt. Leiter (Bildworkflow)

Inhaltliche Beschreibung

Motiv: Palme im Regenwald

Bemerkung: Sehr dunkel

Technische Daten erfassen

Kamera: Nikon

Belichtung: Tageslicht

ISO-Wert: 400

Interessentenliste

Kunde	Preis
Meier	10,00
Müller	20,00
Weitere Kunden	

Nächster Bearbeitungsschritt

Close

Wenn er einen dritten Interessenten eintragen möchte, dann klickt er einfach auf „Weitere Kunden“. Die zusätzliche Zeile kann er dann mit Daten füllen.

Interessentenliste	
Kunde	Preis
Meier	10,00
Müller	20,00
Schulze	30,00
Weitere Kunden	

Interessentenliste	
Kunde	Preis
Meier	10,00
Müller	20,00
Schulze	30,00
Xaver	55,00
Weitere Kunden	

5.2 Listenlänge begrenzen

Im nächsten Schritt wollen wir verhindern, dass der Abteilungsleiter gleich hundert Interessenten einträgt. Er soll auf die 10 wichtigsten Kandidaten begrenzt werden. Hierzu gibt es im „Weitere“ Button eine Option für die maximale Anzahl weiterer Eingabezeilen. Sie wird mit „max:<Zahl>“ in den Optionen hinterlegt. Da es bereits zwei Zeilen gibt, sollen noch maximal 8 weitere Zeilen angelegt werden – daher die Einstellung „max:8“.

Interessentenliste	
Kunde	Preis
Weitere Kunden	

Anzahl Zeichen Read Only

Variablenname JS_AddLine

Stichwortliste

Text Weitere Kunden

Class

Validierung max:8

Selected Cell Properties

Vorschlagwortungsmaske 28: Bilderdemo

Template Name vertrieb

Begrenzter Variablenzugriff

Global Form Properties

Beim Abteilungsleiter wird dann nach der 10. Zeile der Button „Weitere Kunden“ ausgegraut.

Hinweis: falls in der Datenbank mehr Daten als für 10 Zeilen vorhanden sind, werden diese weiterhin angezeigt. Es können dann aber trotzdem keine weiteren Zeilen angelegt werden.

5.3 Gruppen duplizieren

In vielen Fällen wird es ausreichen, wenn man tabellenartig jeweils eine Zeile duplizieren kann. Manchmal kann man die Mehrfacheingabe aber nicht auf eine Zeile beschränken. Deshalb kann sich die Kopierfunktion bei Bedarf auch über mehrere Zeilen erstrecken.

Das neue Interessenfeld besteht nun aus drei Zeilen. In der ersten Zeile werden weiterhin der Kundenname und der Preis aufgeführt. In der zweiten Zeile gibt es ein Bemerkungsfeld über die gesamte Breite. Zum Abschluss gibt es noch eine Leerzeile, damit mehrere Eingabegruppen optisch getrennt werden.

Um dem System mitzuteilen, dass der Kopierbereich nun aus drei Zeilen besteht, muss in der Validierungseigenschaft der zusätzliche Wert „lines:3“ eingetragen werden.

Interessentenliste	
Kunde	Preis
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="button" value="Weitere Kunden"/>	

1
2
3

Anzahl Zeichen Read Only

Variablenname JS_AddLine

Stichwortliste

Text Weitere Kunden

Class

Validierung max:9 lines:3

Selected Cell Properties

Verschlagwortungsmaske 28: Bilderdemo

Template Name vertrieb

Begrenzter Variablenzugriff

Global Form Properties

Der Abteilungsleiter sieht nun folgende Ansicht im Teiformular „vertrieb“:

Interessentenliste	
Kunde	Preis
Meier	100,00
Sehr Preissensibel, vorsichtig kalkulieren	
Müller	120,00
Hohe Qualitätsanforderungen	
Weitere Kunden	

5.4 Automatische Positionszähler

In manchen Fällen möchte man vor der Zeilenliste einen Positionszähler anzeigen (fortlaufende Nummer 1, 2, 3 ...). Dazu setzt man ein schreibgeschütztes Eingabefeld mit der Validierung „count:auto“ ein. Wenn so eine Zeile dupliziert wird, erhöht das System den eingetragenen Wert automatisch um den Wert 1. Da solche Zähler im Normalfall nicht mit gespeichert werden sollen, kann man hier den Variablenamen „NOSAVE1“ verwenden. Wenn die Position mit gespeichert wird, muss man beachten, dass die automatische Unterdrückung leerer Zeilen nicht mehr eingesetzt werden kann, da dann eine Zeile niemals leer ist (sie enthält ja mindestens die Zeilennummer).

