

# [e] forms & more

## Produktgeschichte

Erweiterungen  
Verbesserungen

Version 7.5.7

09.01.2018

## Version 7

### Build 7.5.7 (09.01.2018)

#### Attribute *id* und *name* für RTF-Objekte beim HTML-Export

RTF-Objekte werden beim Export jetzt auch mit den Attributen *id* und *name* ausgestattet.

### Build 7.5.6 (05.01.2018)

#### Konvertieren von Inhalten beim Umwandeln von Objekten

Beim Umwandeln von Text-Objekten in RTF-Objekte wird jetzt der Inhalt konvertiert, so dass der bisherige Text im RTF-Objekt eingetragen wird.

Beim Umwandeln von RTF-Objekten in Text-Objekte wird ebenfalls der Inhalt konvertiert, so dass der RTF-Text jetzt als purer Text im text-Objekt eingetragen wird.

#### Attribut *name* für Textobjekte kann wahlweise erzeugt werden

Es wurde eine neue Option in den HTML-Export-Einstellungen implementiert, mit der eingestellt werden kann, ob die `<div>` Tags, die Textobjekte repräsentieren mit einem Attribut `name="name"` versehen werden, falls ein Name vorhanden ist.

### Build 7.5.5 (16.12.2017)

#### Firefox Workaround für den Export von Objektgruppen nach HTML

Beim Export von Objektgruppen nach HTML addiert der Firefox Browser einen Offset in Höhe des Fonts der Legende auf die vertikale Position aller inneren Objekte. Andere Browser tun das nicht. Es wurde ein Javascript Workaround implementiert, der dies ausgleicht.

### Build 7.5.4 (13.12.2017)

#### Verbesserungen beim Export von Objektgruppen nach HTML

Beim Export von Objektgruppen nach HTML hängt die Größe des erzeugten `<fieldset>` Elements nichtmehr vom Default-Padding ab. Die Position der Objekte innerhalb der Gruppe hängt nicht mehr von der Schriftgröße der Legende ab.

### Begrenzung der Zeichenzahl in nach HTML exportierten Eingabeobjekten

Beim Export von einzeiligen Eingabeobjekten nach HTML werden jetzt automatisch Javascripte generiert, die verhindern, dass mehr Zeichen eingegeben werden können, als in das Feld hineinpassen, falls das horizontale Rollen deaktiviert ist.

### Begrenzung der Zeilenzahl in nach HTML exportierten Eingabeobjekten

Beim Export von mehrzeiligen Eingabeobjekten nach HTML werden jetzt automatisch Javascripte generiert, die verhindern, dass mehr Zeilen eingegeben werden können, als in das Feld hineinpassen, falls das vertikale Rollen deaktiviert ist.

## Build 7.5.3 (12.12.2017)

### Beim HTML-Export wird Attribut *name* in Seiten und Zonen durch Attribut *id* ersetzt

Beim HTML-Export erhalten die <div> Tags von Seiten und Zonen nicht mehr das Attribut *Name* sondern nunmehr das Attribut *id*. Alle Javascripte, die sich darauf beziehen wurden angepasst.

## Build 7.5.2 (10.12.2017)

### HTML Export für Ankreuzfelder vollständig erneuert

Der HTML-Export von Ankreuzfeldern wurde vollständig erneuert. Insbesondere werden Label und Eingabeelement nicht mehr ineinander verschachtelt. Dies wurde zwar von allen Browsern korrekt behandelt, entsprach aber nicht dem HTML5-Standard.

Um die Kompatibilität zu bestehenden Formularen nicht zu gefährden, wurde in den HTML-Export-Einstellungen eine Option implementiert, mit der man wahlweise auch den vormals generierten Code erzeugen kann.

## Build 7.5.1 (08.12.2017)

### HTML Export erzeugt ID-Attribute für Ausgabeobjekte

Der HTML Export erzeugt für benannte Ausgabeobjekte ID-Attribute, die den Namen als ID einbetten.

### HTML Export exportiert Ausrichtungen in RTF-Texten

Der HTML Export exportiert jetzt Ausrichtungen (links, rechts, mittig, Blocksatz) in RTF-Texten.

### HTML Export unterdrückt *Placeholder*-Attribute bei unsichtbaren Eingabefeldern

Der HTML Export unterdrückt die Ausgabe von *Placeholder-Attributen* bei Eingabefeldern, deren Anzeige durch den Style *hidden* unterdrückt wird.

## Build 7.5.0 (02.12.2017)

### HTML Export erfolgt jetzt als HTML5

Der HTML Export erfolgt jetzt nach dem Standard für HTML5. Die Ergebnisdatei beginnt mit der Zeile:

```
<!DOCTYPE HTML>
```

### Positionierung von RTF-Texten in HTML5

RTF-Texte wurden in HTML-5 anders positioniert als in HTML4. Dafür wurde ein Workaround implementiert.

### Aktiver TAB in den HTML-Export-Einstellungen wird gespeichert

Der Editor speichert das aktive TAB in den HTML-Export-Einstellungen und aktiviert dieses automatisch wieder, wenn die Einstellungen erneut aufgerufen werden.

### Aktiver TAB in den PDF-Export-Einstellungen wird gespeichert

Der Editor speichert das aktive TAB in den PDF-Export-Einstellungen und aktiviert dieses automatisch wieder, wenn die Einstellungen erneut aufgerufen werden.

### Änderungen in globalen Java-Scripts werden gespeichert

Wenn im Menü *Skripte* Änderungen in den globalen Java-Skripten gemacht wurden, wurde der Indikator, dass das Formular verändert wurde nicht gesetzt und die Änderungen gingen verloren, wenn das Formular anschließend nicht gespeichert wurde.

Der Fehler wurde behoben.

### Seitenzonen transparent darstellen

Seitenzonen können jetzt auch während der Bearbeitung im Editor transparent dargestellt werden. Die Transparent kann in den Eigenschaften ein- oder ausgeschaltet werden.

### Bibliothek für Reguläre Ausdrücke wurde verbessert

Die Bibliothek für Reguläre Ausdrücke wurde verbessert. Insbesondere wurde der Reguläre Ausdruck zur Erkennung von Webadressen angepasst.

### Erweiterung der Eigenschaften für Objektgruppen

Die Eigenschaften "Nicht nach HTML exportieren" und "Nicht nach PDF exportieren" wurden in den Dialog zur Änderung der Eigenschaften mehrerer Objekte aufgenommen und ermöglichen nunmehr, deren Werte für mehrere Objekte gleichzeitig zu ändern.

### Standardwert für die Auflösung beim PDF-Export wird dauerhaft gespeichert

Der Wert für die Auflösung von PDF-Dateien (bisher immer 150) kann jetzt konfiguriert werden. Das System merkt sich immer den zuletzt eingestellten Wert und verwendet diesen automatisch für neue Formulare.

### Benutzerdefinierte CSS Klassen und Styles für Textobjekte

Für Textobjekte können jetzt benutzerdefinierte CSS-Klassen und CSS-Styles definiert werden, die beim Export nach HTML zusätzlich zu den automatisch generierten Werten eingetragen werden.

### Benutzerdefinierte CSS Klassen und Styles für Buttons

Für Button-Objekte können jetzt benutzerdefinierte CSS-Klassen und CSS-Styles definiert werden, die beim Export nach HTML zusätzlich zu den automatisch generierten Werten eingetragen werden.

### Relative CSS-Datei-Pfade in den Export-Einstellungen für HTML

Die CSS-Datei, die in den Einstellungen für den HTML-Export zur Einbettung vorgegeben werden kann, kann jetzt wahlweise mit einem absoluten Pfad oder mit einem relativen Pfad versehen werden. Relative Pfade beziehen sich auf das Verzeichnis des Formulars.

### Automatische Anzeige des Ergebnisses beim HTML-Export

Die Automatische Anzeige, die nach dem Export nach HTML erfolgt, wenn dies so konfiguriert ist, kann jetzt wahlweise in einem der installierten Browser oder mittels einer frei konfigurierbaren Anwendung erfolgen.

### Export der automatischen Großschreibung nach HTML und PDF verbessert

Textobjekte, für die das Attribut *Großschreibung* aktiviert ist, werden jetzt als Großbuchstaben nach HTML und PDF exportiert. Eingabefelder, für die das Attribut *Großschreibung* aktiviert ist, erhalten im HTML-Export jetzt auch das entsprechende Attribut.

## Build 7.4.14 (24.11.2017)

### Änderung der CSS-Styles beim Export von Eingabegruppen nach HTML

Eingabegruppen wurden bisher beim Export nach HTML mit dem Style *padding:0px* versehen. Dies führte beim Firefox Browser zu einer fehlerhaften Positionierung. Der Style wird jetzt nicht mehr erzeugt.

Sollte der Style dennoch benötigt werden, so kann er im Feld *CSS-Erweiterung* der Objekteigenschaften manuell vorgegeben werden.

### Benutzerdefinierte CSS-Styles für Eingabegruppen wurden nicht erzeugt

In den Eigenschaften von Eingabegruppen können im Feld *CSS-Erweiterung* benutzerdefinierte Styles vorgegeben werden. Diese wurden bisher nicht erzeugt. Der Fehler wurde behoben.

### Schreibfehler korrigiert

In der Eingabemaske *Responsive* der HTML-Einstellungen wurden zwei Schreibfehler korrigiert.

## Build 7.4.13 (21.11.2017)

### Eingabegruppen werden nach PDF exportiert

Die mit Version 7.4.6. eingeführten Eingabegruppen werden ab sofort auch nach PDF exportiert.

## Build 7.4.12 (06.11.2017)

### Option zum Exportieren der Body-Schrift beim HTML-Export

Es wurde in den HTML-Einstellungen für die Webseite eine Option implementiert, mit der man wählen kann, ob die Basis-Schriftart als Style in den `<body>`-Tag exportiert wird oder nicht.

### Seitennummer werden beim HTML-Export als Attribut hinzugefügt

Beim HTML-Export wird ab sofort die Nummer der Seite als Attribut *data-pagenr* in das `<div>`, das die Seite repräsentiert, exportiert.

### Blockatz bei Textobjekten wird nicht gespeichert

Textobjekte mit der horizontalen Ausrichtung *Blockatz* wurden mit der Ausrichtung *linksbündig* gespeichert. Der Fehler wurde behoben.

### Mehrzeilige Labels bei Ankreuzfeldern werden beim Export zu breit

Die Labels von Ankreuzfeldern wurden beim Export zu breit, wenn sie mehrzeilig ausgelegt waren. Der Fehler wurde behoben.

### Horizontale zentrierte Ausrichtung geht bei explizitem Zeilenabstand verloren

Textobjekte mit einer zentrierten horizontalen Ausrichtung wurden nicht korrekt dargestellt, wenn der Zeilenabstand explizit vorgegeben war. Der Fehler wurde behoben.

### Kein Attribut *aria-required* beim Export von Dateifeldern nach HTML

Eingabefelder für Dateinamen erhalten beim Export nach HTML kein Attribute *aria-required*, auch nicht dann, wenn sie Pflichtfelder sind.

## Build 7.4.11 (20.09.2017)

### Filler öffnet Formulare im ReadOnly Modus

Der Filler öffnet Formulare ab sofort im ReadOnly-Modus. Das bedeutet, dass es keine Sharing-Probleme mehr gibt, wenn mehrere Filler gleichzeitig auf dasselbe Formular zugreifen.

### Sendeformate in HTML werden beim Verlassen eines Feldes erzeugt

Wenn im HTML-Export Sendeformate für formatbehaftete Eingabefelder (Datums, Zahl, etc.) definiert sind, werden die assoziierten versteckten Felder, die zum Übertragen der formatierten Eingaben automatisch generiert werden, jetzt bereits beim Ereignis *OnBlur* gefüllt und nicht erst beim Ereignis *OnSubmit*, wie zuvor.

## Build 7.4.10 (05.09.2017)

### Infield Labels für Comboboxen beim Export nach HTML mit Bootstrap

Für Auswahlfelder wurde beim Export nach HTML mit Bootstrap-Unterstützung jetzt auch die Option implementiert, sogenannte *Infield-Labels* zu generieren. Hier befindet sich das Label des Auswahlfeldes innerhalb seiner Umrandung.

### **Infield Errors für Comboboxen beim Export nach HTML mit Bootstrap**

Für Auswahlfelder wurden beim Export nach HTML mit Bootstrap-Unterstützung jetzt auch sogenannte *Infield-Errors* implementiert. Dabei erfolgt die Anzeige von Fehlern immer direkt unterhalb des Auswahlfeldes.

### **Glyphicons für Comboboxen beim Export nach HTML mit Bootstrap**

Für Auswahlfelder werden beim Export nach HTML mit Bootstrap-Unterstützung jetzt auch Glyphicons in Labels generiert.

### **Das automatische Löschen von Fehlern bei der HTML-Eingabe ist konfigurierbar**

Beim Export eines Formulars nach HTML mit Bootstrap können automatisch erzeugte Fehlermeldungen wahlweise als sogenannter Infield-Fehler direkt unter dem Eingabefeld ausgegeben werden. Diese Fehlermeldungen wurden bisher automatisch gelöscht, wenn die Eingabe in Ordnung ist. Dieses löschen kann jetzt in den HTML-Exporteinstellungen ein- oder ausgeschaltet werden, damit eigene Fehlermeldungen ausgegeben werden können, ohne dass diese automatisch entfernt werden.

### **Fehler löschen nach Korrektur in HTML**

Infield-Fehlermeldungen wurden zum Teil nicht gelöscht, auch wenn die Eingabe korrigiert wurde. Der Fehler wurde behoben.

## **Build 7.4.9** (22.08.2017)

### **Konfigurierbare Basisschriftart der Webseite beim HTML-Export**

Die Basisschriftart der Webseite, die als CSS im <body>-tag eingetragen wird, ist jetzt in den HTML-Einstellungen konfigurierbar.

### **Benutzerdefinierte CSS-Styles für die Webseite beim HTML-Export**

Es können in den HTML-Einstellungen jetzt benutzerdefinierte CSS-Styles für den <body>-Tag der Webseite definiert werden.

### **Vertikaler Offset der Box zum Label beim HTML-Export mit Bootstrap**

Der vertikale Offset zwischen Ankreuzfeldern und ihren Labels wird jetzt auch beim Export mit Bootstrap-Unterstützung ausgewertet



## Build 7.4.8 (20.08.2017)

### Eingabeprüfungen in HTML Formularen

Formulare, die nach HTML exportiert werden, generieren eine auf Javascript basierende Prüfung aller Eingaben, die vor dem Absenden automatisch aufgerufen wird. Die Prüfung basierte bisher auf der Reihenfolge in denen die Eingabefelder exportiert wurden. Die Prüfung wurde nun umgestellt und erfolgt nun auf Basis der IDs der Eingabeobjekte. Dazu muss die Option aktiviert sein, dass Eingabeobjekte automatisch eindeutig benannt werden.

## Build 7.4.7 (16.08.2017)

### Export von Eingabegruppen nach Bootstrap

Um den Bootstrap-Export von Eingabegruppen besser gestalten zu können, wurde innerhalb des `<fieldset>`-Tags, der die Eingabegruppe in HTML abbildet, ein weiterer `<div>`-Tag eingebettet, in dem sich alle Objekte der Eingabegruppe befinden.

Ferner wurden in der Style-Datei *eforms.css* zwei neue Klassen implementiert:

- |                                 |   |
|---------------------------------|---|
| <code>fieldset-inner-bs</code>  | Wird dem <code>&lt;div&gt;</code> zugeordnet, in dem sich innerhalb des <code>&lt;fieldset&gt;</code> alle Objekte befinden |
| <code>fieldset-legend-bs</code> | Wird dem <code>&lt;legend&gt;</code> -Tag des <code>&lt;fieldset&gt;</code> zugeordnet                                      |

Außerdem wird das `<legend>` Element immer ohne Rand dargestellt. Dafür erhält es automatisch den Hintergrund der Eingabegruppe, sofern die Option gewählt ist, dass dieser nach Bootstrap exportiert wird.

## Build 7.4.6 (15.08.2017)

### Neues Objekt *Eingabegruppe*

Es wurde ein neues Objekt mit der Bezeichnung *Eingabegruppe* zum optischen Gruppieren von Eingabeobjekten implementiert. Das Objekt ist ein Ausgabeobjekt und kann beliebig oft angelegt werden.

Das Objekt hat einen Titel, der vertikal mittig im oberen Rand eingeblendet wird, ähnlich wie die Gruppen in Dialogen. Der Titel befindet sich vollständig in der Objekt-Outline, so dass der sichtbare obere Rand nicht dem oberen Rand des Objekts entspricht.

Beim Export nach HTML werden Eingabegruppen als Tag `<fieldset>` abgebildet. Der Titel wird als Tag `<legend>` eingebettet. Alle Objekte, die sich von der Position her innerhalb der Eingabegruppe befinden, werden in die Klammer `<fieldset>...</fieldset>` eingebettet.

Wenn der Export ohne Bootstrap-Unterstützung erfolgt, wird das `<fieldset>` Element explizit positioniert.

Wenn der Export ohne Bootstrap-Unterstützung erfolgt, unterbleibt die Positionierung. Eingabegruppen können beim HTML-Export nicht innerhalb von Zonen verwendet werden.

## Build 7.4.5 (11.08.2017)

### Verbesserungen im HTML-Export zur Einhaltung des W3C-Standards für HTML-5

#### **<meta>**

Der Tag `<meta name="date">` wird nicht mehr generiert. Anstelle dessen wird das Datum im Kommentar des `<HTML>` Tags eingetragen.

Der Tag `<meta name="title">` wird nicht mehr generiert.

Folgende `<meta>`-Tags werden jetzt zusätzlich erzeugt:

- `<meta http-equiv="content-style-type" content="text/css">`
- `<meta http-equiv="content-script-type" content="text/javascript">`

#### **<form>**

Der `<form>` Tag wird jetzt immer mit dem Attribut `action="#"` versehen.

Dafür wird das Attribut `role="form"` nicht mehr generiert, weil dies der Default ist und der Standard davon abrät, Default-Werte explizit zu setzen.

#### **Buttons zum Absenden**

Beim HTML-Export werden aus Buttons mit der Aktion *Formular Absenden* wie bisher Elemente der Form `<button type='submit'>` erzeugt.

Im Gegensatz zu früheren Versionen wird aber beim Klicken auch das für HTML hinterlegte Javascript ausgeführt.