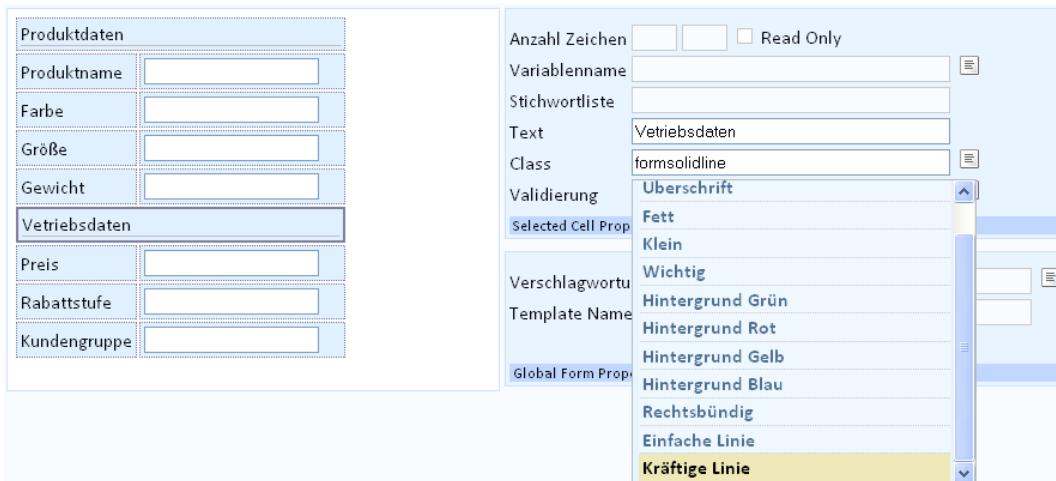
Gruppierung von Eingabebereichen

Im Beispiel weiter oben haben Sie gesehen, wie man unterschiedliche Eingabebereiche durch Leerzeilen (oder eben nur „fast“ Leerzeilen“) voneinander absetzen kann. Es gibt aber noch eine weitere Möglichkeit: unterstrichene Textzeilen. Hierdurch kann man die Bereiche gleich noch mit einer Überschrift versehen.

Im folgenden Beispiel werden in einem Teilformular zwei unterschiedliche Gruppen von Daten zusammengefasst dargestellt. Die gleichförmige Liste ergibt ein „langweiliges“ Aussehen und erschwert die Orientierung, wenn man sich nur für eine Teilgruppe interessiert.

Produktnam	
Farbe	
Größe	
Gewicht	
Preis	
Rabattstufe	
Kundengruppe	

Die Darstellung kann durch das Einfügen von Zwischenüberschriften deutlich verbessert werden. Hierfür werden zuerst am Anfang jeder Gruppe zusätzliche Zeilen eingefügt, die Spalten dieser Zeile verbunden und mit einem Text und einem Unterstrich versehen.



The screenshot shows a form editor interface. On the left, there is a preview of the form with two main sections: "Produktdaten" and "Vetriebsdaten". The "Produktdaten" section contains fields for "Produktnam", "Farbe", "Größe", and "Gewicht". The "Vetriebsdaten" section contains fields for "Preis", "Rabattstufe", and "Kundengruppe". On the right, the properties panel is open for the first row of the "Produktdaten" section. The "Text" property is set to "Vetriebsdaten" and the "Class" property is set to "formsolidline". A dropdown menu for "Validierung" is open, showing options like "Überschrift", "Fett", "Klein", "Wichtig", and "Hintergrund Grün". The "Überschrift" option is highlighted. The "Global Form Prop" dropdown at the bottom shows various styling options, with "Kräftige Linie" currently selected.

Im Aufgabenbereich sieht das dann so aus:

Produktdaten	
Produktname	<input type="text"/>
Farbe	<input type="text"/>
Größe	<input type="text"/>
Gewicht	<input type="text"/>
Vetriebsdaten	
Preis	<input type="text"/>
Rabattstufe	<input type="text"/>
Kundengruppe	<input type="text"/>

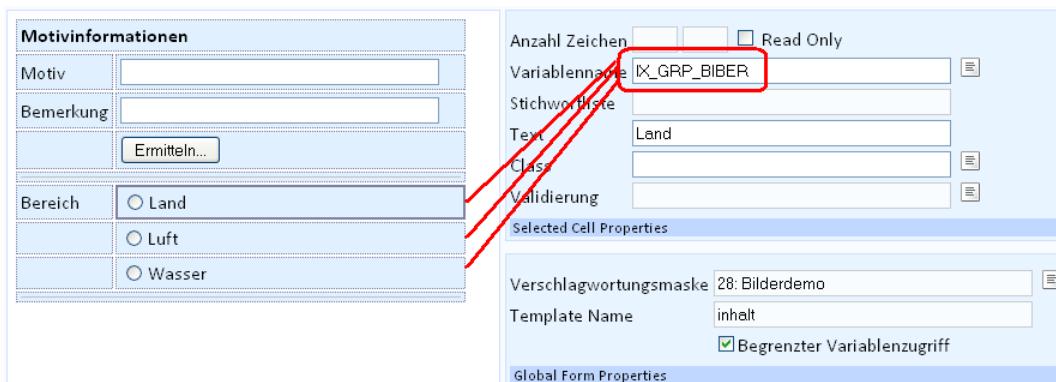
Gerade bei umfangreichen Listen kann so eine Gruppierung für eine erheblich bessere Lesbarkeit und Übersichtlichkeit führen. Auf jeden Fall sollte man den Anwender vor einer langen, monotonen und unübersichtlichen Liste beschützen.

6 Markierungsfelder und Auswahlfelder

In Formularen werden häufig Checkboxen (Markierung) und Radiobuttons (Auswahl) verwendet. Wenn zu einer Option nur ein einziges Element aus einer Liste von Möglichkeiten gewählt werden kann, verwendet man üblicherweise Radiobuttons. Falls mehrere Elemente gewählt werden können, ist eine Liste von Checkboxen sinnvoll.

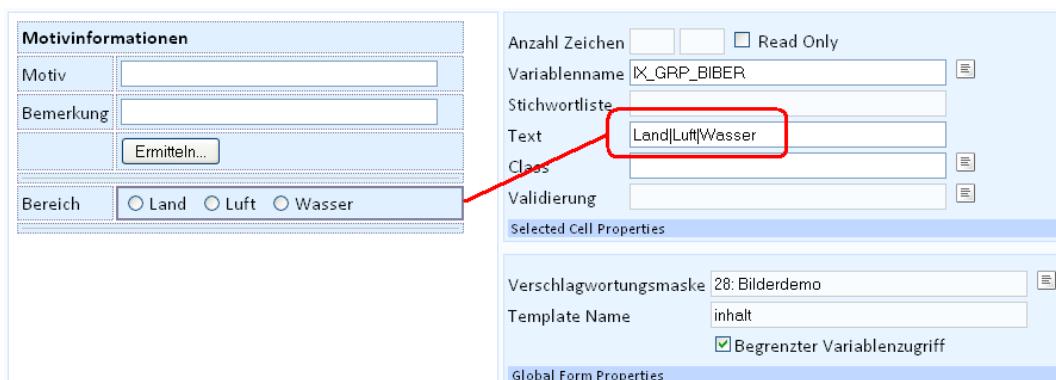
6.1 Radiobuttons (Auswahl)

Über die Werkzeugeleiste kann der Anwender einen Radiobutton in eine Zelle einfügen. Dieser besteht aus einem Variablennamen und einem Text. Mehrere Radiobuttons mit dem gleichen Variablennamen werden zu einer Gruppe zusammengefasst, d.h. der Anwender kann nur einen aus dieser Gruppe selektieren, der zuvor selektierte Eintrag wird dabei automatisch deseletiert.



In der Indexzeile der Verschlagwortung wird der Name der selektierten Zeile eingetragen oder ein Leerstring, wenn keine Option ausgewählt war.

Man kann die Optionen auch in einer Zeile zusammenfassen. Hierzu wird nur ein Radiobutton angelegt, im Namen werden dann alle Optionen, jeweils durch ein Pipe Symbol getrennt, aufgeführt: „Land|Luft|Wasser“.



Diese Variante ist natürlich nur dann sinnvoll einsetzbar, wenn es nur wenige Optionen gibt. Wenn es eine sehr umfangreiche Auswahl gibt, ist ein normales Textfeld mit einer Stichwortliste sinnvoller.

6.2 Checkboxen (Markierung)

Checkboxen können im Prinzip genauso eingesetzt werden wie Radiobuttons. Sie können in unterschiedlichen Zeilen verwendet werden, eine Option pro Zeile oder innerhalb einer Zeile hintereinander gesetzt werden.

Bei Checkboxen wird in der Indexzeile eine Liste der selektierten Einträge in numerischer Form abgelegt. Eine „0“ steht für keinen gewählten Eintrag, eine „1“ für den ersten Eintrag und eine „1|3“ für den ersten und dritten Eintrag. Eine leere Indexzeile steht ebenfalls für keinen gewählten Eintrag.

Motivinformationen	
Motiv	<input type="text"/>
Bemerkung	<input type="text"/>
	<input type="button" value="Ermitteln..."/>
Bereich	<input type="radio"/> Land <input type="radio"/> Luft <input type="radio"/> Wasser
Farben	<input type="checkbox"/> Rot <input type="checkbox"/> Gelb <input type="checkbox"/> Grün <input type="checkbox"/> Blau
Freigabe	<input type="checkbox"/> Freigabe

Anzahl Zeichen Read Only

Variablenname

Stichwortliste

Text

Class

Validierung

Selected Cell Properties

Verschlagwortungsmaske

Template Name

Begrenzter Variablenzugriff

Global Form Properties

6.3 Schreibschutz

Checkboxen und Radiobuttons können genauso wie andere Eingabefelder mit einem Schreibschutz belegt werden. Dabei ergibt sich ein Unterschied zwischen der Zeilen- und der Spaltendarstellung. Wenn mehrere Optionen in einer Zeile liegen, können Sie nur gemeinsam geschützt oder freigeschaltet werden. Liegen die Optionen in unterschiedlichen Zeilen, dann können sie individuell geschützt werden, auch wenn das in der Praxis nur selten sinnvoll sein wird.