#### **<title>**

Der Tag `<title>` im HTML-Header wird jetzt in jedem Fall mit Inhalt gefüllt. Wenn kein Titel in den HTML-Export-Einstellungen definiert ist, wird der Titel *Formular* verwendet.

#### **<script>**

`<script>`-Tags erhalten anstelle des Attributs `Language="Javascript"` das Attribut `type="text/javascript"`..

**Attribut name**

Das Attribute *name*="..." wird künftig für die die Tags `<img>` und `<div>` nicht mehr generiert, da dies nicht konform mit dem Standard ist.

**Automatische Erzeugung von Überschriften `<h1>` .. `<h6>`**

Texte werden beim HTML-Export mit Bootstrap-Unterstützung nicht mehr automatisch in Abhängigkeit von der Schriftgröße mit den Tags `<h1>` .. `<h6>` geklammert. Sie werden immer als `<div>` ausgegeben.

`<h1>` .. `<h6>` wird nur erzeugt, wenn der Text mit einem entsprechenden Barriere-Tag gekennzeichnet ist.

**`<hr>`**

`<hr>`-Tags werden künftig nicht mehr mit `</hr>` geschlossen, da dies nicht dem Standard entspricht.

**Attribut *aria-required* für Eingabefelder**

`<input>`-Tags erhalten das Attribut *aria-required*="true", wenn das Eingabefeld ein Pflichtfeld ist, das nicht leer sein darf.

**Keine schleppenden Leerzeichen in Attributen**

Alle Attribute werden grundsätzlich ohne schleppende Leerzeichen erzeugt, da dies nicht dem Standard entspricht.

**Attribute für das ein- und Ausblenden von Seiten wurden umbenannt**

Die Attribute, die an `<div>` Tags angeheftet werden, die eine Seite repräsentieren, um via Javascript das Ein- und Ausblenden von Seiten zu ermöglichen, wurden umbenannt, da laut dem HTML-5-Standard benutzerdefinierte Attribute grundsätzlich mit dem Prefix "*data-*" beginnen müssen.

**Die HTML-Attribute für das ein- und Ausblenden von Zonen wurden umbenannt**

Die Attribute, die an `<div>` Tags angeheftet werden, die eine Zone repräsentieren, um via Javascript das Ein- und Ausblenden von Zonen zu ermöglichen, wurden umbenannt, da laut dem HTML-5-Standard benutzerdefinierte Attribute grundsätzlich mit dem Prefix "*data-*" beginnen müssen.

**Erweiterungen beim Export nach HTML mit Bootstrap-Unterstützung****Schriftart der Labels von Ankreuzfeldern wird exportiert**

Beim Export von Ankreuzfeldern nach HTML mit Bootstrap-Unterstützung wird die Schriftart exportiert, wenn die entsprechende Option eingeschaltet ist.

### Attribut *aria-describedby* für Eingabefeldern mit Inline-Fehleranzeige

Beim Export von Eingabefeldern nach HTML mit Bootstrap-Unterstützung für die die Option *Inline-Fehleranzeige* aktiviert ist, wird ein Attribut *aria-describedby* generiert, das auf die ID des Elements mit dem Fehlertext verweist.

## Build 7.4.4 (09.08.2017)

### Datumsformate für mehrere Objekte gleichzeitig ändern

Für Datumsobjekte bzw. Eingabeobjekte, die den Typ *Datum* haben, kann das Format ab sofort auch für mehrere Objekte gleichzeitig geändert werden. Eine entsprechende Funktion wurde im Menü *Objekt* im Untermenü *Formate* implementiert. Eingabeobjekte mit einem anderen Datentyp oder Objekte, die nicht über Datumsformate verfügen, werden nicht beeinflusst.

### Uhrzeitformate für mehrere Objekte gleichzeitig ändern

Für Uhrzeitobjekte bzw. Eingabeobjekte, die den Typ *Uhrzeit* haben, kann das Format ab sofort auch für mehrere Objekte gleichzeitig geändert werden. Eine entsprechende Funktion wurde im Menü *Objekt* im Untermenü *Formate* implementiert. Eingabeobjekte mit einem anderen Datentyp oder Objekte, die nicht über Uhrzeitformate verfügen, werden nicht beeinflusst.

### Ganzzahl- bzw. Gleitkommazahlformate für mehrere Objekte gleichzeitig ändern

Für Eingabeobjekte, die den Typ *Ganze Zahl* bzw. *Gleitkommazahl* haben, kann das Format ab sofort auch für mehrere Objekte gleichzeitig geändert werden. Eine entsprechende Funktion wurde im Menü *Objekt* im Untermenü *Formate* implementiert. Eingabeobjekte mit einem anderen Datentyp werden nicht beeinflusst.

### Eingabeprüfung für mehrere Objekte gleichzeitig ändern

Für Eingabeobjekte, die über eine Eingabeprüfung verfügen, kann diese ab sofort mittels der Funktion *Eingabeprüfung* im Menü *Objekt* für mehrere Objekte gleichzeitig verändert werden. Objekte, die nicht über eine Eingabeprüfung verfügen, werden dabei nicht beeinflusst.

### Datentyp mehrerer Objekte gleichzeitig ändern

Für Eingabeobjekte, die über einen Datentyp verfügen (*Eingabefelder* und *Auswahlfelder*) kann der Datentyp in der Funktion *Eigenschaften* im Menü *Objekt* ab sofort auch für mehrere Objekte gleichzeitig verändert werden.

### Vertikaler Offset der Box zum Label beim HTML-Export von Ankreuzfeldern

Es kann ab sofort ein vertikaler Offset zwischen der Box und dem Label für Ankreuzfelder definiert werden, der beim HTML-Export berücksichtigt wird.

Der Offset kann sowohl für das gesamte Formular in den HTML-Export-Einstellungen definiert werden, als auch individuell für jedes einzelne Ankreuzfeld.

Wenn beide Werte definiert sind, werden sie addiert. Die Angabe erfolgt stets in Pixeln. Positive Werte verschieben die Box nach unten und negative Werte nach oben.

## Build 7.4.3 (05.08.2017)

### Export-Unterdrückung von Objekten wirkt nicht beim Export nach HTML mit Bootstrap

Die Eigenschaft *Nicht nach HTML exportieren* wurde beim Export nach HTML mit Bootstrap-Unterstützung bei manchen Objekten nicht ausgewertet. Somit wurden auch solche Objekte exportiert. Der Fehler wurde behoben.

### Vorbesetzung mehrzeiliger Eingabefelder beim Export nach HTML mit Bootstrap

Die Vorbesetzung mehrzeiliger Eingabefelder wurde nicht korrekt nach HTML exportiert, wenn die Bootstrap-Unterstützung verwendet wird. Der Fehler wurde behoben.

## Build 7.4.2 (02.07..2017)

### Attribut für die Landessprache beim HTML Export

Das Attribut lang='de' wird jetzt im <html>-Tag anstatt im <body>-Tag eingebracht.

### Barriere-Tag beim HTML Export

Texte werden beim HTML-Export als <h1> ... <h6> anstatt als <div> ausgegeben, wenn der Barriere-Tag des Textobjekts den Wert *Überschriftsebene x* hat.

## Build 7.4.1 (30.06.2017)

### Export von Schriften nach HTML in Prozent

Die Darstellung der Schriftgröße in Prozent beim Export nach HTML ist nun auch beim Export von RTF-Objekten implementiert.

### Anzeige des Zeilenabstandes bei mehrzeiligen Ankreuzfeldern im Editor

Wenn bei mehrzeiligen Ankreuzfeldern ein expliziter Zeilenabstand für den Export nach HTML eingestellt ist, wird dieser jetzt auch im Editor bei der Anzeige berücksichtigt.

## Build 7.4.0 (27.06.2017)

### Export von Schriften nach HTML in Prozent

Beim Exportieren nach HTML wurde eine Option implementiert, die es erlaubt, Schriftgrößen als Prozentangabe (basierend auf einer 12-Punkt-Basisschriftart) zu exportieren, anstatt wie bisher in Punkten.

### Attribut für die Landessprache beim HTML Export

Beim Exportieren nach HTML wird das Attribut für die Landessprache im <body>-Tag hinterlegt. Momentan wird immer die Sprache *Deutsch* hinterlegt (lang='de').

### Basisschriftart beim HTML Export

Beim Exportieren nach HTML wird im <body>-Tag eine Basisschriftart- und Größe hinterlegt. Darauf beziehen sich unter anderem alle prozentualen Schriftgrößen im gesamten Formular.

## Build 7.3.10 (16.06.2017)

### Fehler beim Export von RTF-Texten nach PDF

Beim Exportieren von RTF-Texten nach PDF trat ein Fehler auf, wenn bestimmte Fonts nicht installiert waren. Der Fehler wurde behoben.

## Build 7.3.9 (07.06.2017)

### Zeilenabstand für Ankreuzfelder mit mehrzeiligen Labels beim Export

Für den Export von Ankreuzfeldern mit mehrzeiligen Labels nach HTML oder PDF kann jetzt ein Zeilenabstand (in Punkten) vorgegeben werden. Der Zeilenabstand ist nur beim Export wirksam.

### Datumpicker im HTML-Export beginnt jetzt mit Montag

Der Datumpicker in einem nach HTML exportierten Formular beginnt jetzt mit Montag anstatt wie bisher mit Sonntag.

### Darstellung transparenter Bilder im PNG-Format

In der Bildverarbeitung werden jetzt Bilder im PNG-Format, die einen transparenten Hintergrund haben, besser verarbeitet, als bisher. Die transparenten Bildpunkte werden bei der Darstellung automatisch mit der im Bild hinterlegten Hintergrundfarbe des Bildes gefüllt. Bei Bildern, für die keine Hintergrundfarbe definiert ist, wird als Hintergrund das aus Photoshop bekannte Schachbrettmuster dargestellt, mit dem die Transparenz visuell dargestellt wird.

### Build 7.3.8 (12.05.2017)

#### Ankreuzfelder mit mehrzeiligen Labels

Ankreuzfelder können jetzt mit mehrzeiligen Labels versehen werden.

### Build 7.3.7 (10.05.2017)

#### Geometrisches Sortieren von Objekten beim HTML-Export

Der HTML-Export wurde um eine Option erweitert, die das Sortieren der exportierten Objekte nach ihren geometrischen Positionen ermöglicht. Dabei werden Objekte, die gruppiert sind innerhalb ihrer Gruppe sortiert.

### Build 7.3.6 (08.05.2017)

#### Inline Fehleranzeige beim Ausfüllen in HTML

Beim Exportieren von Formularen nach HTML mit Bootstrap-Unterstützung wurde ein neuer Modus implementiert, der es ermöglicht, Fehler beim Ausfüllen direkt unterhalb des Eingabefeldes anzuzeigen, in dem die fehlerhafte Eingabe steht.

Die Fehleranzeige wird automatisch entfernt, wenn der Fehler korrigiert und das Eingabefeld verlassen wird.

Die Option wurde in den HTML-Exporteinstellungen im Reiter *Responsive* implementiert und hat die Beschriftung *Eingabefehler unterhalb des Eingabefeldes anzeigen*.

#### Nach Fehleranzeige beim Ausfüllen in HTML Sprung ins fehlerhafte Eingabefeld

Beim Ausfüllen von Formularen, die nach HTML exportiert wurden, wird der Cursor nach der Anzeige eines Fehlers zurück in das fehlerhafte Eingabefeld gestellt.

### Konfigurierbares ALT-Attribut beim Export von Bildern nach HTML

Beim Exportieren von Bildern nach HTML wurde die Möglichkeit der Hinterlegung eines ALT-Textes implementiert. Wenn ein Text hinterlegt ist, wird er als Attribut ALT in das exportierte Bild geschrieben.

### Benutzerspezifische CSS-Klassen und CSS-Styles für HTML Hintergrundbilder

Beim Exportieren von Formularen nach HTML, für die im Reiter *Webseite* in den Export - Einstellungen nach HTML ein Hintergrundbild definiert ist, können nun benutzerdefinierte CSS-Klassen und CSS-Styles hinterlegt werden.

## Build 7.3.5 (02.03.2017)

### Popups für Bildobjekte beim Überstreichen mit der Maus in HTML

Bildobjekte können jetzt mit einem Popup-Text ausgestattet werden. Beim Export nach HTML wird ein Popup-Element angelegt, welches beim Überstreichen des Bildes mit der Maus per Javascript angezeigt und positioniert und beim Verlassen wieder ausgeschaltet wird.

## Build 7.3.4 (27.02.2017)

### Neue Makro-Events für Datenzugriffsobjekte

Für Datenzugriffsobjekte wurde ein neuer Makro-Event *OnRecordSelection* implementiert. Dieser wird immer aufgerufen, wenn eine Selektion eines Datenzugriffs stattgefunden hat.

## Build 7.3.3 (13.02.2017)

### Fehler beim Export von mehrseitigen Formularen nach HTML mit Bootstrap

Beim Export von mehrseitigen Formularen nach HTML mit Bootstrap-Unterstützung wurden beim wiederholten Export manchmal Objekte nicht korrekt exportiert. Dies führte zu Fehlern bei der Ausführung des Javascripts. Der Fehler wurde behoben.



## Build 7.3.2 (06.02.2017)

### Export von Zonen nach HTML mit Bootstrap

Beim Export von Formularen, die Zonen beinhalten, nach HTML mit Bootstrap-Unterstützung, werden die Zonen als eigenständige Einheiten exportiert, deren Sichtbarkeit durch Objekte ein- oder ausschaltbar ist. Dies war bisher nur beim Export ohne Bootstrap-Unterstützung möglich.

### Verschieben von Objekten, deren Position fixiert ist

Wenn von mehreren markierten Objekten bei einigen die Position fixiert ist und bei anderen nicht, wurde bisher das Bewegen der gesamten Gruppe verweigert. Diese Logik wurde dahingehend geändert, dass nun diejenigen Objekte, deren Position nicht fixiert ist, bewegt werden und die anderen ihre Position beibehalten.

### Verändern der Größe von Objekten, deren Größe fixiert ist

Wenn von mehreren markierten Objekten bei einigen die Größe fixiert ist und bei anderen nicht, wurde bisher das Verändern der Größe aller Objekte der gesamten Gruppe verweigert. Diese Logik wurde dahingehend geändert, dass nun diejenigen Objekte, deren Größe nicht fixiert ist, verändert werden und die anderen ihre Größe beibehalten.

## Build 7.3.1 (04.02.2017)

### Erweiterte HTML-Export-Einstellungen

Die Einstellungen des HTML-Exports wurde um den Tab-Reiter *Webseite* erweitert. Hier können die Minimalbreite und die Maximalbreite der Seite sowie die Vorgabe eines Hintergrundbildes konfiguriert werden.

### Hot-Key für die HTML-Export-Einstellungen

Die Einstellungen des HTML-Exports können nun mit Strg+Shift+H aufgerufen werden.

### Hot-Key für die PDF-Export-Einstellungen

Die Einstellungen des PDF-Exports können nun mit Strg+Shift+P aufgerufen werden.

## Build 7.3.0 (03.02.2017)

### Neues Objekt *HTML-Code*

Es wurde ein neuer Objekttyp *HTML-Code* implementiert. In diesem Objekt kann man nativen HTML-Code definieren, der beim Export von Formularen nach HTML als `<div>`-Tag exportiert wird.

Auf diese Weise ist es möglich, HTML-Code zu generieren, der sich nicht aus Formularobjekten ableiten lässt.

### CSS-Datei für HTML-Export

Beim Export nach HTML konnte in den Eigenschaften definiert werden, ob die von *eforms & more* bereitgestellte CSS-Datei oder eine vom Benutzer bereitgestellte CSS-Datei in das HTML-Formular eingebracht wird.

Ab sofort ist es alternativ auch möglich, beide Dateien oder keine der beiden Dateien auszuwählen.

### Export von Rechtecken nach Bootstrap

Rechtecke werden jetzt auch nach HTML mit Bootstrap exportiert. Sie nehmen die volle Breite des Containers ein und haben die im Formular definierte Höhe.

### Erweiterung des Objekts *Responsive Container*

Das Objekt *Responsive Container* für den Export nach HTML mit Bootstrap ist um einige Features erweitert worden. Insbesondere können für jeden Container folgende Komponenten vorgegeben werden:

- Rand um den Container im exportierten HTML
- Benutzerdefinierter CSS-String für den Container
- Liste benutzerdefinierter Klassen, die an den Container angehängt wird
- Liste benutzerdefinierter Klassen, die an jede Einheit angehängt wird

Ferner kann für jede einzelne Einheit des Containers folgende Komponenten vorgegeben werden:

- Benutzerdefinierter CSS-String, der an die betreffende Einheit angehängt wird
- Liste benutzerdefinierter Klassen, die an die betreffende Einheit angehängt wird

### Erweiterung von *Bild*-Objekten

Bild-Objekte verfügen ab sofort über die Möglichkeit, benutzerdefinierte CSS-Klassen und CSS-Styles zu hinterlegen, die beim Export nach HTML und Bootstrap in das HTML-Objekt eingebracht werden.

### Erweiterung von *Eingabefeld*-Objekten

Eingabefeld-Objekte verfügen ab sofort über die Möglichkeit, benutzerdefinierte CSS-Klassen und CSS-Styles zu hinterlegen, die beim Export nach HTML und Bootstrap in das HTML-Objekt eingebracht werden.

### Labels innerhalb von Eingabeobjekten beim HTML-Export mit Bootstrap

Eingabefelder, die nach HTML mit Bootstrap exportiert werden und über ein zugeordnetes Label verfügen, können jetzt so gestaltet werden, dass sich das Label oben rechts innerhalb des Eingabefeldes befindet. Die Eingabefelder werden zu diesem Zweck etwas höher gestaltet.

### *Glyphicons* für Labels von Eingabefeldern beim HTML-Export mit Bootstrap

Für Eingabefelder, die nach HTML mit Bootstrap exportiert werden, werden die von Bootstrap etablierten *Glyphicons* unterstützt. Diese können hinter jedes Label gesetzt werden. Beim Klicken auf das Icon wird ein Popup-Fenster eröffnet, in dem der Hilfetext des Eingabefeldes angezeigt wird.

## Build 7.2.9 (24.01.2017)

### Anordnen von Objekten

Die Funktion *Anordnen von Objekten* wurde funktional verändert.

Bei der Anordnung von Objekten in vertikaler Richtung werden die diese in der Reihenfolge angeordnet, die ihrer bisherigen vertikalen Anordnung entspricht. Bisher erfolgte die Orientierung am primär markierten Objekt und die Reihenfolge war undefiniert.