7 Die komplette Werkzeugleiste



Die Werkzeugleiste besteht aus vier Bereichen:

- Dem Tabellenbereich: hier können dem Formular Zeilen und Spalten hinzugefügt oder entfernt werden.
- Den Steuerungselementen: hier können Textelemente oder Eingabefelder in die aktuell ausgewählte Tabellenzelle eingefügt werden.
- Dem Zellenbereich: hier können Zellen geleert, verbunden oder geteilt werden.
- Dem Systembereich: Speichern des Formulars oder Beenden der Bearbeitung ohne Speichern.

Die prinzipielle Vorgehensweise im Formulareditor läuft immer so, dass zuerst eine Zelle ausgewählt wird, auf die die Operation angewendet werden soll. Anschließend wird über die Werkzeugleiste durch Klicken eine Operation aufgerufen. Von diesem Schema weichen nur wenige Funktionen ab. Die Funktionen „Neue Zeile“ und „Neue Spalte“ fügen eine Zeile oder Spalte am Ende der Liste an, wenn keine Zelle selektiert ist. Und die Funktionen „Speichern“ und „Übernehmen“ speichern natürlich das komplette Formular, nicht nur die aktuelle Zelle. Mittels „Ausschneiden“, „Kopieren“ und „Einfügen“ können Zelleninhalte kopiert werden, wie man es auch von anderen Anwendungen gewohnt ist.

7.1 Verfügbare Eingabeelemente

- Eingabe: fügt eine Texteingabe in das Formular ein. Diese kann mit einer Stichwortliste belegt werden. Prinzipiell kann die Stichwortliste frei gewählt werden, durch einen Doppelklick in das Feld für die Stichwortliste kann automatisch die Standardliste für dieses Indexfeld eingefügt werden. Weiterhin kann das Eingabefeld über die Validierungsanweisungen zum Numerikfeld mit einem vorgegebenen Eingabebereich eingeschränkt werden.
- Datum: fügt eine Texteingabe für ein Datum ein. Zusätzlich wird ein „Date Picker“ eingeblendet.
- Text: fügt ein Textelement ein. Im Normalfall muss diese Funktion nicht ausdrücklich aufgerufen werden, da durch das Schreiben eines Textes in eine leere Zelle automatisch ein Textelement erzeugt wird. Es wird lediglich benötigt, wenn man ein leeres Textelement erzeugen möchte (z.B. für einen reinen Unterstrich).
- Editor: fügt ein mehrzeiliges Textelement ein. In den Eigenschaften kann die Zahl der Zeilen und Spalten definiert werden.

- **Markierung:** fügt eine oder mehrere Checkboxen in eine Zeile ein. Wenn mehrere Checkboxen benötigt werden, kann in dem Eigenschaftsfeld für den Text ein String mit allen Bezeichnern, getrennt durch ein Pipe Symbol, eingegeben werden.
- **Schalter:** fügt eine Schaltfläche für einen ELOas Aufruf oder zur Errichtung zusätzlicher Eingabezeilen ein. Es gibt die vorkonfigurierten Skripte JS_AddLine für das hinzufügen neuer Eingabezeilen und JS_Print zum Drucken des aktuellen Formulars. Weiterhin können Sie in dem Kopfdatensatz weitere eigene Skripte hinterlegen, die mit dem Namen JS_ beginnen müssen (z.B. JS_CalcDays). Diese Skriptnamen können dann auch in der Schaltfläche als Namen hinterlegt werden, das onclick Event des Buttons wird dann durch das ELOwf Laufzeitsystem automatisch mit dem Skript verbunden. Alle anderen Namen werden als ELOas Aufruf gewertet.
- **Bild:** fügt ein Bild in das Formular ein. Das Bild muss im Ordner „Images“ vorhanden sein. Es kann entweder im Text Feld ein Bildname hinterlegt werden oder im Namensfeld ein Variablenname einer Indexzeile angegeben werden, die den Namen des Bildes enthält.
- **Auswahl:** fügt einen Radiobutton ein. Genauso wie bei den Checkboxen können mehrere Radiobuttons in eine Zeile gepackt werden. Die Namen werden im Textfeld durch ein Pipe Symbol getrennt dargestellt.

7.2 Zellen verbinden und trennen

Der ELO Formularworkflow arbeitet mit einem Tabellenlayout. Jedes Teilformular besteht aus einer Anzahl von Zeilen und Spalten in einem regelmäßigen Gitter. Alleine durch diese Basisstruktur ergeben sich im Normalfall „aufgeräumte“ Formulare. Es ist kaum möglich die unterschiedlichen Eingabefelder in einem wilden Zick-Zack Muster anzugeordnen. Manchmal muss man aber von der Basisstruktur abweichen, z.B. um ein langes Feld für eine Zwischenüberschrift über die gesamte Breite zu erstellen. Das kann über die Funktion „Verbinden“ ausgeführt werden.