Dies gilt in entsprechender Form auch für die horizontale Anordnung.

Die Ausrichtung von Objekten erfolgt wie bisher und orientiert sich am primär markierten Objekt.

Die Funktion *Anordnen* im Menü *Objekt* kann jetzt auch durch die Tastenkombination Strg+R aufgerufen werden. Bisher war der Aufruf nur über das Menü möglich,

## Build 7.2.8 (20.01.2017)

### Anordnen von Objekten mit gleichen Abständen

Die Funktion *Anordnen von Objekten* wurde um die Möglichkeit erweitert, Objekte so anzuordnen, dass sie alle den gleichen vertikalen bzw. horizontalen Abstand voneinander haben.

### Explizite Auswahl des nächsten Objekts bei der Festlegung der TAB-Reihenfolge

Die Funktion *TAB-Reihenfolge manuell festlegen* wurde erweitert. Bisher konnte man die Reihenfolge nur ab dem ersten oder am einem bestimmten vorgegebenen Objekt festlegen. Jetzt kann während des Festlegens durch Klicken auf ein Objekt mit gedrückter Taste STRG auch zwischen durch ein neues Objekt bestimmt werden, ab dem die Reihenfolgedefinition fortgesetzt wird.

### Objekte mit fixierter Position können durch Cursortasten nicht verschoben werden

Markierte Objekte, deren Eigenschaft *Position fixieren* aktiv ist, konnten bisher mit der Maus nicht verschoben werden, wohl aber mit den Cursortasten der Tastatur. Dieses Verhalten wurde geändert. Ab sofort können fixierte Objekte auch mittels der Cursortasten nicht mehr verschoben werden.

### Objekte mit fixierter Größe können durch Cursortasten nicht verändert werden

Die Größe markierter Objekte, deren Eigenschaft *Größe fixieren* aktiv ist, konnten bisher mit der Maus nicht verändert werden, wohl aber mit den Cursortasten und gedrückter UMSCHALT-Taste. Dieses Verhalten wurde geändert. Ab sofort kann die Größe solcher Objekte auch mittels der Tastatur nicht mehr verändert werden.

## Build 7.2.7 (17.01.2017)

### Automatisches Vergrößern von Textobjekten beim Variantenangleich

Beim Angleichen von Objekten zweier Varianten wurde eine neue Option implementiert, die es ermöglicht, Textobjekte automatisch zu vergrößern, falls sich der Inhalt verlängert hat.

### Verlorene Texte beim Export von RTF-Texten

Beim Export von RTF-texten nach HTML oder PDF gingen mitunter Teile des Textes verloren. Der Fehler wurde beseitigt.

### Unsichtbare Rahmen beim Export von mehrzeiligen Eingabeobjekten

Beim Export mehrzeiliger Eingabeobjekte ging bei manchen Browsern der rechte oder untere Rand verloren. Der Fehler wurde beseitigt.

## Build 7.2.6 (28.12.2016)

### Zusammenfassen von Linien zu einem Rechteck

Die Funktion wurde dahingehend erweitert, dass nun auch Rechtecke einbezogen werden.

## Build 7.2.5 (24.12.2016)

### Angleichen von Varianten

Beim Angleichen zwischen Objekten zweier Varianten wurde die Funktionsweise geändert, so dass Objekte nur auf Basis ihres Namens angeglichen werden – unabhängig in welcher Zone oder auf welcher Seite sie sich befinden.

### Export und Import der Listen von Auswahlfeldern

Die Listen von Auswahlfeldern können jetzt in eine SCV-Datei exportiert werden und auch aus einer solchen wieder eingelesen werden.

### Eingabepflicht für Ankreuzfelder

Auch für Ankreuzfelder kann jetzt die Eigenschaft *Eingabepflicht* vorgegeben werden. Ein Ankreuzfeld mit Eingabepflicht muss vor dem Speichern oder Drucken des Formulars angekreuzt werden.

Insbesondere wird diese Eigenschaft auch nach HTML exportiert.

### Zusammenfassen von Linien zu einem Rechteck

Eine neue Funktion erlaubt es, mehrere (mindestens vier) Linien zu einem Rechteck zusammenzufassen. Die Linien werden dabei entfernt und das Rechteck wird als neues Objekt eingefügt.

### Automatische Erzeugung der TAB-Reihenfolge auf Basis der Position

Eine neue Funktion erlaubt es, die TAB-Reihenfolge aller Eingabeobjekte einer Seite auf Basis der Position der Objekte neu festzulegen.

### Blocksatz für mehrzeilige Texte beim HML-Export

Mehrzeilige Texte haben eine neue Option, mit denen die horizontale Ausrichtung auf *Block-satz* eingestellt werden kann. Der Blocksatz ist jedoch nur für den Export nach HTML relevant. Beim Ausfüllen des Formulars wird er wie Linksbündigkeit behandelt.

### Anzeige der HTML-Export-Zeilenhöhe im Editor

Wenn für mehrzeilige Eingabeobjekte oder für RTF-Objekte eine explizite Zeilenhöhe für den Export nach HTML festgelegt wird, wird diese jetzt auch im Editor angezeigt.

### Erneuerung des RTF-Parsers

Der RTF-Parser, der unter anderem verwendet wird, um RTF-Texte nach PDF oder HTML zu exportieren, wurde vollständig erneuert.

### Automatisches Löschen von Eingaben beim Zuklappen von HTML-Zonen

Wenn Formulare mit dynamischen Zonen nach HTML exportiert werden, können diese Zonen im exportierten HTML-Formular durch Checkboxen oder Comboboxen auf und zugeklappt werden.

Es wurde in den Einstellungen für den HTML-Export unter dem Reiter *JavaScript* eine neue Option implementiert, die es ermöglicht, beim Zuklappen einer Zone die Eingaben aller Felder zu löschen.

### Fehler beim Export von Barriere-Tags nach PDF

Seit der Verwendung der Bibliothek DynaPDF 3.0 sind Probleme beim Export von Barriere-Tags nach PDF aufgetreten. Diese wurden behoben.

## Build 7.2.4 (03.11.2016)

### Randeinstellungen von Eingabefeldern beim HTML-Export mit Bootstrap

Beim Export nach HTML mit der Option *Bootstrap* kann ab sofort die Gestaltung des Randes von Eingabefeldern explizit angegeben werden. Die Einstellung ist optional schaltbar und hat Vorrang vor der Bootstrap-Vorgabe. Auf diese Weise können auch Felder ohne Rand hergestellt werden.

### Wahl der Formulkategorie beim HTML-Export mit Bootstrap

Beim Export nach HTML mit der Option *Bootstrap* kann in den HTML-Einstellungen jetzt die Klasse des Formulars vorgegeben werden.

### Build 7.2.3 (15.09.2016)

#### Randfarbe von Container-Objekten wurde nicht gespeichert

Die Randfarbe von Container-Objekten wurde nicht gespeichert. Der Fehler wurde behoben.

#### Eingabefelder mittig beim PDF-Import

Beim Import eines Formulars aus einer PDF-Datei werden Eingabefelder jetzt mit der Voreinstellung *vertikal mittig* importiert.

### Build 7.2.2 (05.09.2016)

#### Eingabe neunstelliger Zahlen in HTML

In ganzzahligen Eingabefeldern können normalerweise Zahlen bis 999.999.999 eingegeben werden. In HTML wurde die Eingabelänge von ganzzahligen Eingabefeldern auf 9 Zeichen begrenzt. Dadurch war die Eingabe einer neunstelligen Zahl mit Tausendertrennern nicht möglich. Der Fehler wurde behoben.

#### Export von Ankreuzfeldern mit Eigenschaft *nicht änderbar* bzw. *nicht betretbar*

Ankreuzfelder mit der Eigenschaft *nicht änderbar* werden jetzt beim Export nach HTML auf Checkboxen abgebildet, die man zwar betreten, aber nicht ändern kann.

Ankreuzfelder mit der Eigenschaft *nicht betretbar* werden beim Export nach HTML auf Checkboxen mit dem Attribut *disabled* abgebildet.

### Build 7.2.1 (05.09.2016)

#### Farbe einstellbar für Container-Objekte

Die Farbe der Darstellung von Container-Objekten für Bootstrap kann jetzt durch den Anwender eingestellt werden. Bisher war diese fest vorgegeben.

#### Rechtsbündiges Ausrichten von Bildern beim HTML-Export mit Bootstrap

Bilder mit dem Attribut *rechtsbündig* werden jetzt auch in HTML mit Bootstrap rechtsbündig dargestellt.

#### Codepage Codierung vervollständigt

Die Codierung der Codepage beim Export nach HTML wurde bei den noch fehlenden Objekten ergänzt.

## Build 7.2.0 (26.08.2016)

### HTML-Export mit Bootstrap-Klassen

Der HTML-Export kann ab Version 7.2.0 wahlweise klassisch oder mit Bootstrap-Klassen zur Unterstützung eines *Responsive Design* ausgeführt werden.

In den Export-Einstellungen wurde ein neuer Reiter *Bootstrap* hinzugefügt. Hier können die Einstellungen für den Export mit Bootstrap-Unterstützung vorgenommen werden.

Folgende Objekte werden als Bootstrap-Objekte nach HTML exportiert:

- Texte
- RTF-Texte
- Bilder
- Buttons
- Eingabefelder
- Auswahlfelder
- Ankreuzfelder
- Linien

### Erweiterte Objekteigenschaften für den Bootstrap-Export

Wenn die Bootstrap-Unterstützung im HTML-Export eingeschaltet ist, werden in den Eigenschaften-Dialogen derjenigen Objekte, die nach HTML exportiert werden, spezifische Kontrollelemente für die Bootstrap-Konfiguration bereitgestellt.

### Neues Container-Objekt für den Bootstrap-Export

Wenn die Bootstrap-Unterstützung im HTML-Export eingeschaltet ist, steht im Editor ein neues Objekt *Responsive Container* zur Verfügung. Dieses erlaubt die Zuordnung von Objekten oder Objektgruppen zu den Spalten eines Grids, das sich in HTML automatisch umbreitet, wenn die Breite des Mediums kleiner wird.

### Erweiterte Konfiguration für den Bootstrap-Export

In der Konfiguration des Editors können die URLs konfiguriert werden, an denen die Skripte und Style-Klassen für Bootstrap hinterlegt sind. Diese werden in das exportierte HTML eingebettet.

### Textobjekte Eingabeobjekten als Labels zuordnen

Textobjekte können jetzt Eingabeobjekten als Label zugeordnet werden. Die Zuordnung kann entweder von Hand oder mittels einer automatischen Funktion erfolgen, mit der beliebig viele Paare von Label/Eingabeobjekt nacheinander verbunden werden können.



## Build 7.1.0 (08.08.2016)

### Duplizieren von Objekten in Zonen

Die Funktion *Duplizieren* im Menü *Bearbeiten* wurde dahingehend erweitert, dass nun auch Objekte dupliziert werden können, die sich in Seitenzonen befinden.

### Auswahl des Zeichensatzes beim HTML-Export

In den Optionen des HTML-Exports wurde eine Option implementiert, um den Zeichensatz auszuwählen, in dem die HTML-Seite codiert wird.

### Sortieren der Objekte in der Sammelbearbeitung

Bei der Sammelbearbeitung von Objekten werden diese jetzt sortiert angezeigt. Die Sortierung erfolgt nach Seitennummern, nach Objekttypen und wenn es sich um Eingabeobjekte handelt, nach der TAB-Reihenfolge. Bei Ausgabeobjekten wird nach der Position sortiert.

### Zonen automatisch anordnen

Es wurde eine neue Funktion implementiert, die es erlaubt, die Zonen einer Seite automatisch untereinander anzuordnen.

### Erweiterte Zoneneigenschaften

Der Dialog für die Bearbeitung der Eigenschaften einer Seitenzone verfügt ab sofort über ein Eingabefeld für die Eingabe der Zonenhöhe.

### Dynamische Zonen in HTML können Seitengröße anpassen

In den Export-Eigenschaften für HTML wurde unter dem Reiter *Javascript* eine neue Option implementiert. Wenn diese eingeschaltet ist, wird immer beim Auf- oder Zuklappen von Zonen in HTML automatisch die Länge der Seite der untersten sichtbaren Zone angepasst.

### Anzeige des Dateinamens im Fenstertitel

Die Form der Anzeige des Dateinamens im Fenstertitel kann jetzt in den *Einstellungen* im Menü *Extras* konfiguriert werden.

### Kein Sprung auf die erste Seite nach dem HTML-Export

Beim Export von mehrseitigen Formularen nach HTML wurde nach erfolgreichem Export immer zurück auf die erste Seite gesprungen. Jetzt bleibt die Position unverändert, es sei denn, das Formular verfügt über eine Variante mit dem Namen HTML, die beim Export automatisch aktiviert wird.

### Fehler beim Export von Datumsfeldern mit Picker im Firefox Browser

Beim Export von Datumsfeldern mit einem Picker nach HTML trat im Browser *Firefox* der Fehler auf, dass beim ersten Betreten des Feldes fehlerhafterweise der Schriftzug *false* ausgegeben wurde. Der Fehler wurde behoben.

## Build 7.0.7 (03.02.2016)

### Option zum Abschalten der automatischen Benennung von Buttons beim PDF-Export

In den Optionen des PDF-Exports wurde ein neuer Schalter implementiert, mit dem die automatische Benennung gleichnamiger Buttons beim Export nach PDF abgeschaltet werden kann.

## Build 7.0.6 (21.12.2015)

### Codepage 1252 für den HTML-Export

Beim HTML-Export wird ab sofort der Zeichensatz *WINDOWS-1252* verwendet und auch im Header der HTML-Datei eingetragen. Für den reinen HTML-Inhalt ist dies unbedeutend, da *eforms* alle kritischen Zeichen ohnehin als Entity exportiert. Wenn aber Sonderzeichen in Javaskripten verwendet werden, wie z.B. das Eurozeichen im Format für Zahlenformatierungen, wurden diese nicht korrekt wiedergegeben.

### Oben offene Eingabefelder und Auswahlfelder werden nach HTML exportiert

Die Eigenschaft *Rand oben offen* wird für Eingabefelder und Auswahlfelder jetzt auch nach HTML exportiert.

### Definierbarer Zeilenabstand für den Export von Text- und RTF-Objekten

Für mehrzeilige Textobjekte und für RTF-Objekte kann jetzt ein Zeilenabstand definiert werden, der beim Export nach HTML und PDF berücksichtigt wird. Die Angabe erfolgt in Punkten. Wenn kein Wert definiert wird oder der Wert 0 vorgegeben wird, erfolgt der Export wie bisher mit einem impliziten Abstand.

### Gleichnamige Buttons erhalten beim PDF-Export eindeutige Namen

Gleichnamig benannte Buttons erhalten beim Export nach PDF jetzt automatisch unterschiedliche Namen. Dies ist erforderlich, da es in einer PDF-Datei nicht erlaubt ist, wenn sie zwei gleichnamige Buttons enthält. Für unbenannte Buttons war dies schon immer der Fall.

### **Fehler beim Verarbeiten von RTF-Objekten, die eine Autorenangabe enthalten**

RTF-Objekte, deren RTF-Text einen Eintrag mit dem Namen des Autors enthielt, wurden nicht korrekt verarbeitet. Der Fehler wurde behoben.

## **Build 7.0.5** (15.11.2015)

### **Reihenfolge beim Export von Zonen nach HTML**

Die Reihenfolge der Zonen beim Export nach HTML wurde geändert. Zonen werden jetzt nicht mehr in ihrer nativen Reihenfolge exportiert. Vielmehr werden sie nach ihrer vertikalen Position sortiert und dann exportiert.

## **Build 7.0.4** (13.11.2015)

### **Eingabefelder in Zonen wurden im exportierten HTML nicht vor dem Absenden formatiert**

Für formatbehaftete Eingabefelder, wie z.B. Datumsfelder, kann beim Export nach HTML ein Format definiert werden, gemäß dem der Inhalt dieser Felder beim Absenden zum Server formatiert wird. Die Formatierung erfolgt nicht in Feldern, die sich innerhalb von Seitenzonen befanden. Der Fehler wurde behoben.

## **Build 7.0.3** (06.11.2015)

### **Seitenüberhang von Objekten jetzt auch für Zonen**

Die Einstellungen für den Seitenüberhang von Objekten die unter *Extras – Einstellungen* festgelegt werden können und mit denen man bestimmen kann, wie weit Objekte über den Rand einer Seite hinausragen dürfen, werden jetzt auch für Zonen verwendet.

### **Linien in Zonen werden nicht korrekt nach HTML exportiert**

Linien, die Zonen zugeordnet sind, wurden nicht korrekt nach HTML exportiert, da ihre Position relativ zur Seite anstatt zur Zone berechnet wurde. Dadurch wurden die Linien im Browser teilweise nicht mehr angezeigt. Der Fehler wurde behoben.

## Build 7.0.2 (28.10.2015)

### Automatisches Benennen unbenannter Objekte beim Duplizieren von Varianten

Beim Duplizieren von Varianten wurde eine neue Option implementiert, die es ermöglicht, automatisch alle unbenannten Objekte der zu duplizierenden Variante zu benennen, bevor das Duplikat erzeugt wird.

### Buttons in Zonen werden nicht korrekt nach HTML exportiert

Buttons, die Zonen zugeordnet sind, wurden nicht korrekt nach HTML exportiert, da ihre Position relativ zur Seite anstatt zur Zone berechnet wurde. Dadurch wurden die Buttons von den Browsern nicht mehr angezeigt. Der Fehler wurde behoben.

## Build 7.0.1 (03.10.2015)

### Formularname beim HTML-Export

Der Name des Formulars, der beim HTML-Export im `<form>` Tag als Attribut *name* erzeugt wird, kann jetzt durch den Anwender in den Einstellungen für den Formularexport unter *Absenden* konfiguriert werden. Bisher war als Name immer *WaimeaForm* verwendet worden.

## Build 7.0.0 (28.08.2015)

### Varianten

Im Formulareditor wurde eine Variantenverwaltung implementiert. Sie erlaubt es, ein Formular in verschiedenen Varianten zu pflegen. Jede Formularvariante beinhaltet unterschiedliche Ketten von Formularseiten.

Alle Varianten sind namentlich eindeutig benannt. Ein neu angelegtes Formular beinhaltet immer automatisch eine Variante mit dem Namen *Standard*.

Bestehende Formulare können ohne Einschränkung weiterverwendet werden. Für sie wird beim Aufruf automatisch eine Variante *Standard* angelegt, die das bestehende Formular repräsentiert. Auf diese Weise sind die neu implementierten Varianten vollständig aufwärtskompatibel.

Varianten haben nur Bedeutung im Editor. Im Ausfüller wird immer die Standard-Variante verwendet.

### Varianten-Manager

Für die Verwaltung der Varianten wurde ein Varianten-Manager implementiert. Dieser zeigt alle vorhandenen Varianten an. Er ermöglicht die Auswahl der augenblicklich aktiven Variante. Darüber kann der Varianten-Manager:

- Varianten anlegen
- Varianten duplizieren
- Varianten umbenennen
- Varianten löschen

Der Varianten-Manager kann während der Formularbearbeitung geöffnet bleiben. So kann jederzeit zwischen Varianten hin- und her geschaltet werden.

### Neue Buttonleiste für Varianten

Für die Varianten-Funktionen wurde eine eigene neue Symbolleiste implementiert. Diese kann mit der Funktion *Symbolleisten* im Menü *Anzeige* ein- und ausgeschaltet werden. Sie bietet neben der Schaltfläche zum Ein- und Ausschalten des Varianten-Managers auch eine Combobox zur direkten Auswahl der aktiven Variante an.

### Automatische Variantenumschaltung beim HTML- und PDF-Export

In den Einstellungen für den Export von Formularen kann eine Option eingeschaltet werden, die automatisch die Variante *HTML* bzw. *PDF* aktiviert, wenn ein Formular nach HTML bzw. PDF exportiert wird.

### Automatisches Angleichen von Objekten verschiedener Varianten

Es wurde eine neue Funktion implementiert, mit der alle oder bestimmte Eigenschaften von Objekten zwischen zwei Varianten angeglichen werden können. Der Abgleich basiert auf den Objektnamen. Diese sollten daher manuell eindeutig gemacht oder automatisch generiert werden.

### Automatisches Benennen von unbenannten Objekten

Es wurde eine neue Funktion implementiert, mit der alle markierten Objekte einer Seite, die noch keinen Namen haben, automatisch mit einem eindeutigen automatisch generierten Namen versehen werden. Diese Funktion ist nützlich, wenn man die Objekte verschiedener Varianten angleichen will, da die Funktion zum Angleichen auf der Namensgebung der Objekte basiert.

### Zonen ein- oder Ausblenden durch gegenseitig auslösende Ankreuzfelder

Zonen können jetzt auch in Abhängigkeit von sich gegenseitig auslösenden Ankreuzfeldern (Radio-Buttons) ein oder ausgeblendet werden. Diese Funktion wird auch im HTML-Export

abgebildet. Eines der sich gegenseitig auslösenden Ankreuzfelder muss den Namen tragen, der in den Zoneneigenschaften als Referenz angegeben wird.

#### **Ein- oder Ausblenden von Zonen mittels einer Javascript-Funktion**

Es wurde eine globale Javascript-Funktion *showZone* bereitgestellt, mit der eine einzelne Seite unter Bezug auf Ihren Namen ein- oder ausgeblendet werden kann. Die Funktion hat zwei Parameter. Der erste bezeichnet den Namen der Zone. Der zweite kann die Werte *true* oder *false* annehmen und entscheidet darüber, ob die Zone ein- oder ausgeblendet werden soll.

## Version 6

### Build 6.11.2 (17.08.2015)

#### Seiten ein- oder Ausblenden durch gegenseitig auslösende Ankreuzfelder

Seiten können jetzt auch in Abhängigkeit von sich gegenseitig auslösenden Ankreuzfeldern (Radio-Buttons) ein oder ausgeblendet werden. Diese Funktion wird auch im HTML-Export abgebildet. Eines der sich gegenseitig auslösenden Ankreuzfelder muss den Namen tragen, der in der Seitendarstellung als Referenz angegeben wird.

#### Ein- oder Ausblenden von Seiten mittels einer Javascript-Funktion

Es wurde eine globale Javascript-Funktion *showPage* bereitgestellt, mit der eine einzelne Seite unter Bezug auf Ihren Namen ein- oder ausgeblendet werden kann. Die Funktion hat zwei Parameter. Der erste bezeichnet den Namen der Seite. Der zweite kann die Werte *true* oder *false* annehmen und entscheidet darüber, ob die Seite ein- oder ausgeblendet werden soll.

### Build 6.11.1 (05.08.2015)

#### Seiten ein- oder ausblenden

Es wurde eine neue Funktion implementiert, mit der einzelne Seiten ein- oder ausgeblendet werden können. In den Eigenschaften jeder Seite kann eingestellt werden, ob die Seite immer sichtbar ist oder nur unter bestimmten Bedingungen.

Die Sichtbarkeit der Seite kann vom Zustand eines Ankreuzfeldes oder vom selektierten Eintrag eines Auswahlfeldes abhängig gemacht werden.

Die Funktionalität wird auch beim Export nach HTML unterstützt.

### Build 6.11.0 (20.07.2015)

#### Erweiterte Funktionen für Seitenzonen

Die Funktionen für die Bearbeitung von Seitenzonen wurden deutlich erweitert.

Für jede selektierte Zone wird jetzt ein Mittelpunkt angezeigt. An dieser Stelle kann die Zone mittels der Maus angefasst und auf der Seite nach oben oder unten geschoben werden.

Selektierte Zonen können jetzt mittels der Tasten *Umschalt+Oben* und *Umschalt+Unten* verkleinert bzw. vergrößert werden.

Beim Anlegen einer neuen Seitenzone wird diese selektiert und von allen anderen Zonen wird die Selektion entfernt.

### Fehlerbehebung beim Löschen von Seitenzonen

Wenn eine selektierte Seitenzone gelöscht wurde, waren alle Seitenzonen betroffen und wurden entfernt. Der Fehler wurde behoben.

### Export von Seitenzonen nach HTML

Der Export nach HTML wurde für Seitenzonen dahingehend erweitert, dass diese als eigenständiger `<div>`-Tag exportiert werden. Alle zur Zone gehörigen Objekte werden nicht mehr seitenrelativ positioniert, sondern in das `<div>` der Zone eingebettet und relativ zur Zone positioniert.

Zonen, die absolut zur Seite positioniert sind, erhalten den CSS-Style *position:absolute*.

Zonen, die sondern relativ zu vorhergehenden Zone positioniert sind, erhalten den CSS-Style *position:relative*.

### Erzeugung von Javascript Code für das Auf- und Zuklappen von Zonen in HTML

Beim Export von relativ positionierten Zonen nach HTML wird automatisch Javascript-Code erzeugt, der die Zone im Browser auf- bzw. zuklappt, wenn das der Zone zugeordnete Ankreuzfeld seinen Status ändert oder wenn das der Zone zugeordnete Auswahlfeld seinen Wert ändert.

## Build 6.10.9 (17.07.2015)

### Fehlerbehebung beim Anlegen von Objekten

Beim Anlegen neuer Objekte wurden diese außerhalb der Seite platziert. Der Fehler wurde behoben.

## Build 6.10.8 (12.07.2015)

### Fehlerbehebung bei der Darstellung von Ankreuzfeldern im PDF-Export

Beim Export von Ankreuzfeldern nach PDF wurde nicht immer alle Feldattribute korrekt exportiert. Der Fehler wurde behoben.



## Build 6.10.7 (03.07.2015)

### Fehlerbehebung bei der Vorbesetzung von Ankreuzfeldern im PDF-Export

Beim Export von Ankreuzfeldern nach PDF wurde die Vorbesetzung nicht korrekt exportiert. Der Fehler wurde behoben.

## Build 6.10.6 (20.05.2015)

### Erweiterter HTML-Export Auswahlfelder mit Eingabe

Der Export von Auswahlfeldern mit Eingabemöglichkeit nach HTML wurde erweitert. Es wird jetzt anstelle eines `<select>`-Tags ein `<input>`-Tag mit einer `<datalist>`-Anbindung erzeugt.

### Keine HTML Javaskripte für nicht exportierte Felder

Wenn beim Export nach HTML Eingabefelder das Attribut haben *nicht nach HTML exportieren*, werden für diese Felder auch keinerlei Javaskripte erzeugt – insbesondere auch keine welche prüfen, ob diese Felder gefüllt sind.

## Build 6.10.5 (12.04.2015)

### Übernahme der Eingabeprüfung beim Umwandeln von Eingabeobjekten

Beim Umwandeln von Eingabeobjekten in Auswahlfelder mit Eingabe wird ab sofort die Eingabeprüfung übernommen.

### Epilog und Prolog beim PDF-Export

Beim Export eines Formulars in eine PDF-Datei kann jetzt eine Prolog-Datei und eine Epilog-Datei konfiguriert werden. Beides müssen ebenfalls PDF-Dateien sein. Die Prolog-Datei wird der exportierten PDF-Datei vorangestellt, die Epilog-Datei wird hinten angehängt.

### Schriftart-Fehler beim Export nach PDF behoben

Beim Export eines Formulars in eine PDF-Datei werden Schriften mit der Höhe Null als automatische Schriftgröße exportiert. Die Schriftart wurde hierbei aber gelegentlich fehlerhaft übertragen. Der Fehler wurde behoben.

### Verbesserter PDF-Import

Beim Import von PDF-Dateien wurde das Zusammenziehen von Textkomponenten zu einem Text verbessert.

**Build 6.10.4** (21.01.2015)**Text in HTML-Eingabefeldern wird abgeschnitten**

In Eingabefeldern mit dem Format HTML wurden bei bestimmten Texten Sequenzen abgeschnitten. Der Fehler wurde behoben.

**Build 6.10.3** (17.12.2014)**Export von Leerzeilen beim Export nach PDF**

Beim Export von RTF-Texten nach PDF wurden leere Zeilen mit der Höhe Null exportiert. Der Fehler wurde behoben.

**Anführungszeichen beim Export nach PDF**

Beim Export von RTF-Texten nach PDF wurden schließende Anführungszeichen nicht exportiert. Der Fehler wurde behoben.

**Build 6.10.2** (01.12.2014)**Speicherfehler beim Export großer Formulare nach PDF**

Beim Export sehr großer Formulare nach PDF konnte es zu Speicherengpässen kommen, da temporär benutzte Ressourcen nicht wieder frei gegeben wurden. Der Fehler wurde behoben.

**Fehler beim Export bestimmter Zeichensequenzen nach PDF**

Beim Export verschiedener Zeichensequenzen nach PDF (z.B. obere Anführungszeichen) traten Fehler auf. Diese wurden behoben.

**Fehler bei der Erkennung von Textverbänden beim PDF-Import behoben**

Beim Import von Formularen aus PDF-Dateien wurden Verbände von Textpassagen mit bestimmten Abständen nicht als solche erkannt und verbunden. Der Fehler wurde behoben.

## Build 6.10.1 (18.11.2014)

### Neue Anzeigeoptionen für den Ausfüller

Im Editor wurden in den Anzeigeeinstellungen einige Optionen hinzugefügt, mit denen das Fenster des Ausfüllers manipuliert werden kann. Folgende Optionen wurden hinzugefügt:

- Ausblenden der Buttonleiste
- Ausblenden der Menüleiste
- Ausblenden der Statuszeile
- Setzen eines expliziten Titels für das Hauptfenster
- Setzen eines expliziten Titels für das Formularfenster

### Neue Makros zur Steuerung des Fensters des Ausfüllers

Es wurden einige neue Makrobefehle hinzugefügt, mit denen das Fenster des Ausfüllers beeinflusst werden kann:

- HideButtons Aus- oder Einblenden der Buttonleiste
- HideMenu Aus- oder Einblenden der Menüleiste
- HideStatus Aus- oder Einblenden der Statuszeile
- SetWindowSize Setzen der Fenstergröße des Fensters des Ausfüllers

### Export von leeren Einträge in Auswahlfeldern nach PDF

Leere Einträge in Auswahlfeldern wurden beim Export nach PDF bisher nicht exportiert. Diese werden jetzt in ein Leerzeichen umgewandelt und dadurch im PDF korrekt dargestellt.

## Build 6.10.0 (11.11.2014)

### Vereinfachte Bearbeitung mehrzeiliger Textobjekte

Im Eigenschaftendialog von Textobjekten können jetzt sowohl einzeilige als auch mehrzeilige Texte direkt bearbeitet werden, ohne einen Editor öffnen zu müssen.

### Bearbeitung globaler JavaScripts für den Export

Für die Bearbeitung der globalen JavaScripts für den HTML- und PDF-Export wurden eigenständige Funktionen implementiert. Bisher konnten die globalen Skripte nur in den Export-Einstellungen bearbeitet werden. Die Funktionen können durch die Shortcuts (Strg+Alt+P und Strg+Alt+H direkt aufgerufen werden.

### Externe Editoren für Makros und JavaScripts

Für die Bearbeitung von Makros und JavaScripts kann jetzt ein externer Editor konfiguriert werden. Dieser kann in den Einstellungen im Menü *Extras* konfiguriert werden. Wenn kein externer Editor konfiguriert wird, wird weiterhin der interne Editor verwendet.

### Export von Textobjekten nach HTML ohne Codierung

Beim Exportieren von Textobjekten werden alle Zeichen im Text, die eine Sonderbedeutung in HTML haben (z.B. <, > oder &) in sogenannte HTML-Entitäts umcodiert. Dies kann nun optional auch abgeschaltet werden. Dadurch können HTML-Sequenzen in den Textobjekten eingegeben werden. Der Anwender ist für die korrekte HTML-Syntax dann selbst verantwortlich.

### Objekte vom Export nach HTML oder PDF ausschließen

Es können Objekte vom Export nach HTML oder nach PDF ausgeschlossen werden. Dafür wurden Optionen in den Eigenschaften der Objekte vorgesehen.

### Keine Datumspicker in schreibgeschützten Eingabefeldern

In schreibgeschützten Eingabefeldern werden keine automatischen Datumspicker mehr ausgeblendet, da hier eine Datumseingabe ohnehin nicht möglich ist.

### Mehrere Textobjekte zu einem mehrzeiligen Textobjekt vereinen

Im Menü *Objekte* wird eine neue Funktion bereitgestellt, die alle markierten Textobjekte zu einem mehrzeiligen Textobjekt vereint. Die markierten Textobjekte werden aus dem Formular entfernt und es wird ein neues, mehrzeiliges Textobjekt eingefügt, welches sich an der Stelle befindet, an der sich das oberste der ursprünglichen Objekte befunden hat.

### Automatische erzeugte JavaScripts erkennen fehlerhafte Jahreszahlen

Automatisch erzeugte JavaScripts prüfen jetzt auch, dass die Jahreszahl in Datumsfeldern einen sinnvollen Wert darstellt.

## Build 6.9.5 (22.08.2014)

### CSS Styles zum Verstecken von Objekten beim HTML-Export

Beim Export nach HTML wurde der CSS-Style zum Verstecken von Objekten geändert. Es wird jetzt *visibility:hidden* anstelle von *display:none* verwendet.

## Build 6.9.4 (19.08.2014)

### Schmale Rechtecke beim PDF-Import in Linien umwandeln

Beim PDF-Import wurde eine Option hinzugefügt, die es ermöglicht, beim Importieren einer PDF-Datei schmale, randlose Rechtecke automatisch in Linien zu konvertieren.

## Build 6.9.3 (15.08.2014)

### Formularobjekte mit Standardwert vorbesetzen

Die Funktion *Eingaben mit Standardwert vorbesetzen* im Menü *Bearbeiten* des Ausfüllers hatte den Fehler, dass nur die aktuelle Seite vorbeasetzt wurde und nicht das gesamte Formular. Der Fehler wurde behoben.

Dadurch ist derselbe Fehler auch in der API-Funktion *Form\_SetDefaults* behoben.

### Weitere Verbesserungen beim PDF-Import

Der Import von PDF-Dateien als Formular wurde weiter verbessert.

## Build 6.9.2 (01.08.2014)

### Erweiterung der Feldteiler

Die maximale Anzahl der Feldteiler für Eingabefelder wurden von 30 auf 40 erhöht. Somit können jetzt auch Eingabefelder für SEPA-Überweisungen mit bis zu 40 Hauptteilern versehen werden.

### Neue Headerzeile für exportiere HTML-Dokumente

Die Kopfzeile für exportierte HTML-Dokumente wurde geändert. Sie lautet jetzt:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

## Build 6.9.1 (05.07.2014)

### Verbesserungen am PDF Import

Es wurden verschiedene Verbesserungen am PDF-Import implementiert.

## Build 6.9.0 (20.06.2014)

### Neue Funktion: PDF Import

Im Menü *Datei* wurde die Funktion *Neues Formular aus PDF-Datei importieren* hinzugefügt. Hiermit lassen sich Formulare oder interaktive Formulare, die als PDF-Datei vorliegen nach *eforms & more* importieren.

Beim Aufruf der Funktion öffnet sich zunächst ein Dialog, in dem verschiedene Parameter eingestellt werden können. Nachdem eine PDF-Datei für den Import ausgewählt wurde, kann diese importiert werden.

Der Import bildet die PDF-Datei technisch bedingt nicht hundertprozentig ab. Aber alle wesentlichen Design-Merkmale werden übernommen.

Die neu importierte Datei hat keinen Namen und muss beim Speichern benannt werden.

### Automatische Schriftverkleinerung beim Export nach PDF

Die Optionen des PDF-Exports wurden erweitert. Wenn die Option eingeschaltet ist, dass Eingabeobjekte nicht als interaktive Objekte exportiert werden, sondern als statische Texte, kann zusätzlich eine Option gewählt werden, die die Schriftart automatisch verkleinert, wenn der Feldinhalt nicht in das Eingabeobjekt passt.

### Option beim Export nach PDF: *Nur druckbare Objekte exportieren*

Die Optionen des PDF-Exports wurden erweitert. Es kann eine Option gewählt werden, so dass nur druckbare Objekte nach PDF exportiert werden.

### Fehler beim Laden von RTF-Texten

Beim Laden von manchen RTF-Texten gab es einen Programmabbruch, aufgrund eines Pufferüberlaufs in der Farbtabelle der RTF-Datei. Der Fehler wurde behoben.

### HTML Meta Tags

Die `<meta>` Tags beim Export nach HTML werden konform des Standards jetzt mit den Parametern *name* und *content* versehen.

### Attribut *readonly* beim HTML-Export

Das Attribut für HTML-Eingabefelder, die nicht verändert werden dürfen, wird ab sofort in der Syntax: `readonly="readonly"` codiert.