Die Funktion „Verbinden“ verbindet die aktuell ausgewählte Zelle mit der nächsten Zelle in dieser Zeile. Es entsteht eine neue Zelle, welche den Platz der beiden ursprünglichen Zellen einnimmt. Dieser Vorgang kann natürlich fortgesetzt werden. Wenn eine Tabelle aus 5 Spalten besteht, kann nach der Auswahl einer Zelle in der ersten Spalte und viermaliger Anwendung der Verbinden Funktion eine Zelle über die gesamte Breite erzeugt werden.

Spalte 1	Spalte 2	Spalte 3	Spalte 4	Spalte 5
Eine Spalte				
Zwei Spalten				
Drei Spalten				
Vier Spalten				
Fünf Spalten				

Die gegenteilige Operation dazu ist „Trennen“. Dieser Funktionsaufruf trennt die letzte Tabellenspalte von einem verbundenen Eintrag ab.

8 Validierung

Wenn ein Eingabefeld für numerische Werte verwendet werden soll, wird im Allgemeinen eine andere Formatierung und Gültigkeitsprüfung als bei Textfeldern erwartet. Im Feld „Validierung“ kann man deshalb den Datentyp angeben und weitere Prüfungen einstellen.

Eingabefelder können prinzipiell zwischen Numerikfeldern und Textfeldern unterscheiden. Das Feld Validierung beginnt deshalb immer mit „num“ oder „text“. Danach können optional weitere Parameter folgen, die z.B. Eingabebereiche festlegen. Alle Parameter haben die Form „name:wert“ und sind jeweils durch ein Leerzeichen getrennt.

Die Kennung num oder text muss jeweils am Anfang der Liste stehen wenn es mehrere Attribute gibt. Alle folgenden Werte können in beliebiger Reihenfolge stehen (z.B. „num min:7 max:19“).

Name	Verfügbar	Wert
min	text, num	Bei numerischen Eingaben wird über den min Parameter der kleinste zulässige Zahlenwert festgelegt. Beachten Sie bitte, dass sich bei Zahlen mit Nachkommawerten Rundungsfehler ergeben können. Bei Textfeldern gibt dieser Wert die Mindestanzahl von Zeichen an.
max	text, num, Schalter	Bei numerischen Eingaben wird über den max Parameter der größte zulässige Zahlenwert festgelegt. Beachten Sie bitte, dass sich bei Zahlen mit Nachkommawerten Rundungsfehler ergeben können. Bei Textfeldern gibt dieser Wert die Höchstzahl von Zeichen an. Bei einem JS_AddLine Schalter gibt dieser Wert die maximale Zahl von Zusatzzeilen an, die der Anwender erzeugen kann.
nk	num	Gibt die Anzahl von Nachkommastellen bei numerischen Eingaben an. Beim Verlassen des Felds wird die Eingabe nach einer Veränderung automatisch auf diese Zahl formatiert.
asname	Schalter	Wenn ein Button eine ELOas Funktion aufrufen soll, dann muss hier der Name des Rulesets angegeben werden, der ausgeführt werden soll. Optional können noch Parameter mitgegeben werden (param2 und param3).

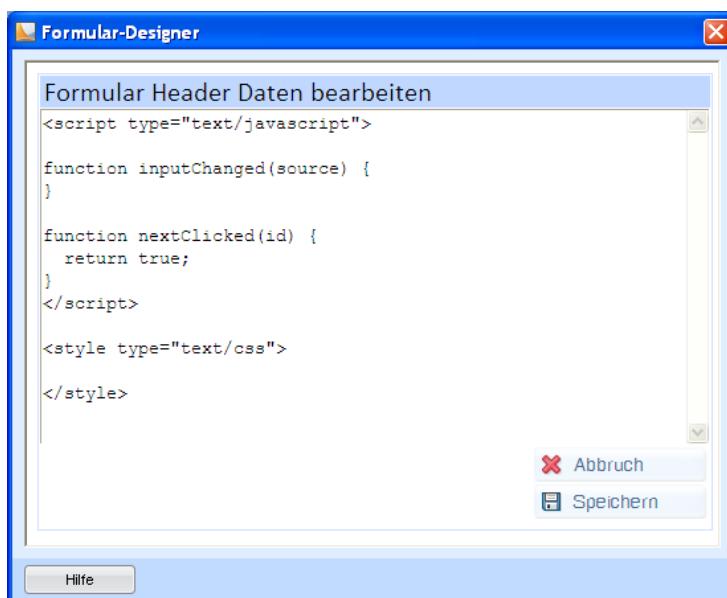
param2 param3	Schalter	Hier können Parameter für den ELOas Aufruf mitgegeben werden. Wenn der Wert mit einem Ausrufezeichen beginnt (!123), dann wird direkt dieser Wert (ohne Ausrufezeichen) übertragen. Andernfalls wird der aktuelle Wert des Eingabefelds mit diesem Namen (z.B. IX_GRP_RENUM) übertragen.
copy	Alle Felder	true oder false: dieser Wert bestimmt, ob beim Kopieren einer Eingabezeile der aktuelle Feldinhalt mit kopiert wird oder nicht. Diese Einstellung kann nicht mit dem count Parameter kombiniert werden.
count	num	auto: diese Einstellung kann für die automatische Erzeugung von Zeilennummern bei kopierten Eingabezeilen verwendet werden. Wenn das Validierungsfeld einen Eintrag count:auto enthält, dann wird in der neuen Zeile der um eins erhöhte Wert der Vorgängerzeile eingetragen. Diese Einstellung kann nicht mit dem copy Parameter kombiniert werden.
lines	Schalter	Bei einem JS_AddLine Schalter gibt dieser Parameter an, wie viele Zeilen kopiert werden (default: 1 Zeile).
add<*>	Eingabefeld	Über diese Einstellung kann bestimmt werden, dann bei einer mehrfachen Stichwortauswahl der Wert im Eingabefeld nicht überschrieben wird sondern die Eingaben hintereinander gehangen werden. Mittels add wird ein Leerzeichen als Trennzeichen verwendet. Man kann nach dem add auch eine weitere Zeichenfolge angeben, dann wird diese als Trennzeichen verwendet (Ein Unterstrich wird dabei als Leerzeichen interpretiert). add,_ fügt nach dem Vorgänger jeweils ein Komma und ein Leerzeichen an.