### Attribute *rows* und *cols* für mehrzeilige HTML-Eingabefelder

Mehrzeilige HTML-Eingabefelder werden mit dem HTML-Tag `<textarea>` codiert. Dieser Tag erhält ab sofort die Pflichtattribute `cols="1"` und `rows="1"`.

### Attribut *type* für Javascript-Verweise beim HTML-Export

Beim HTML-Export werden `<language>` Tags, mit denen auf Javascripte verwiesen wird, ab sofort mit dem Attribut `type="text/javascript"` ausgestattet.

## Build 6.8.11 (13.05.2014)

### Extra Tags für den HTML Header beim Export nach HTML

In den Parametern für den HTML-Export wurde ein weiterer Karteikartenreiter implementiert. Hier können beliebige HTML-Tags definiert werden, die beim HTML-Export in den HTML-Header übernommen werden. Diese Option ist sehr nützlich, wenn zusätzliche Skripte verwendet werden sollen.

### Vereinfachte Eingabe in Uhrzeit-Feldern

In Uhrzeit-Feldern kann die Uhrzeit jetzt statt in der Form 13:55 auch vereinfacht in der Form 1355 eingegeben werden.

### Transparente Bilder bleiben beim HTML-Export transparent

Bilder mit transparentem Hintergrund werden jetzt transparent nach HTML exportiert und nicht länger mit einem weißen Hintergrund.

## Build 6.8.10 (16.04.2014)

### Eingabe des Zeichens & bei HTML-Links

Bei der Eingabe von Links für den HTML-Export konnte das Zeichen & nicht eingegeben werden. Der Fehler wurde beseitigt.

### Leerzeilen in Auswahlfeldern

Bei Auswahlfeldern war es bisher nicht möglich, leere Einträge in der Auswahlliste abzulegen. Dies ist nun möglich.

### Statustext als *Mouse Over* Text beim HTML-Export von Buttons

Beim Export von Buttons nach HTML wird der Statustext als *Mouse Over* Text exportiert.

**Build 6.8.9** (10.04.2014)**Notification Message beim Ändern eines Eingabefeldes per Makro**

Wenn der Inhalt von Eingabefeldern per Makro geändert wurde, wurde keine Notification Message an die aufrufende Anwendung gesendet. Der Fehler wurde behoben.

**Notification Message beim Ändern eines Eingabefeldes per Makro**

Wenn der Inhalt von Eingabefeldern per Makro geändert wurde, wurde keine Notification

**Build 6.8.8** (26.03.2014)**Kontextmenü für Eingabefelder**

Eingabefelder mit einfachem Text zeigten beim Rechtsklick im Ausfüllmodus kein Kontext-Menü. Der Fehler wurde behoben.

**Speicherverluste beseitigt**

Beim Umschalten vom Bearbeitungsmodus in den Ausfüllmodus traten Speicherverluste auf. Der Fehler wurde behoben.

**Build 6.8.7** (13.03.2014)**Speicherverluste beseitigt**

Bei längerem Betrieb der Formularbibliothek traten Speicherverluste auf. Der Fehler wurde behoben.

**Build 6.8.6** (18.02.2014)**Fehler in der Speicherverwaltung behoben**

Es wurde ein Fehler in der Speicherverwaltung behoben, der in seltenen Fällen beim Schließen eines Formulars Nebeneffekte hervorrief. So konnte es z.B. vorkommen, dass das Hauptfenster der Anwendung kommentarlos geschlossen wurde.



## Build 6.8.5 (06.02.2014)

### Export geteilter Eingabefelder nach PDF

Der Export von geteilten Eingabefeldern nach PDF wurde implementiert.

### Fehler in der englischen Übersetzung beseitigt

In der englischen Übersetzung wurden drei Fehler beseitigt.

### Funktion *alles markieren* in mehrzeiligen Eingabefeldern

Die Funktion *Alles Markieren* mit dem Shortcut *Strg+A* funktionierte bisher nur in einzeiligen Eingabefeldern. Sie ist ab sofort auch in mehrzeiligen Eingabefeldern verfügbar.

### Export von formatbehafteten Eingabefeldern nach HTML

Beim Export von ausgefüllten Eingabefeldern nach HTML wurden die Inhalte von Eingabefeldern nicht korrekt exportiert, wenn diese die Option RTF- oder HTML-Format hatten. Der Fehler wurde behoben.

## Build 6.8.4 (27.01.2014)

### Neue Makrobefehle

Es wurden neue Makrobefehle implementiert:

<b>Alert</b>	Textnachricht in einem Systemmodalen Fenster anzeigen
<b>GetFontProperty</b>	Auslesen der Eigenschaften des Fonts eines Objekts
<b>SetFontProperty</b>	Setzen der Eigenschaften des Fonts eines Objekts

Der Befehl *Alert* ist in der Dokumentation *Allgemeine Makrobefehle* dokumentiert. Die Befehle *GetFontProperty* und *SetFontProperty* sind in der Dokumentation *Formular Makrobefehle* beschrieben.

### Neue Makroereignisse für das gesamte Formular

Es wurden neue Makroereignisse implementiert:

<b>Beim Doppelklicken auf ein Objekt</b>	Event: <i>OnAnyObjectDbkClk</i>
<b>Beim Rechtsklicken auf ein Objekt</b>	Event: <i>OnAnyObjectRightClick</i>

### Neues Makroereignis für Buttons, Auswahlfelder und Ankreuzfelder

Für Buttons, Auswahlfelder und Ankreuzfelder wurde ein neues Makroereignis implementiert:

**Nach dem Rechtsklicken auf das Objekt** Event: *OnObjectRightClick*

### Build 6.8.3 (20.01.2014)

#### Gemeinsame Objekteigenschaften erweitert

Der Dialog zur Bearbeitung gemeinsamer Objekteigenschaften wurde um die Eigenschaft *Eingabepflicht* erweitert. Es werden nur solche Objekte eingeschlossen, die über diese Eigenschaft verfügen. Alle anderen Objekte bleiben unverändert.

### Build 6.8.2 (17.01.2014)

#### Einbetten von Java Scripts in die HTML-Datei

Es wurde eine neue Option implementiert mit der bestimmt werden kann, dass Java Scripts in die HTML eingebettet werden anstatt in einer separaten Datei abgelegt zu werden.

### Build 6.8.1 (13.01.2014)

#### Automatische Drehung von Bildern gemäß EXIF-Kenndaten

Bilder, in deren EXIF-Kenndaten eine Bilddrehung hinterlegt ist (90 Grad gegen der Uhrzeiger, 90 Grad im Uhrzeiger oder 180 Grad), werden ab sofort auch mit dieser Drehung dargestellt.

### Neue API-Schnittstelle zum Einfügen eines Textes in das aktive Eingabefeld

Es wurde eine neue API-Schnittstelle implementiert, die in das Windows-Control des augenblicklich aktiven Eingabefeldes eine Textsequenz einfügt. Der Name der Schnittstelle ist:

*FormObj\_InsertToControl*

Die Schnittstelle ist in der API-Dokumentation *formdev-049.pdf* beschrieben.

## Build 6.8.0 (21.12.2013)

### Export eines ausgefüllten Formulars nach HTML

Neben dem Export der Formularvorlage mit allen interaktiven Elementen kann jetzt auch ein fertig ausgefülltes Formular nach HTML exportiert werden. Das HTML-Formular ist nicht interaktiv und kann nicht verändert werden. Die Funktion ist im Editor verfügbar und kann nur im Ausfüllmodus verwendet werden.

Eingabefelder werden mit ihrem augenblicklichen Inhalt exportiert. Sie können betreten werden, sind aber nicht veränderbar. Auswahlfelder und Ankreuzfelder werden mit der aktuellen Auswahl exportiert. Sie sind *disabled* und können nicht verändert werden. Buttons werden nicht exportiert, da das Zielformular nicht interaktiv sein soll.

Die Leistung steht auch als API-Funktion zur Verfügung. Setzen Sie in der API-Funktion *Form\_Export* das Flag *FORM\_EXPORT\_FILLED*.

## Build 6.7.15 (19.12.2013)

### Option zur Unterdrückung von Druckerfehlermeldungen

Es wurde eine neue Option implementiert, mit der Fehlermeldungen beim Öffnen eines Druckers unterdrückt werden können. Die Option wird als Schalter in der API-Funktion *Form\_Open\_Expanded* angeboten.

## Build 6.7.14 (13.12.2013)

### Schriftgröße 0

Wenn als Schriftgröße 0 eingestellt wird, erfolgt die Darstellung im Formular und im Export nach HTML als 12-Punkt-Schrift. Beim Export nach PDF wird die Schriftgröße auf *AUTO* abgebildet.

### Reihenfolge von Ausgabeobjekten beim Export nach HTML

Der Export von Ausgabeobjekten nach HTML erfolgt jetzt in der gleichen Reihenfolge, wie diese auch angezeigt werden. Dadurch bleiben Objekte, die im Vordergrund liegen, auch nach dem Export im Vordergrund.

## Build 6.7.13 (10.12.2013)

### HTML Eingabeobjekte

Wenn in mehrzeiligen Eingabeobjekten der Textart *HTML* eines der Zeichen < oder > eingegeben wurde, traten Fehler bei der Konvertierung auf. Der Fehler wurde behoben.

## Build 6.7.12 (02.12.2013)

### Markieren mit der Maus erweitert

Das Markieren von Objekten mit der Maus wurde um eine neue Funktion erweitert. Wenn beim Drücken der linken Maustaste die *Umschalt*-Taste gedrückt ist, wird immer die Funktion *Markieren durch Umschließen mit einem Rahmen* ausgeführt. Dies ist sonst nur der Fall, wenn außerhalb eines Objekts geklickt wird.

### Neue Option beim HTML-Export

Es wurde eine neue Option für den HTML-Export implementiert, die es erlaubt, Objekte, die vollständig außerhalb jeder Seite liegen, nicht zu exportieren.

### Attribut *name* für Ausgabeobjekte beim HTML-Export

Beim Export nach HTML wird für textuelle Ausgabeobjekte (Texte, Datum, Uhrzeit, etc.) deren Name nicht leer ist, das Attribut *name="Name"* hinzugefügt.

## Build 6.7.11 (22.11.2013)

### CSS für Textobjekte

Für Textobjekte kann jetzt ein individueller CSS Style vorgegeben werden, der beim Export nach HTML an den automatisch generierten Style angehängt wird.

### Passwort-Eingabeobjekte

Für Eingabeobjekte kann die neue Option *Passworteingabe* gesetzt werden. Das Eingabefeld wird dann so ausgelegt, dass sein Inhalt nicht angezeigt wird. Die Option wird auch beim Export nach HTML berücksichtigt.

### Absturz beim Doppelklick neben eine Seite

Beim Doppelklick in den Leerraum neben einer Seite im Editor konnte ein Programmabsturz auftreten. Der Fehler wurde behoben.

### Export von Auswahlfeldern mit Leerzeichen im Wert nach HTML

Auswahlfelder mit Einträgen, in denen als Wert nur ein oder mehrere Leerzeichen eingetragen sind, wurden nicht korrekt nach HTML exportiert. Als Wert wurde der Anzeigename exportiert anstelle der Leerzeichen. Der Fehler wurde behoben.

## Build 6.7.10 (06.10.2013)

### Fehler beim Exportieren von Ankreuzfeldern nach PDF

Ankreuzfelder erhielten beim Export nach PDF falsche Namen. Der Fehler wurde behoben.

## Build 6.7.9 (02.10.2013)

### HTML Links in neuem Tab öffnen

Textobjekte und Bildobjekte können beim HTML-Export Links auf andere HTML-Seiten repräsentieren. Die Links können ab sofort wahlweise im selben Tab-Element des Browsers geöffnet werden, wie das Formular oder in einem neuen Tab-Element.

### Horizontale Ausrichtung von Bildern beim HTML-Export

Die horizontale Ausrichtung von Bildobjekten wurde bisher beim HTML-Export nicht berücksichtigt. Dies wurde jetzt nachgerüstet.

### Führende Leerzeichen in Textobjekten

Führende Leerzeichen in Textobjekten gingen beim Speichern verloren. Der Fehler wurde behoben.

### Hochkommata in Listen von Auswahlfeldern

Hochkommata in den Einträgen von Auswahlfeldern führten zu einem zerstörten Formular, da die XML-Syntax der Formulardatei inkorrekt wurde. Der Fehler wurde behoben.

### Wasserzeichen nach PDF exportieren

Beim Exportieren von Formularen mit Wasserzeichen nach PDF trat ein Programmfehler auf, wenn das Wasserzeichen Ankreuzfelder beinhaltete. Der Fehler wurde behoben.

## Build 6.7.8 (24.09.2013)

### Datumpicker in HTML

Der Datumpicker im nach HTML exportierten Formular wurde so modifiziert, dass er sich nur beim Klicken in ein Datums-Eingabefeld öffnet.

### Placeholder in mehrzeiligen Eingabefeldern

Beim Export nach HTML von mehrzeiligen Eingabefeldern wurde das Attribute *Placeholder* auch dann exportiert, wenn der Export ausgeschaltet war. Der Fehler wurde behoben.

### Tausendertrenner in numerischen HTML-Eingabefeldern

Wenn in numerischen Eingabefeldern Tausendertrenner zur Formatierung verwendet wurden und dann der Wert mittels der Javascript-Funktion *valueAsFloat* oder *valueAsInt* ausgelesen wurde, wurde der Tausendertrenner als Dezimalkomma interpretiert. Die Routinen wurden so modifiziert, dass die Tausendertrenner nun ausgefiltert werden.

## Build 6.7.7 (22.09.2013)

### Position von RTF-Objekten beim Export nach HTML

Die Position von RTF-Objekten beim Export nach HTML war manchmal fehlerhaft. Der Fehler wurde behoben.

### Export von RTF-Texten mit Aufzählungen nach HTML

Beim Export von RTF-Texten mit Aufzählungen traten Font-Fehler auf. Der Fehler wurde behoben.

### Datumpicker in HTML

Formulare, für die ein Datumpicker definiert ist, werden auch im exportierten HTML mit einem Javascript Datumpicker unterstützt.

## Build 6.7.6 (18.09.2013)

### Verbesserter Export von Ankreuzfeldern nach HTML

Die Position des Labels beim Export von Ankreuzfeldern nach HTML wurde verbessert.

### Export von €-Zeichen nach HTML

€-Zeichen wurden bisher beim Export nach HTML unterdrückt. Der Fehler wurde behoben.

### Einträge von Auswahlfeldern

Leere Felder als Anzeigetext von Einträgen in Auswahlfeldern führten zu Vertauschungen von Werten der Liste. Der Fehler wurde behoben.

## Build 6.7.5 (16.09.2013)

### Export nach HTML und PDF gleichzeitig

Es wurde eine neue Funktion im Editor implementiert, mit der ein Formular gleichzeitig nach HTML und nach PDF exportiert wird. Alle entstehenden Zieldateien werden in einer ZIP-Datei abgelegt.

### Erzeugung der *placeholder*-Attribute ein und ausschaltbar

Es wurde eine neue HTML-Export-Option implementiert, mit der ausgewählt werden kann, ob *placeholder*-Attribute erzeugt werden oder nicht.

### Leerzeilen und Texthervorhebungen verschwinden in mehrzeiligen HTML-Eingabefeldern

In mehrzeiligen Eingabefeldern mit HTML-Inhalt verschwanden gelegentlich Leerzeilen und Texthervorhebungen. Der Fehler wurde behoben.

### Textgröße in HTML- oder RTF-Eingabefeldern

In HTML- oder RTF-Eingabefeldern wurde der Text zu groß angezeigt, wenn das Eingabefeld den Fokus hatte. Die Größe wurde angeglichen.

## Build 6.7.4 (03.09.2013)

### Option zum Drucken der Objektnamen im Editor

Es wurde in den Druckeinstellungen eine Option implementiert, mit der man wählen kann, ob die Namen der Eingabeobjekte beim Drucken im Editor ausgegeben werden oder nicht. Diese Option ist im Ausfüller ausgeblendet, da sie hier keine Wirkung hat.

### Option zum Bearbeiten von Custom-Strings

Alle Eingabeobjekte von *efoms & more* verfügen über einen sogenannten *Custom-String*. Hier können OEM-Anwendungen eine anwendungsspezifische Zeichenfolge speichern. Bisher konnte der *Custom-String* nur mittels einer API-Schnittstelle gesetzt oder gelesen werden. Nun wurde in den Einstellungen des Editors im Menü *Extras* eine neue Option implementiert, mittels der das Bearbeiten von Custom-Strings auch im Editor ermöglicht wird. Wenn diese Option eingeschaltet wird, erscheint in den Eigenschaften-Dialogen aller Eingabeobjekte ein Button *Custom String...*, der die Bearbeitung ermöglicht.

### Entfernen von Hintergrundfarben in mehrzeiligen Eingabefeldern

In mehrzeiligen Eingabefeldern, deren Inhalt im Format *RTF* oder *HTML* gespeichert wird, ist es möglich, für den Text oder einzelne Passagen eine explizite Hintergrundfarbe zu wählen. Bisher konnte diese nicht zurückgesetzt werden. Ab dieser Version setzt die Funktion *Normal* im Kontextmenü der betreffenden Eingabefelder nicht nur die Schriftgestaltung sondern auch die Hintergrundfarbe auf die Normaleinstellung zurück.

### Exportieren von Button-Skripten nach HTML

HTML-Skripte, die in Buttons hinterlegt sind, wurden bisher nur dann nach HTML exportiert, wenn für den Button die Aktion *Java Skript ausführen* eingestellt war. Ab sofort erfolgt der Export auch dann, wenn die Voreinstellung *Makro ausführen* eingestellt ist.

### Export der Voreinstellung von Auswahlfeldern nach HTML

Bei Auswahlfeldern wird jetzt auch die Voreinstellung nach HTML exportiert. Das bedeutet, dass im Browser automatisch dasjenige Element vorselektiert ist, welches als Voreinstellung eingetragen ist.

## Build 6.7.3 (13.08.2013)

### Neue Option für den Export von Absätzen nach HTML

Es wurde im HTML-Export eine neue Option implementiert, mit der Absätze in RTF-Texten als Zeilenschaltungen exportiert werden können.



**Build 6.7.2** (06.08.2013)**Javascript-Fehler beim HTML-Export behoben**

Formatbehaftete Felder wurden nicht korrekt übertragen. Der Fehler wurde behoben.

**Neue Javascript-Funktionen für den HTML-Export**

Die Funktion *util.scand* wurde für das Format TT.MM.JJJJ implementiert.