Numerikfelder akzeptieren als Nachkommatarenner sowohl Komma wie auch Punkt. 1,23 wird genauso akzeptiert wie 1.23. Das führt allerdings dazu, dass keine Tausendertrenner bei der Eingabe zulässig sind (1.000,23).

Datumsfelder akzeptieren das Datum sowohl mit 2-stelligen wie auch mit 4-stelligen Jahreszahlen. Bei einer 2-stelligen Eingabe werden alle Werte größer/gleich 70 als 19xx gewertet und alle anderen als 20xx (70 wird zu 1970, 11 wird zu 2011).

9 Eigene Skripte und Styles

Einfache Validierungen können durch die oben aufgeführten Einstellungen automatisch durchgeführt werden. An manchen Stellen benötigt man aber weitergehende Prüfungen. So kann man zum Beispiel in einer Maske mit Butto-, Netto- und MWSt-Betrag nachprüfen, ob die drei Eingaben zusammenpassen. Solche Aufgaben kann man per Skript erledigen. Diese Skripte werden am Einfachsten in dem Template „Kopfdaten erfassen“ eingetragen, welches beim Erzeugen eines neuen Formulars automatisch angelegt wird. Wenn Sie dieses Template auswählen, werden Sie in einen Editor geführt, der eine Bearbeitung der Vorgabe erlaubt.



Im script Bereich können beliebige eigene Skriptfunktionen hinterlegt werden. Damit diese nicht mit den Standardfunktionen kollidieren, sollten Sie eigene Funktionsnamen immer mit einem Präfix belegen (z.B. statt `readValue` `fctReadValue`).

Weiterhin finden Sie zwei vordefinierte Funktionen: `inputChanged` und `nextClicked`.

Die Funktion `inputChanged` wird jedesmal aufgerufen, wenn ein Anwender ein Eingabefeld ändert und verlässt. In diesem Fall enthält der `source` Parameter das betroffene Eingabefeld. Zudem wird bei der Anzeige des Formular auch einmal diese Funktion aufgerufen um Initialisierungen vornehmen zu können. Diesen Zeitpunkt erkennen Sie daran, dass der `source` Parameter `undefined` ist.

Die Funktion `nextClicked` wird aufgerufen, nachdem der Anwender die Workflowweiterleitung ausgewählt hat – aber bevor die Weiterleitung tatsächlich ausgeführt wurde. Sie können hier noch Kontrollen durchführen und bei Bedarf die Weiterleitung abbrechen (durch einen Returnwert: `false`).

9.1 Nützliche Hilfsfunktionen

Zur Unterstützung eigener Funktionen gibt es eine Reihe von Hilfsroutinen, die die Programmierung häufiger Aktionen vereinfachen

\$val(<name>): mit dieser Funktion können Sie den Inhalt eines Eingabefelds mit dem Namen <name> abfragen. Die Funktion liefert diesen als Text zurück.

Beispiel: var name = \$val("IX_GRP_NAME");

\$num(<name>): analog zur Funktion \$val liefert diese Funktion den Inhalt des Felds mit dem Namen <name> - in dieser Form aber als numerischen Wert und nicht als Text.

Beispiel: var mwst = \$num("IX_GRP_MWST");

\$var(<name>): zum Setzen eines Werts können Sie auch die Variable ermitteln, die das Eingabefeld selber enthält. Falls es mehrere Eingabefelder mit diesem Namen gibt, wird das erste Eingabefeld ausgewählt, welches nicht auf ReadOnly gesetzt ist.

Beispiel: \$var("IX_GRP_MWST").value = 19;

\$update(<name>, <wert>): füllt das Eingabefeld <name> mit dem angegeben Wert und ruft anschließend die Validierung auf. Diese prüft die zulässigen Eingaben, zeigt bei Bedarf eine Fehlermeldung an und formatiert die Anzeige.

formatDate(<isoDate>): erzeugt aus einem ISO Datumseintrag einen normalen Datumsstring der Form TT.MM.YYYY.