**Build 6.7.1** (22.07.2013)**Fehler beim Festlegen der Reihenfolge**

Beim Festlegen der Reihenfolge von Eingabeobjekten waren gelegentlich Speicherletzungen aufgetreten. Der Fehler wurde behoben.

**Build 6.7.0** (20.07.2013)**Neues Eingabeobjekt: *Listbox***

Es wurde der neue Objekttyp *Listbox* implementiert. Listboxen sind Eingabeobjekte. Sie verhalten sich ähnlich wie Auswahlfelder. Im Gegensatz zum Auswahlfeld ist die Auswahlliste immer aufgeklappt.

**Build 6.6.4** (02.07.2013)**Unterstützung des HTML-Attributs *placeholder***

Das Attribut *placeholder*, das in HTML 5 zur Verfügung steht, wird ab sofort unterstützt. Der Statustext wird als Attribut *placeholder* exportiert.

**Unterstützung des *value*-Members für Radiobuttons in JavaScript**

Das Klassenmember *value* wird ab sofort für sich gegenseitig auslösende Ankreuzfelder (Radio Buttons) unterstützt. Es kann mit den Werten *On* oder *Off* besetzt werden. Die Groß- und Kleinschreibung ist relevant.

### **Build 6.6.3** (19.06..2013)

#### **Neue Funktion *Skripte löschen***

Es wurde eine neue Funktion implementiert, die es erlaubt, Skripte aller Objekte zu löschen.

#### **Attribut *hidden* wird jetzt auch in HTML-Javascript unterstützt**

Das Attribut *hidden* wird jetzt in HTML-Javascripten unterstützt.

#### **Ankreuzfelder erzeugen jetzt auch Javascripte**

HTML-Javascripte in Ankreuzfeldern wurden bisher nicht korrekt nach HTML exportiert. Der Fehler wurde behoben.

### **Build 6.6.2** (12.06..2013)

#### **Fehler im PDF-Javascript-Konverter behoben**

Es wurden verschiedene Fehler bei der Konvertierung von PDF-Javascript nach HTML-Javascript behoben.

### **Build 6.6.1** (06.06..2013)

#### **Erweiterung der Einstellungen**

Es kann im Menü *Extras* in der Funktion *Einstellungen* vorgegeben werden, wie weit im Editor Objekte außerhalb der Seitenränder verschoben werden können.

### **Build 6.6.0** (27.05.2013)

#### **Anzeige des Makronamens Makro-Fehlermeldungen**

Wenn in einem Makro ein Fehler auftritt, wird in der Fehlermeldung der Name des Objekts und der Name des Makros ausgegeben, um bei komplexen Formularen das fehlerhafte Makro besser lokalisieren zu können.

#### **Vereinheitlichung der Javaskripte für HTML und PDF**

Um Kunden, die Formulare nach HTML und nach PDF exportieren wollen, die Programmierung von Javascript zu erleichtern, wurde ein Framework implementiert, das die Syntax der in PDF verwendeten Skripte auch im HTML-Umfeld erkennt.

Das gesamte Framework ist separat dokumentiert.

## **Build 6.5.8** (24.04.2013)

### **Neue Funktion zum Kopieren von Objekt-Skripten**

Es wurde eine Funktion implementiert, die es erlaubt, alle Skripte aller Objekte in eine andere Objektklasse zu kopieren. Insbesondere können die Javaskripte für den PDF-Export in die Javaskripte für den HTML-Export übernommen werden. Die Funktion befindet sich im Menü *Objekte* im Untermenü *Skripte und Makros*.

## **Build 6.5.7** (16.04.2013)

### **Neue Funktion zum Bearbeiten aller Makros oder Skripte**

Um große Formulare mit vielen Makros oder Skripten besser verwalten zu können, wurde eine neue Funktion implementiert, die alle Skripte bzw. Makros aller Eingabeobjekte aufammelt und als Block in einem Editor zur Bearbeitung anbietet. Im Menü *Objekte* gibt es ein neues Untermenü *Skripte und Makros*. Hier ist die Funktion zu finden.

## **Build 6.5.6** (03.04.2013)

### **Erzeugung von *Title-Attributen* beim HTML-Export**

Beim Export nach HTML wird der Anzeigetext für die Statuszeile für alle Eingabeobjekte als Attribut *Title=""* exportiert. Die meisten Browser zeigen diesen Text dann als Tooltip an.

### **Erzeugung von *ID-Attributen* beim HTML-Export**

Beim Export nach HTML wird für alle Eingabeobjekte ein Attribut *id=""* mit einem eindeutigen Wert generiert. Der Wert der ID hat folgende Form:

X-00001

Das erste Zeichen ist:

- P, wenn das Feld Eingabepflicht hat
- F, wenn das Feld keine Eingabepflicht hat

Die Nummer ist ein eindeutiger fünfstelliger Wert mit führenden Nullen.

### Fehler beim Namen von Ankreuzfeldern beim HTML-Export.

Beim Export nach HTML wurde für Ankreuzfelder ein fehlerhafter Name erzeugt. Der Fehler wurde behoben.

## Build 6.5.5 (08.03.2013)

### Wahl des Browsers beim HTML-Export

Im Menü *Extras* wurde eine neue Funktion *Browser-Optionen* implementiert. Hier kann eingestellt werden, ob nach dem HTML Export ein Browser geöffnet wird, um den Export anzuzeigen. Ferner kann ausgewählt werden, ob der Standardbrowser verwendet wird, oder ob der Browsertyp beim Anwender angefragt wird.

### Zahlreiche Verbesserungen beim HTML-Export

Es wurden zahlreiche Verbesserungen beim HTML-Export implementiert:

- Umrandete Schaltflächen werden jetzt mit Umrandung exportiert
- Linien erscheinen jetzt auch im Browser *Google Chrome*
- Unsichtbare Eingabefelder werden jetzt exportiert (mit Attribut *hidden*)
- Schreibgeschützte Auswahlfelder werden schreibgeschützt exportiert
- Leere Einträge in Auswahlfeldern werden jetzt korrekt exportiert
- Eine Folge von Leerzeichen in RTF-Texten wird nun korrekt exportiert
- Die Positionierung von Checkboxes und Radiobuttons wurde verbessert

### Makrobefehle für die aktuelle Seite

Bei Makrobefehlen bedeutet ein leerer Seitenname, dass sich der Befehl auf die aktuelle Seite beziehen soll. Dies war nicht immer der Fall. Der Fehler wurde behoben.

## Build 6.5.4 (16.02.2013)

### HTML-Links für Bilder

Ab sofort können Bilder mit einem HTML-Link versehen werden. Dies war bisher nur für Textobjekte möglich. Wenn ein HTML-Link konfiguriert ist, wird beim HTML-Export automatisch ein Link-Tag für das Bild generiert.

### Aktivieren der Software ohne Internet-Verbindung

Kunden ohne Internet-Verbindung konnten die Software bisher nicht aktivieren, da hierzu eine Internet-Verbindung erforderlich war. Es wurde eine Alternative zur Online-Aktivierung

implementiert, bei der die Aktivierung mittels eines Aktivierungscodes erfolgt, der per Mail oder Telefon beim Hersteller angefragt werden kann.

### **Build 6.5.3** (04.02.2013)

#### **Löschen der Eingabe nach Ausgabe einer Fehlermeldung**

Beim Exportieren nach HTML kann eine neue Option gesetzt werden, die die erzeugten Java-Skripte so modifiziert, dass fehlerhafte Eingaben nach Anzeige der Fehlermeldung gelöscht werden. Diese Option ist standardmäßig eingeschaltet.

#### **Export von RTF-Texten nach HTML**

Beim Exportieren von RTF-Texten nach HTML wurde manchmal der gesamte Text nicht ausgegeben. Der Fehler wurde behoben.

#### **Export von Buttons nach HTML**

Der Export von Buttons nach HTML wurde vollständig überarbeitet. Es wurde eine neue Option implementiert, die es erlaubt, Buttons grundsätzlich in der nativen Browserdarstellung und nicht in der im Formular vorgegebenen Gestaltung zu exportieren. Diese Einstellung wirkt auf alle Buttons des Formulars.

### **Build 6.5.2** (01.02.2013)

#### **Styles (CSS) einbetten beim HTML-Export**

Beim Exportieren nach HTML können CSS-Styles jetzt wahlweise in die Ziel-HTML-Datei eingebettet oder in eine separate Style-Datei exportiert werden.

#### **Buttons für HTML-Export und PDF-Export**

Für die Funktionen *HTML-Export* und *PDF-Export* wurde in der Buttonleiste des Editors je eine Schaltfläche implementiert.

### **Build 6.5.1** (31.01.2013)

#### **Positionskorrektur für Ankreuzfelder beim HTML-Export**

Die Position von Ankreuzfeldern kann beim Export nach HTML nach links und nach oben korrigiert werden. Eine entsprechende Einstellung kann in den Exportheigenschaften eingestellt werden. Die Korrektur gilt für alle Ankreuzfelder des Formulars.

## Build 6.5.0 (29.01.2013)

### HTML Export vollständig überarbeitet

Der Export von Formularen nach HTML wurde vollständig überarbeitet. Das Layout und alle Funktionen wurden erheblich verbessert.

### Export von RTF-Objekten nach HTML

RTF-Objekte werden nach HTML konvertiert. Dabei werden folgende Schrifteigenschaften unterstützt:

- Schriftart
- Schriftgröße
- Schriftdekoration (Unterstreichung)
- Schriftfarbe
- Schriftgestaltung (kursiv, fett)

### Unterstützung von CSS-Style-Sheets beim HTML-Export

Beim Export nach HTML werden können CSS-Dateien als Vorlage definiert werden. Ferner werden beim Export CSS-Dateien erzeugt, die mit dem generierten HTML-Code verlinkt werden. CSS-Dateien können entweder automatisch generiert werden, als auch von Hand beeinflusst.

### Trennung der Exportfunktionen nach HTML und PDF

Der Export nach HTML und PDF war bisher unter einem Menüeintrag *Export* zusammengefasst. Die Funktionen wurden in zwei separate Menüeinträge aufgeteilt: *Export nach HTML* und *Export nach PDF*.

### Separate Lizenz für HTML-Export

Der HTML Export ist ab Version 6.5.0 nicht mehr in der Standardlizenz des Editors enthalten. Er muss separat lizenziert werden. Wie bei allen Lizenzprodukten von Waimea kann eine zeitlich begrenzte Demo-Version zum Ausprobieren eingerichtet werden.

## Build 6.4.5 (17.01.2013)

### Makrobefehle für die Erzeugung von XML-Dateien

Es wurden neue Makrobefehle implementiert, die es ermöglichen, eine XML-Datei zu erzeugen:

*XMLCreate*  
*XMLDelete*  
*XMLGetRoot*  
*XMLAddChild*  
*XMLSetAttr*  
*XMLWriteFile*

Die Makrobefehle sind in der Dokumentation *Allgemeine Makrobefehle* beschrieben.

### **Makrobefehl für die Erzeugung eines XML-Knotens für ein FritzBox Telefonbuch**

Es wurde ein neuer Makrobefehl:

`XMLAddFritzContactNode`

implementiert, der aus den Eingabefeldern eines Formulars in Abhängigkeit von den ihnen zugewiesenen Bedeutungen einen XML-Knoten für ein Telefonbuch für die Fritz!Box der Firma AVM erzeugt. Dadurch können Fritz!Box-Telefonbücher makrogesteuert erzeugt werden.

## **Build 6.4.4** (13.01.2013)

### **Wasserzeichen für einzelne Seiten**

Ähnlich dem Wasserzeichen für das gesamte Formular, das auf jeder Seite bei der Anzeige und beim Druck unterlegt wird, wurde nun auch ein seitenspezifisches Wasserzeichen implementiert. Dieses kann für jede Seite einzeln bestimmt werden.

### **Größe für Texte und Eingabeobjekte im Eigenschaftendialog einstellbar**

Für Text- und Eingabeobjekte werden Höhe und Breite jetzt im Eigenschaftendialog zur Änderung angeboten. Dadurch kann insbesondere die Standardgröße dieser Objekttypen vorgegeben werden. Diese wird verwendet, wenn ein Objekt dieses Typs neu angelegt wird.

### **Wasserzeichen werden nach PDF exportiert.**

Wasserzeichen werden jetzt auch nach PDF exportiert. In der exportierten PDF-Datei werden die Objekte der Wasserzeichen den eigentlichen Seitenobjekten vorangestellt.

### **Schaltflächen zum Ausrichten am unteren und oberen Rand**

Im Editor wurden zwei weitere Schaltflächen zum Ausrichten von Objekten am unteren und am oberen Rand hinzugefügt.

### Erweiterung des Exports in VCF-Dateien

Beim Exportieren eines Formulars in eine VCF-Datei wird nun auch der Eintrag FN:Name geschrieben. Damit kann der Anzeigename einer Person definiert werden.

## Build 6.4.3 (01.01.2013)

### Datumpicker werden nicht nach PDF exportiert

Seit Version 6.3.0 wurden Datumpicker nicht immer korrekt nach PDF exportiert. Der Fehler wurde behoben.

### Linien nach oben verschieben

Markierte Linien konnten seit Version 6.3.0 mittels der Cursortasten nicht mehr nach oben verschoben werden. Der Fehler wurde behoben.

### Objekte können nicht nach oben oder links vergrößert werden

Objekte konnten mit der Maus seit Version 6.3.0 nicht mehr nach oben oder links vergrößert werden. Der Fehler wurde behoben.

### Setzen der Eingabereihenfolge übergeht nicht betretbare Eingabeobjekte

Nicht betretbare Eingabeobjekte wurden seit der Version 6.3.0 beim Anordnen der Eingabereihenfolge nicht mehr berücksichtigt. Der Fehler wurde behoben.

## Build 6.4.2 (04.12.012)

### Neuer Makrobefehl zum Starten des Standardbrowsers

Es wurde ein neuer Makrobefehl implementiert, mit dem der Standardbrowser gestartet und eine beliebige URL angezeigt werden kann. Der Befehl heißt *OpenBrowser* und ist in der Dokumentation *Allgemeine Makrobefehle* beschrieben.

### Ligaturen „fl“ und „fi“ in Unicode-RTF-Dateien

In Unicode-RTF-Dateien wurden die Ligaturen „fl“ und „fi“ nicht korrekt konvertiert. Der Fehler wurde behoben.



### Verschlüsseln von Formularen

Seit Version 6 werden Formulare im XML-Format gespeichert. Damit Formulare weiterhin geschützt werden können, wurde in der Funktion *Passwortschutz* im Menü *Formular* die Option *Formulardatei verschlüsseln* implementiert.

Beim Öffnen eines verschlüsselten Formulars wird dieses automatisch entschlüsselt. Die Verschlüsselung dient nur dazu, die klartextliche Lesbarkeit der Formulardatei zu verhindern.

## Build 6.4.1 (09.11.2012)

### Erweiterung der API-Funktion *Form\_FindNextObject* für Wasserzeichen

Die API-Funktion *Form\_FindNextObject*, die Objekte im gesamten Formular oder auf einzelnen Formularseiten sucht, wurde dahingehend erweitert, dass auch Objekte in Wasserzeichen gesucht werden können. Dadurch können jetzt auch Objekte in Wasserzeichen ausgelesen oder modifiziert werden. Um Objekte in Wasserzeichen zu suchen, muss der Seitenname "watermark" angegeben werden.

### Fehler im Datumspicker behoben

In einem Eingabefeld mit Datumspicker konnte der aktuelle Tag im Gegensatz zu allen anderen Tagen nicht durch Klicken in das zugehörige Eingabefeld übertragen werden. Der Fehler wurde behoben.

### Anzeigefehler beim Zuordnen von Objekten zu einer Zone behoben

Wenn markierte Objekte der sie umschließenden Zone zugeordnet wurden oder die Zuordnung aufgehoben wurde, ist die Anzeige nicht erneuert worden, so dass die Aktion zwar durchgeführt wurde, aber nicht sichtbar war. Der Fehler wurde behoben.

### Fehler beim Ändern des Zonenabstands behoben

Wenn im Eigenschafts-Dialog einer Zone ein Abstand zur vorherigen Zone vorgegeben wurde, ging der Wert beim Speichern verloren. Der Fehler wurde behoben.

### Geändertes Verhalten bei der Vorgabe der Eingabereihenfolge

Wenn während der Vorgabe der Eingabereihenfolge einer Seite in ein Objekt einer anderen Seite geklickt wurde, passierten ungewollte Effekte. Dies wird nun geprüft. Es wird ein Warnsignal ausgegeben, wenn während der Vorgabe der Eingabereihenfolge in eine andere als die aktive Seite geklickt wird.

## Build 6.4.0 (01.11.2012)

### Export von Seitenzonen nach HTML und PDF

Die in Version 6.3.0 neu implementierten Seitenzonen werden jetzt auch nach HTML und PDF exportiert.

### Fehler in der Wochentagberechnung behoben

Beim Export nach PDF wurden Wochentage mittels Javascript nicht korrekt berechnet. Der Fehler wurde behoben.

## Build 6.3.0 (01.10.2012)

### Seitenzonen

Ab der Version 6.3.0 ist *eforms & more* mit einem neuen Werkzeug ausgestattet, den sogenannten Seitenzonen.

Seitenzonen sind horizontale Bereiche innerhalb einer Seite, die eine vorgegebene Höhe und immer die volle Breite der Seite haben. Alle Objekte des Formulars können jetzt wahlweise einer Seite oder der Zonen auf der Seite zugeordnet werden.

Wenn Seitenzonen verschoben werden, verschieben sich dabei auch alle Objekte der Zone.

Die wesentliche Funktion der Seitenzonen ist, dass diese während des Ausfüllens des Formulars durch Knopfdruck oder in Abhängigkeit vom Inhalt anderer Objekte auf- oder zugeklappt werden können. Wenn eine Seitenzone zugeklappt ist, sind die Objekte nicht mehr sichtbar und können auch nicht mehr betreten oder ausgefüllt werden. Ihre Inhalte gehen aber beim Zuklappen nicht verloren und sind weiterhin vorhanden, wenn eine Zone erneut aufgeklappt wird.

Seitenzonen können eine statische oder dynamische Position haben. Wenn die Position dynamisch ist, werden weiter unten stehende Seitenzonen automatisch nach oben gezogen wenn darüber eine andere Seitenzone zugeklappt wird.

Dadurch ist die Gestaltung interaktiver Formularstrukturen möglich.

Alle Funktionen für die Seitengestaltung finden Sie in dem neuen Hauptmenüeintrag *Seitenzonen*.