Beispiel: alert(formatDate("20110131"));

toDay(): liefert das aktuelle Tagesdatum zurück.

addErrorMsg(source, msg): zeigt eine Fehlermeldung (msg) zu einem Eingabefeld (source) an.

sumAllInput(sourcelitems, destination): diese Funktion addiert eine Spalte von Werten auf, die durch das Kopieren von Eingabezeilen entstanden sind. Diese Spalten haben immer das

Namensschema <name><counter>, also z.B. BETRAG1, BETRAG2 usw. – die Spalte heißt also BETRAG (sourceItems).

Wenn man nun sumAllInput("BETRAG", "SUMME") aufruft, sucht das System alle Eingabefelder BETRAG1, BETRAG2... zusammen, addiert diese auf und schreibt das Ergebnis in das Eingabefeld mit dem Namen „SUMME“.

getNumberOfWorkDays(startDate, endDate): diese Funktion ermittelt die Anzahl der Wochentage zwischen den beiden Terminen. Dabei kann startDate auch hinter dem endDate liegen, die Reihenfolge wird dann automatisch umgedreht (d.h. es gibt dann auch keine negativen Tageszahlen). Beachten Sie bitte, dass diese Funktion keine Feiertage berücksichtigt. Das wäre in der Praxis enorm aufwändig, da neben den festen Feiertagen nicht nur die beweglichen Feiertage berechnet werden müssten, sondern zusätzlich auch noch Bundesland- und Länderabhängige Feiertagsregelungen beachtet werden müssten.

inputBox(): diese Funktion steht für Anwendereingaben zur Verfügung. Sie kann in eigenen Skripten durch den Aufruf „inputBox(<Titel>, <Nachricht>, <Breite>, <Ok Funktion>)“ verwendet werden.

Beispiel:

```
function JS_test(source) {  
    inputBox("Andere Bemerkung eingeben", "Geben Sie hier ... ein", 300, "JS_update");  
}
```

Wenn der Anwender auf Ok klickt, dann wird die Ok Funktion aufgerufen. Sie besitzt einen Parameter, den vom Anwender eingegebenen Text.

```
function JS_update(text) {  
    $update("IX_GRP_BIMODELL", text);  
}
```



getInput(source): diese Funktion wird im onclick Event eines Schalters aufgerufen und liefert als Ergebnis den aufrufenden Button zurück. Unter Firefox wird dieser aus dem source-Event gewonnen, unter IE aus window.event.

9.2 Eigene Styles

In dem Feld „Darstellung“ kann festgelegt werden, welche HTML Style Definitionen zur Anzeige des Eingabefeldes verwendet werden. Es gibt eine Reihe von Standarddefinitionen, die über die Stichwortliste erreicht werden. Sie können aber beliebige weitere Styles eintragen. Damit diese dem System bekannt sind, können Sie sie im style Abschnitt der Kopfdaten hinterlegen.

Tipp: im Zusatztext des Ordners „Classes“ können Sie diese auch in die Liste der Standardstyles aufnehmen. Diese sind dann über die Stichwortliste automatisch auswählbar.

9.3 Eigene Bilder

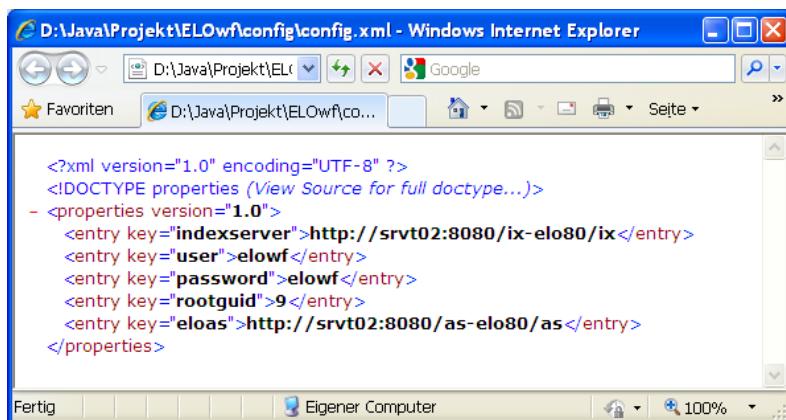
Im Ordner Images können Sie eigene Bilder hinterlegen, die dann vom Formularfeldtyp „Bild“ verwendet werden können. Die Bilder, die dort hinterlegt werden, kopiert das System beim Start in den images Ordner der Webapplikation, so dass sie von dem Browser aus erreichbar sind.

9.4 Eigene Web-Seiten, Skript- oder Style-Dateien

Alle Dokumente, die Sie im Ordner „Webapp“ hinterlegen, werden beim Systemstart in das Webapps Verzeichnis des Workflowformulars gespeichert. Hier können Sie also zusätzliche Skriptdateien oder Styledateien hinterlegen und diese dann in den Formularen (bzw. in den Kopfdaten) verwenden.