### XML Format für Formulardateien

Ab der Version 6.3.0 werden Formulardateien grundsätzlich im XML-Format abgelegt. Bisher galt dies nur für Dateien mit der Erweiterung *\*.xml*. Formulardateien früherer Versionen werden automatisch konvertiert. Die Namensweiterung *\*.fvl* wird aber weiterverwendet.

### XML Format in der Zwischenablage

Objekte werden ab Version 6.3.0 im XML-Format in die Zwischenablage geschrieben. Dadurch kann der Inhalt der Zwischenablage insbesondere auch in jedem Texteditor verwendet werden.

### Verlaufsfarben für Hintergründe

Der Hintergrund aller rechteckigen Objekte kann mit einstellbaren Verlaufsfarben dargestellt werden. Der Verlauf kann horizontal oder vertikal eingestellt werden. Diese Eigenschaft kann nicht nach PDF exportiert werden.

### Neue Eigenschaften-Dialoge für alle Objekte

Die Dialogfenster zur Einstellung der Eigenschaften von Objekten wurden für alle Objekttypen neu gestaltet. Es wurden zahlreiche optische Verbesserungen eingebracht. Ferner wurden die Kontrollelemente für die Verlaufsfarben hinzugefügt.

### Wasserzeichen

Im Menü *Formular* wurde eine neue Funktion *Wasserzeichen* implementiert. Hier kann der Name einer einseitigen Formulardatei angegeben werden, die als Wasserzeichen verwendet werden soll. Wenn eine gültige Wasserzeichendatei konfiguriert ist und zum Zeitpunkt der Bearbeitung existiert und zugreifbar ist, wird das Wasserzeichen auf jeder Seite des bearbeiteten Formulars unterlegt.

### Tasten als Hotkey an die Host-Anwendung senden

Es wurde eine neue API-Funktion *Form\_WantHotKey* implementiert, mit der bestimmte Tasten oder Tastenkombinationen als Hotkey deklariert werden können. Tasten, die als Hotkey deklariert sind, werden nicht mehr vom Formular interpretiert, sondern als Notifikation an die Host-Anwendung gesendet.

### Objekthintergrund über die Zwischenablage kopieren

Die Funktion *Hintergrundfarbe in Zwischenablage kopieren* wurde erweitert. Es wird sofort nicht nur die Hintergrundfarbe sondern die vollständige Hintergrundgestaltung in die Zwischenablage übernommen. Entsprechend werden alle diese Komponenten beim Einfügen

auf die entsprechenden Zielobjekte übertragen. Dadurch können insbesondere auch Verlaufsfarben und Hintergrundtyp transportiert werden.

### Erweiterte Kontextmenüs

Die Kontextmenüs wurden um einige sinnvolle Funktionen erweitert. Insbesondere wurde das Angleichen der Größen bei mehreren markierten Objekten als Menüfunktion hinzugefügt.

### Vereinfachung der API-Funktionen

Viele API-Funktionen für die Einbindung in andere Projekte wurden vereinfacht und übersichtlicher gestaltet. Die Dokumentation wurde vollständig überarbeitet. Alle Funktionen sind aber vollständig kompatibel zu früheren Versionen.

### Formular bereinigen

Die bisher im Menü *Objekte* angesiedelte Funktion *Stapelverarbeitung...* wurde erweitert und umbenannt. Sie findet sich jetzt im Menü *Extras* unter dem Namen *Formular bereinigen...* Sie wurde um die Möglichkeit erweitert, Objekte, die ganz oder teilweise außerhalb von Zeiten oder Zonen liegen, in die entsprechende Seite oder Zonen zu bewegen. Dadurch können Objekte, die „verlorengegangen“ waren, z.B. weil sie weit außerhalb einer Seite gelandet sind, wieder bearbeitet werden.

## Build 6.2.3 (23.08.2012)

### Benutzerdefinierte Funktion *OnSubmit* beim HTML-Export

Beim Exportieren eines Formulars nach HTML kann die Funktion *OnSubmit*, die im JavaScript automatisch generiert wird, nun auch benutzerdefiniert vorgegeben werden.

### JavaScript Code beim Ereignis *OnClick* im HTML-Export

Der JavaScript-Code welcher für ein Objekt für das Ereignis *OnClick* im Formular hinterlegt ist, wurde beim Export nach HTML nicht korrekt erzeugt. Der Fehler wurde behoben.

## Build 6.2.2 (17.08.2012)

### Mehr Notifikationen beim Ändern von Ankreuzfeldern

Die Notifikation, die gesendet wird, wenn ein Ankreuzfeld sich ändert wird ab sofort auch gesendet, wenn Ankreuzfelder durch Makrobefehle verändert werden. Sollten mehrere An-

kreuzfelder gleichzeitig geändert werden, wird für jedes Ankreuzfeld eine separate Notifikation gesendet.

### **Build 6.2.1** (01.07.2012)

#### **Notifikation wenn ein Objekt den Fokus erhält**

Es wurde eine neue Notifikation implementiert, die immer dann gesendet wird, wenn ein Eingabeobjekt den Fokus erhält. Dabei ist es gleichgültig, wie das Feld den Fokus erlangt. Die Notifikation wird insbesondere auch dann gesendet, wenn das Formularfenster in den Vordergrund kommt und dabei das aktive Feld den Fokus erhält.

Die Notifikation hat den Namen *FORMMSG\_ OBJECT\_GETS\_FOCUS*.

Der Wert der Notifikation ist 162.

Das Objekt wird als Parameter übergeben.

#### **Notifikation wenn in einem Objekt ein Doppelklick stattfindet**

Es wurde eine neue Notifikation implementiert, die immer dann gesendet wird, wenn in einem Eingabeobjekt ein Doppelklick ausgeführt wird.

Die Notifikation hat den Namen *FORMMSG\_ LBUTTONDBLCLK\_OBJECT*.

Der Wert der Notifikation ist 146.

Das Objekt wird als Parameter übergeben.

### **Build 6.2.0** (14.06.2012)

#### **Trennung der Lizenznummern von *Editor* und *Ausfüller***

Ab Version 6.2.0. werden getrennte Lizenznummern für die Komponenten Editor und Ausfüller vergeben. Der Grund ist, dass der Ausfüller jetzt auch als eigenständiges Produkt vertrieben wird. Dadurch können Kunden, die mehrere Ausfüller-Lizenzen, aber nur eine Editor-Lizenz benötigen, Lizenzkosten sparen.

Da der Ausfüller nach wie vor im Lizenzumfang des Gesamtpakets enthalten ist, erhalten Kunden, die eine Vollversion von [e] forms & more erwerben, ab sofort zwei Lizenznummern - je eine für den Editor und eine für den Ausfüller.

**Build 6.1.0** (07.06.2012)**Fehlerbehebung**

Es wurde ein Fehler bei der Ausgabe von Umlauten in RTF-Texten behoben.

**Build 6.0.2** (21.05.2012)**Proxy Server Unterstützung bei Aktivierung**

Bei der Aktivierung der Software wird auf HTTP-Server zugegriffen. Hier werden künftig auch Proxy-Server unterstützt.

**Build 6.0.1** (08.05.2012)**Neue Funktion *Schriftfarben ändern***

Es wurde eine neue Funktion implementiert, mit der die Schriftfarbe aller selektierten Objekte geändert werden kann. Die Farbe kann ein beliebiger RGB-Wert sein. In frühere Versionen konnten nur 16 Farben für Schriften verwendet werden.

**Build 6.0.0** (25.04.2012)**Datumpicker für jedes Eingabefeld schaltbar**

Es kann für jedes Eingabefeld einzeln bestimmt werden, ob beim Betreten des Feldes automatisch ein Datumpicker eingeblendet wird. Die Möglichkeit, Datumpicker global für das gesamte Formular auszuschalten, bleibt hiervon unberührt.

Diese Eigenschaft wird auch beim Export nach PDF berücksichtigt.

**Neues Lizenzsystem**

[e] forms & more wird ab Version 6.0.0 mit einem Lizenzschlüssel online freigeschaltet. Als Inhaber einer freigeschalteten Version können Sie künftig neue Versionen kostenlos herunterladen und nutzen, ohne dafür Lizenzgebühren zu entrichten. Erst wenn sich die erste Ziffer der Versionsnummer ändert, wird eine Upgrade-Lizenz benötigt.

**Neues Produktinformationsfenster**

Das Fenster für die Produktinformation im Menü ? wurde vollständig erneuert. Insbesondere kann die Aktivierung oder Deaktivierung von [e] forms & more auf einem Arbeitsplatzrechner direkt von hier vorgenommen werden.

## Version 5.9

### Build 5.9.10 (15.04.2012)

#### Bearbeitung des Gruppennamens in der Sammelbearbeitung

Der Name der Gruppe, der ein Objekt angehört, wird ab sofort als zusätzliche Spalte in

### Build 5.9.10 (15.04.2012)

#### Bearbeitung des Gruppennamens in der Sammelbearbeitung

Der Name der Gruppe, der ein Objekt angehört, wird ab sofort als zusätzliche Spalte in der Sammelbearbeitung für Objekte angeboten. Auch beim Importieren und Exportieren der Sammelbearbeitungstabelle wird der Gruppenname berücksichtigt.

### Build 5.9.9 (13.02.2012)

#### Selbstausschaltende Ankreuzfelder ausschalten

Wenn mehrere Ankreuzfelder die Eigenschaft *Ankreuzfelder der gleichen Gruppe auslösen* haben und der selben Gruppe angehören, so lösen diese sich beim Ausfüllen gegenseitig aus. Es war bisher aber nicht möglich, ein Ankreuzfeld mit dieser Eigenschaft wieder auszuschalten.

Wenn die Eigenschaft *Mindestens ein Feld muss angekreuzt sein* nicht eingeschaltet ist, ist es ab sofort möglich, ein angekreuztes Feld wieder auszuschalten. Die übrigen Felder der Gruppe bleiben hiervon unberührt.

#### Datumpicker für PDF-Export

Wenn unter *Laufzeiteinstellungen* im Menü *Formular* die Verwendung eines Datumpickers eingeschaltet ist, wird immer beim Betreten eines Eingabefeldes vom Typ *Datum* automatisch ein Datumpicker eingeblendet.

Diese Funktionalität ist ab sofort auch in der PDF-Datei verfügbar, wenn ein Formular als PDF exportiert wird.



### Fehlerbehebung beim Export von RTF-Objekten nach PDF

Der Export eines RTF-Objekts mit einem 1,5-zeiligen Text in eine PDF-Datei hatte einen Fehler. Dieser wurde behoben.

## Build 5.9.8 (21.01.2012)

### Fehler im *Change-Flag* behoben

Das *ChangeFlag* eines Formulars, welches anzeigt, ob ein Formular verändert worden ist, wurde nicht korrekt gesetzt. Dadurch funktionierten einige Makrobefehle nicht korrekt. Der Fehler wurde behoben.

## Build 5.9.7 (13.12.2011)

### Automatische Schriftverkleinerung bei Druck und Anzeige

Es kann im Menü *Formular* unter *Laufzeiteinstellungen* eine neue Option gesetzt werden, die bewirkt, dass bei Anzeige und Druck eines Formulars die Schriftgröße in den Objekten automatisch so weit verkleinert wird, dass der vollständige Inhalt jedes Objekts dargestellt wird. Dadurch kann gewährleistet werden, dass keine wesentliche Textinformation abgeschnitten wird.

Die Verkleinerung erfolgt nur beim Drucken und Anzeigen. Die im Objekt eingestellte Schriftgröße wird nicht verändert.

## Build 5.9.6 (31.10.2011)

### Lange Vorbesetzungen und Hilfetexte beim Speichern im XML-Format abgeschnitten

Beim Speichern von Formularen im XML-Format wurden verschiedene Textelemente ungewollt abgeschnitten, wenn diese eine bestimmte Größe überschritten. Dieser Fehler trat insbesondere bei Hilfetexten und Vorbesetzungen von Eingabefeldern auf. Der Fehler wurde behoben.

### Gepunktete Linien beim Export nach PDF

Beim Export eines Formulars nach PDF werden gepunktete Linien jetzt korrekt exportiert.

## Build 5.9.5 (19.10.2011)

### Anzeigen veränderter Seiten

Es wurde eine neue Option implementiert, die oben rechts auf einer Seite eine Markierung anzeigt, wenn ein Objekt auf dieser Seite verändert wurde. Die Option kann im Dialog für *Anzeigeeinstellungen* im Menü *Formular* eingeschaltet werden. Aus Gründen der Kompatibilität ist die Funktion als Voreinstellung ausgeschaltet.

### Neue Makrobefehle

Es wurden folgenden neuen Makrobefehle implementiert:

- `GetPageNrByName`
- `SetRuntimeChangeFlag`
- `GetRuntimeChangeFlag`
- `CountRuntimeChangedPages`

Mittels des Befehl *GetPageNrByName* kann die Nummer einer Seite auf Basis ihres Namens ermittelt werden.

Mittels der Befehle *SetRuntimeChangeFlag* und *GetRuntimeChangeFlag* lassen sich die Zustandsvariablen, in denen nachgehalten wird, ob eine einzelne Seite verändert wurde, abfragen und setzen.

*CountRuntimeChangedPages* gibt Auskunft darüber, auf wie vielen Seiten zur Laufzeit die Inhalte von Objekten verändert wurden.

### Neue Druckoption

Beim Drucken kann anstelle eines Seitenbereiches künftig auch bestimmt werden, dass nur alle veränderten Seiten gedruckt werden. Diese Eigenschaft kann wie alle anderen Druckoptionen auch als Voreinstellung im Formular hinterlegt werden.

## Build 5.9.4 (14.10.2011)

### Neues Look andFeel

Das *Look andFeel* der Programmoberfläche wurde den neuen Windows-Versionen angepasst. Das Menü und alle Dialoge haben somit ein neues Erscheinungsbild.

### Build 5.9.3 (12.10.2011)

#### Neuer Zoom-Modus

Es wurde ein weiterer Zoom-Modus implementiert. Dieser erlaubt die Einstellung des Zoom-Faktors in der Form, dass zwei Seitenbreiten nebeneinander in das Fenster passen. Dieser Modus unterscheidet sich vom Modus *Zwei Seiten* dadurch, dass beide Seiten nebeneinander die Fensterfläche ausfüllen, auch wenn die Seiten nicht in der vollen Höhe dargestellt werden können, da das Fenster nicht hoch genug ist.

### Build 5.9.2 (07.10.2011)

#### Fehler bei der Anzeige von RTF-Objekten

Es hat sich ein Fehler eingeschlichen, der dazu führte, dass RTF-Objekte nicht in allen Fällen angezeigt wurden. Der Fehler wurde behoben.

### Build 5.9.1 (07.09.2011)

#### Funktion zur Stapelbearbeitung von Objekten

Es wurde eine neue Funktion *Stapelbearbeitung* im Menü *Objekte* des Editors implementiert. Diese ermöglicht es, alle Objekte des gesamten Formulars gleichzeitig zu bearbeiten. Folgende Optionen werden zur angeboten:

- Objekte, die über einen der Seitenränder hinausragen werden so verkleinert, dass dies nicht mehr der Fall ist
- Objekte, die ganz außerhalb einer Seite liegen, werden gelöscht

Damit können Formulare *aufgeräumt* werden. Beide Eigenschaften sollten eigentlich nicht vorkommen. Wenn Objekte außerhalb der Seite stehen, sind dies in der Regel Relikte, die versehentlich übriggeblieben sind und die nicht entfernt wurden, weil sie nicht sichtbar sind.

#### Keine Datums picker in schreibgeschützten Feldern

Der Datums picker wird ab sofort in Eingabefeldern, die nicht verändert werden können (schreibgeschützt) nicht mehr aufgeblendet, da dies keinen Sinn macht.

## Build 5.9.0 (01.09.2011)

### Mehrseitendarstellung

Editor und Filler wurden mit einer komfortablen Mehrseitenanzeige ausgestattet. Es kann ab sofort zwischen folgenden Anzeigeeinstellungen direkt per Knopfdruck umgeschaltet werden:

- Halbe Seitenbreite
- Ganze Seitenbreite
- Eine Seite
- Zwei Seiten
- Drei Seiten
- Viele Seiten

Wenn das Formular weniger Seiten enthält, als angezeigt werden sollen, wird automatisch auf die nächst niedrigere Seitenzahl umgeschaltet.

### Erweiterung der automatischen Zoom-Einstellungen

Im Zuge der Mehrseitendarstellung wurden auch das Automatische Zoomen erweitert. Im Menü *Formular* kann unter *Anzeigeeinstellungen* ab sofort auch die mehrseitige Anzeige beim Öffnen eingestellt werden.

### Erweiterung und Überarbeitung der Formularbibliothek

Die herstellerseitig mitgelieferte Formularbibliothek wurde umfangreich erweitert. Mehrseitige Formulare wurden so modifiziert, dass sie beim Öffnen automatisch in der Mehrseitenanzeige dargestellt werden.

## Version 5.8

### Build 5.8.5 (27.08.2011)

#### Abgeschnittene Texte in mehrzeiligen Eingabefeldern

In mehrzeiligen Eingabefeldern trat der Effekt auf, dass Textteile am Ende des Eingabefeldes je nach Vergrößerungsstufe der Darstellung oder beim Drucken abgeschnitten wurden, wenn das Eingabefeld vollständig gefüllt war. Das Problem für diesen Effekt ist systematischer Natur und liegt in der Skalierung von Schriftarten. Diese ist nicht immer proportional zum umgebenden Rechteck.

Damit wesentliche Dateninhalte - vor allem beim Ausdruck - auf jeden Fall dargestellt werden, wurde ein Algorithmus implementiert, der die Schrifthöhe bei der Ausgabe automatisch um einen Punkt verkleinert, falls dieser Effekt auftritt, damit das Problem kompensiert wird.

### Build 5.8.4 (22.08.2011)

#### Datumpicker rechts oder unten

Der in Version 5.8.3 implementierte Datenpicker für Datumsfelder wird ab dieser Version unterhalb anstatt rechts des betroffenen Datums-Eingabefeldes eingeblendet.

Es kann jedoch für jedes Eingabefeld einzeln festgelegt werden, ob die Einblendung rechts oder unter erfolgen soll. Auf diese Weise kann das Design des Formulars berücksichtigt werden.

#### Neuer Makrobefehl zum Anlegen von Verzeichnissen

Es wurde ein neuer Makrobefehl *CreateDirectory* implementiert. Mit Hilfe dieses Befehls können Verzeichnisse oder Verzeichnisketten angelegt werden. Der Befehl ist in der Dokumentation *intfunc.pdf* der Makrobibliothek *Winculadokumentiert*.