10 ELOwf Struktur im Archiv

Genauso wie bei den ELO Automation Services liegt in der Konfiguration des ELO Formular Workflows nur die minimale Information zum Aufbau der Verbindung. Die restliche Konfiguration und alle Formulardaten liegen im ELO Archiv. Deshalb lassen sich Workflowformulare relativ einfach mit dem normalen ELO Export/ Import von einem Archiv in ein anderes Archiv übertragen.



The screenshot shows a Windows Internet Explorer window displaying the contents of a config.xml file. The URL in the address bar is "D:\Java\Projekt\ELOWf\config\config.xml". The page content is an XML document with the following structure:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE properties (View Source for full doctype...)>
- <properties version="1.0">
  <entry key="indexserver">http://srvt02:8080/ix-elo80/ix</entry>
  <entry key="user">elowf</entry>
  <entry key="password">elowf</entry>
  <entry key="rootguid">9</entry>
  <entry key="eloas">http://srvt02:8080/as-elo80/as</entry>
</properties>
```

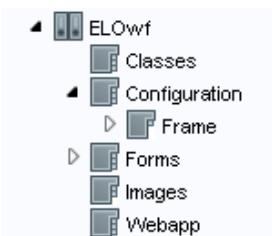
10.1 Einträge in der config.xml Datei

Name	Inhalt
indexserver	URL des Indexservers zum Archiv.
user	Anmeldename des Workflowformular Dienstes am Indexserver. Dieses Konto wird zum Einlesen der Konfiguration und der Formulare benötigt. Die spätere Bearbeitung der Formulareingaben erfolgt jeweils unter dem Konto des Benutzers damit die Berechtigungskontrollen greifen und der Report richtig geschrieben wird.
password	Passwort zur IX Anmeldung.
rootguid	GUID oder ELO Objektid zum Startordner der Workflowformular Konfiguration.

eloas	URL zum ELOas Server für dieses Workflowformular. Achtung: diese URL muss auf dem gleichen Server liegen wie der Formularworkflow. Andernfalls wird der Aufruf durch die in modernen Browern üblichen Security Einstellungen unterbunden. Im ELOprofessional Umfeld ist das immer der Fall. Unter ELOenterprise können prinzipiell alle Serverkomponenten auf unterschiedlichen Servern liegen. In diesem Fall muss dann auf dem ELOWf Server zusätzlich ein eigener lokaler ELOas installiert werden.
-------	--

10.2 Struktur im Archiv

Das ELOWf Verzeichnis ist der Startordner, der durch den config Parameter rootguid adressiert wird. Alle anderen Ordner liegen direkt unterhalb dieses Startordners und müssen deshalb nicht speziell konfiguriert werden. Allerdings sind die Namen vorgeschrieben und müssen exakt so eingetragen werden. Andernfalls werden die Ordner nicht gefunden.



Im Zusatztext des Classes Ordners liegen die HTML css Stylenamen für die Darstellung. Diese Liste kann beliebig erweitert werden, es ist aber darauf zu achten, dass die Styles dann auch wirklich verfügbar sind (entweder durch eine Erweiterung der css Datei, durch eine eigene css Datei oder durch Eintrag in die Kopfdaten).

formheader=Überschrift
formbold=Fett
formtiny=Klein
formwarn=Wichtig
formgreen=Grün
fromred=Rot
formyellow=Gelb
formblue=Blau
formright=Rechtsbündig
formline=Einfache Linie
formsolidline=Kräftige Linie
formlarge=Großer Abstand
formtop=Oben ausrichten

Im Zusatztext des Configuration Ordners liegen die Pfade zum Temp-Ordner und zum Webapp Verzeichnis.

```
webappdir=C:\\Programme\\Apache Software Foundation\\Tomcat 6.0\\webapps\\wf-elo80  
temp=C:\\temp
```

Zudem enthält dieser Ordner als Unterordner das Standard HTML Dokument als Rahmen für alle Formulare. Sie können für einzelne Formulare bei Bedarf ein eigenes spezielles Frame Dokument erstellen, per Default wird aber dieses als Standard für neue Formulare verwendet.

Im Forms Ordner legt der Formulareditor Unterordner mit den Formularen und Templates an. Hier müssen Sie im Normalfall nicht manuell eingreifen. Lediglich, wenn Sie ein Formular oder Template umbenennen wollen. Solange es im Formulareditor dazu noch keine Option gibt, können Sie es direkt im Archiv ausführen. Nach einer Änderung müssen Sie im Formulareditor noch auf Reload klicken damit die gespeicherten Formulare aus dem Cache neu eingelesen werden.

Im Images Ordner können Sie eigene Bilddateien hinterlegen, die dann im Formularelement „Bild“ verwendet werden können.

Im Webapp Ordner können Sie eigene Skript-, css- oder html Dateien hinterlegen. Diese werden dann beim Programmstart oder beim Reload in das webapp Verzeichnis des Formularworkflows kopiert und können dann vom Formular aus verwendet werden.