#### Benutzervorgabe für Fehlermeldungen in Skripten

Fehlermeldungen, die in Skripten (z.B. JavaScript in HTML oder PDF) eingebettet werden, können ab sofort durch anwenderspezifische Texte ersetzt werden. Hierzu können vom Verisierten Anwender Eintragungen in der vorgenommen werden. Im Verzeichnis:

```
HKEY_CURRENT_USER\Software\Waimea Software\eforms& more\General\ScriptErrorMessages
```

können folgende Werte angelegt werden:

BadDate	Meldung für fehlerhafte Datumseingaben
BadTime	Meldung für fehlerhafte Zeiteingaben
BadFloat	Meldung für fehlerhafte Gleitkommazahleingaben
BadInteger	Meldung für fehlerhafte Ganzzahleingaben

Wenn keine Meldungen eingetragen sind, werden wir bisher die vorgegebenen Texte verwendet.

### Fehler beim Umwandeln von Objekten behoben

Wenn Objekte, die nicht über eine Schriftart verfügen (z.B. Rechtecke oder Linien), in Objekte umgewandelt wurden, die eine Schriftart benötigen (z.B. Texte, Eingabefelder, etc.), blieb der Editor beim Bearbeiten der umgewandelten Objekte stehen. Der Fehler wurde behoben.

### Überflüssige Menüeinträge entfernt

Im Menü *Extras* des Editors wurden einige überflüssige Menüeinträge entfernt.

### Fehler beim Bearbeiten der Voreinstellung für Objekte behoben

Im Menü *Extras* des Editors können die Standardwerte für verschiedene Objekte eingestellt werden. Wenn dabei auch der Name des Objekts definiert wurde, konnte die entsprechende Funktion danach nie wieder aufgerufen werden. Der Fehler wurde behoben. Die Einstellung eines Namens als Voreinstellung für Objekte wurde unterbunden.

### Fehler beim Anzeigen des Lizenzschlüssels behoben

Im Info-Fenster des Editors wurde der Lizenzschlüssel nicht korrekt angezeigt. Der Fehler wurde behoben.

## Build 5.8.3 (14.07.2011)

### Datumpicker für Datumsfelder

Für Eingabefelder vom Typ *Datum* wurde eine neue Option implementiert, die beim Betreten des Eingabefeldes automatisch einen *Datumpicker* einblendet. Hier kann das Datum durch Klicken mit der Maus ausgewählt werden. Dabei ist es gleichgültig, ob das Feld mittels der TAB-Taste oder durch Klicken in dasselbe betreten wird.

Beim Verlassen des Feldes wird der Datumpicker automatisch wieder ausgeblendet.

Die Funktion ist unter dem Menüpunkt *Formular->Laufzeiteinstellungen* ein- oder ausschaltbar. Die Voreinstellung ist *ein*.

### API-Schnittstelle zum Öffnen von Formularen ohne Locking

Für die API-Schnittstelle *Form\_Open\_Expanded* wurde ein Schalter implementiert. Der es ermöglicht, Formulare ohne Locking zu öffnen.

## Build 5.8.2 (22.05.2011)

### Erweiterte Sammelbearbeitung

Die Sammelbearbeitung für Objekte wurde erweitert. Es können jetzt auch alle Objekte aller Seiten gleichzeitig bearbeitet werden. Die Bearbeitung der Objekte der aktuellen Seite ist als weiterer Menübefehl erhalten geblieben. Die Funktion kann auch rückgängig gemacht werden.

### Neue Funktion: *Seite duplizieren*

Mittels dieser Funktion kann die aktuelle Seite mit allen Objekten dupliziert werden.

## Build 5.8.1 (30.04.2011)

### Eingabefelder im HTML-Format

Die API-Schnittstelle *FormObj\_SetText* wurde dahingehend angepasst, dass man mittels dieser Schnittstelle auch HTML-Texte in HTML-Eingabefelder einsetzen kann.

Wenn Text in HTML-Eingabefeldern mit einer Hintergrundfarbe versehen wird, wird dieser jetzt auch gespeichert und wieder eingelesen.

Zeilenschaltungen im HTML-Format werden ab sofort durch `<br/>` anstelle von `<br>` codiert.

Wenn ein HTML- oder RTF-Eingabefeld das Attribut *nur lesen* hat, wird das Kontextmenü, welches das Verändern von Schriftattributen ermöglicht, unterdrückt.

Die Hintergrundfarbe von HTML- oder RTF-Eingabefeldern wird auch dann korrekt dargestellt, wenn man sich in der Eingabe befindet. Bisher wurde ein weißer Hintergrund verwendet, auch wenn eine andere Farbe eingestellt war.

**Build 5.8.0** (28.04.2011)**Eingabefelder im HTML-Format**

Für Eingabefelder kann ab sofort spezifiziert werden, in welchem Format der Text abgelegt wird. Folgende Formate stehen zur Auswahl:

- Reiner Text (Plain Text)
- RTF (Rich Text Format)
- HTML

Wenn RTF oder HTML gewählt wird, können für markierte Bereiche Schriftattribute mittels der rechten Maustaste vergeben werden.

Die Ablage als HTML formatiert attributierte Textpassagen mittels des `<span>`-Tags und entsprechenden CSS-Styles.

Diese Eigenschaft wird beim Export nach PDF nicht berücksichtigt.

**Fehlerbehebung beim Export von Buttons mit Grafiken nach PDF**

Beim Export von Buttons mit Grafiken nach PDF wurde ein Fehler bei der Größenberechnung behoben.



## Version 5.7

### Build 5.7.2 (05.04.2011)

#### Fehlerbehebung

Wenn das Verzeichnis c:\temp auf der Maschine, auf der die Software installiert ist, nicht existiert, wurde eine Fehlermeldung ausgegeben. Der Fehler wurde behoben.

### Build 5.7.1 (01.04.2011)

#### Namensvergleiche unabhängig von Groß- und Kleinschreibung

Bisher wurde bei namentlicher Bezugnahme auf Objekte eines Formulars immer die Groß- und Kleinschreibung der Namen berücksichtigt. Das bedeutet, dass zwei Objekte mit den Namen *testund* *TEST* als verschiedene Objekte behandelt wurden.

Dieses Verhalten kann ab sofort vom Benutzer abgeschaltet werden. Dadurch werden Objekte, die sich im Namen nur durch Groß- und Kleinschreibung unterscheiden, als namentlich gleiche Objekte behandelt.

Die Einstellung kann im Editor durch den Anwender in der Funktion *Laufzeiteinstellungen* im Menü *Formular* vorgenommen werden.

Entwickler können diese Eigenschaft durch Aufruf der API-Funktionen:

```
Form_SetXmlPropertyByAttrName (hForm, "NamesCaseIns", "1")
```

```
Form_SetXmlPropertyByAttrName (hForm, "NamesCaseIns", "0")
```

ein oder ausschalten.

Insbesondere betroffen sind die Bezüge auf Namen von Objekten, Seiten und Makroereignissen.

#### Erweitertes Auslösen von Ankreuzfeldern einer Gruppe

Ankreuzfeldern mit der Eigenschaft *Felder der gleichen Gruppe auslösen*, entfernen beim Ausfüllen automatisch das Kreuz in allen Ankreuzfeldern, die der gleichen Gruppe zugeordnet sind.

Dies galt bisher nur für Ankreuzfelder, die sich auf derselben Seite befanden. Die Funktion wurde dahingehend erweitert, dass dies nun für alle Ankreuzfelder des gesamten Formulars zutrifft.

## **Build 5.7.0** (15.03.2011)

### **Kopieren von Hintergrundfarben**

Im Editor wurden Menüfunktionen implementiert, mit denen die Hintergrundfarbe eines Objekts in die Zwischenablage kopiert und anschließend auf alle markierten Objekte angewendet werden kann.

### **Neue Funktion *Objekte umwandeln***

Es wurde eine neue Funktion implementiert, die es erlaubt, alle markierten Objekte in Objekte eines anderen Typs umzuwandeln. Ausgenommen von der Umwandlung sind Datenzugriffsobjekte und Linien. Diese werden nicht umgewandelt.

Achtung: Beim Umwandeln können typspezifische Objekteigenschaften verloren gehen, da nicht sicher ist, ob der neue Objekttyp auch über diese Eigenschaften verfügt.

Alle gemeinsamen Eigenschaften des neuen und alten Objekttyps (Farbe, Größe, etc.) werden übernommen.

### **Erweiterte Darstellungsmöglichkeiten des aktiven Eingabefeldes**

Für die Hervorhebung des augenblicklich aktiven Eingabefeldes kann jetzt eine der folgenden Varianten eingestellt werden:

- a) *Keine Hervorhebung*  
Das aktive Eingabefeld wird farblich nicht verändert, wenn es aktiv ist
- b) *Leichte farbliche Hervorhebung*  
Das aktive Eingabefeld wird farblich etwas hervorgehoben. Die Hervorhebung erfolgt Ton in Ton mit der Farbe, die das Feld hat, wenn es nicht aktiv ist. Helle Felder werden etwas dunkler dargestellt und dunkle Felder etwas heller.
- c) *Explizite Farbvorgabe*  
Es kann eine explizite Farbe vorgegeben werden, in der aktive Eingabefelder dargestellt werden

### Rich Text in Eingabefeldern

Für Eingabefelder vom Datentyp *Text* wurde die Möglichkeit implementiert, anstelle eines einfachen Textes auch Text mit Schriftattributen (fett, kursiv, etc.) eingeben und ablegen zu können.

Die Ablage des attributierten Textes kann in RTF-Syntax oder in HTML-Syntax erfolgen. Diese Eigenschaft wird nicht nach PDF exportiert.

## Version 5.6

### Build 5.6.4.0<sub>(26.01.2011)</sub>

#### Neue Formular-Makrobefehle

Es wurden vier neue Makrobefehle implementiert, die das Auslesen und Setzen von Seiteneigenschaften erweitern:

- GetPageBgrMode
- SetPageBgrMode
- GetPageBgrColor
- SetPageBgrColor

Die Dokumentation *Makrofunktionen [e]forms & more* beschreibt die Makrobefehle mit allen Parametern.

### Build 5.6.3.0

#### Neue Beispielvorlage *Jahreskalender*

Die Beispielvorlagen wurden um einen Jahreskalender erweitert. Dieser hebt Sonntage und bundeseinheitliche Feiertage farblich hervor.

#### Neue allgemeine Makrobefehle

Es wurden folgende neue allgemeine Makrobefehle für die Datumsarithmetik implementiert:

GetWeekdayFromYMD	Liefert den Wochentag eines Datums
GetMonthLng	Liefert die Anzahl Tage eines Monats
IsSunday	Liefert zurück, ob ein Datum ein Sonntag ist
IsHoliday	Liefert zurück, ob ein Datum ein Feiertag ist

Die Dokumentation *Allgemeine Makrofunktionen* beschreibt die Makrobefehle mit allen Parametern.

### Neue Formular-Makrobefehle

Es wurden zwei neue Makrobefehle implementiert, um Eigenschaften von Objekten zu setzen oder auszulesen:

GetXmlProperty	Liefert eine beliebige Objekteigenschaft als XML-Wert
SetXmlProperty	Setzt eine beliebige Objekteigenschaft als XML-Wert

Die Dokumentation *Makrofunktionen [e]forms & more* beschreibt die Makrobefehle mit allen Parametern.

## Build 5.6.2.0

### Fehlerbehebungen

Das Import-Modul für PDF-Dateien aus dem AdobeAcrobat wurde in einem falschen Verzeichnis installiert. Der Fehler wurde beoben.

## Build 5.6.1.0

### Neue Makrobefehle

Es wurde ein neuer Makrobefehl *SetPictureFile* implementiert, mit dem der Name und Pfad der Bilddatei in Bildobjekten gesetzt werden kann ,deren Bildquelle sich in einer externen Datei befindet.

### Erweiterte Makrobefehle

Ferner wurde der Makrobefehl *GetStringProperty* um den Parameter *DataSourceFile* erweitert. Mittels des neuen Parameters kann aus Datenzugriffsobjekten der Name und Pfad der Datenquellen-Datei ermittelt werden. Dies ist im Falle von Datenbanken, der Name der Datenbank-Datei.

### Neue Objektbedeutungen

Für den Export von Formularen in Datenformate, in denen die semantische Bedeutung von Objekten relevant ist (z.B. VCF-Dateien), kann jedem Objekt eine semantische Bedeutung zugeordnet werden.

eforms& more verfügt über eine vorgegebene Liste von semantischen Bedeutungen. Diese Liste wurde um folgende Begriffe erweitert:

- Foto der Person

- Straße der Privatadresse
- Postleitzahl der Privatadresse
- Ort der Privatadresse

Durch diese Erweiterung können z.B. beim Export von Formularen in Visitenkarten (VCF-Dateien) jetzt auch ein Foto der Person sowie die postalische Anschrift exportiert werden.

### Semantische Bedeutung für Bildobjekte

Die Zuordnung einer semantischen Bedeutung kann ab sofort auch zu einem Bildobjekt erfolgen. Bisher konnten nur Eingabeobjekte semantisch zugeordnet werden.

Durch diese Erweiterung kann z.B. beim Export von Datensätzen in Visitenkarten (VCF-Dateien) jetzt auch ein Foto der Person exportiert werden.

### Export von Labels nach HTML

Ankreuzfelder, die Labels enthalten, werden ab sofort auch in exportierten HTML-Dateien korrekt dargestellt. Bisher wurden die Labels ignoriert.

### Standardwerte beim Anlegen von Rechtecken

Wenn im Editor ein neues Rechteck angelegt wird, werden ab sofort die Standardwerte für ein flaches, weißes, umrandetes Rechteck verwendet und nicht wie bisher die Einstellungen eines 3D-Rechtecks.

### Button zum Anlegen von Rechtecken

Im Editor wurde ein neuer Button hinzugefügt, mit dem der Anwender direkt und ohne weitere Rückfragen ein neues Rechteck anlegen kann.

### Sammelbearbeitung von Objekten

In den Funktionen *Objekte ändern (Sammelbearbeitung)* und *Markierte Objekte ändern (Sammelbearbeitung)* wird in der Spalte *Text / Inhalt* bei Objekten, die über keinen eigenen Objekttext verfügen, eine andere sinnvolle Objektkomponente zur Bearbeitung angeboten:

- Bei Eingabefeldern und Auswahlfeldern wird der *Standardwert* bearbeitet
- Bei Ankreuzfeldern wird der *Text wenn angekreuzt* bearbeitet

### Vorgeschlagener Dateiname beim *Speichern als...*

Bei der Funktion *Speichern als* im Editor wurde im Auswahldialog für den Dateinamen bisher der Name und Pfad der zuletzt geöffneten Formulardatei vorgeschlagen. Dieses Verhalten

konnte dazu führen, dass man ein anderes Formular versehentlich überschrieb. Daher wurde das Verhalten geändert.

Wenn das zu speichernde Formular bereits über einen Namen und ein Verzeichnis verfügt – dies ist bei allen Formularen der Fall, die mittels der Funktion *Öffnen* geöffnet wurden oder die zuvor bereits durch *Speichern als...* gespeichert wurden - wird immer der vorhandene Formularname und das vorhandene Verzeichnis verwendet.

Wenn das Formular noch keinen Namen (und somit auch noch kein Verzeichnis) hat, wird als Verzeichnis dasjenige verwendet, aus dem zuletzt ein Formular geöffnet wurde oder in das zuletzt ein Formular gespeichert wurde. Als Dateiname wird der Name *Neues Formular.fvl* vorgeschlagen.

#### **Fehler behoben beim Exportieren von RTF-Objekten nach PDF**

Wenn RTF-Objekte mit sehr großen Zeichentabellen in eine PDF-Datei exportiert wurden, sind Programmabstürze aufgetreten.

Die Programmabstürze wurden durch einen internen Pufferüberlauf verursacht.

Der Fehler wurde behoben.

#### **Fehler behoben beim Speichern von Formularen im XML-Format**

Das Abspeichern von Formularen im XML-Format war unter verschiedenen Umständen fehlerhaft. Die erzeugten XML-Dateien waren unbrauchbar und konnten nicht wieder eingelesen werden.

Der Fehler trat insbesondere auf, wenn das Formular eine oder mehrere der folgenden Komponenten enthielt:

- Schaltflächen oder Bildobjekte mit großen eingebetteten Bildern
- Java-Skripte – vor allem wenn diese einen gewissen Umfang haben

Der Fehler wurde behoben.

#### **Fehler behoben in den Javaskripten beim Ereignis *Klicken***

Die Javaskripte für das Ereignis *Objekt wurde angeklickt* wurden beim Export nach HTML bisher immer doppelt ausgegeben.

Der Fehler wurde behoben.

## Build 5.6.0.0

### Beispiel-Formularbibliothek

Die Bibliothek der Beispielformulare wurde vollständig überarbeitet und erweitert.

Sowohl im Ausfüller als auch im Editor wurde die neue Funktion *Neues Formular aus "e-forms" Formularbibliothek* implementiert. Im Editor ermöglicht sie die Auswahl eines Bibliothekformulars als Basis zur Erstellung einer neuen Vorlage. Im Ausfüller dient sie wie bisher dazu, ein Bibliotheksformular zum Ausfüllen auszuwählen.

Die Funktion öffnet einen Dialog, der die Bibliotheksformulare als Baum, also hierarchisch nach Themen gegliedert, anbietet. Zusätzlich ist aber auch eine Suchfunktion vorgesehen, um ein Formular auf Basis seines Namens schnell zu finden.

Eine Vorschau zeigt das augenblicklich ausgewählte Formular, um die Auswahl zu erleichtern.

### Eigene Formularbibliothek

Um neben der herstellerseitig installierten Bibliothek mit Beispielformularen auch eine eigene Formularbibliothek aufbauen zu können, wurde eine weitere neue Funktion: *Neues Formular aus eigener Formularbibliothek* implementiert.

Die eigene Bibliothek wird in einem eigenständigen Verzeichnis gepflegt, welches der Anwender bei der Installation vorgeben kann. Es können auch mehrere Arbeitsplätze auf die selbe Bibliothek zugreifen, wenn diese im Netzwerk liegt und der entsprechende Netzwerkpfad als Bibliotheksverzeichnis eingestellt ist. Das Verzeichnis kann im laufenden Betrieb geändert werden.

Die Hierarchie kann einfach durch Anlegen von Unterverzeichnissen abgebildet werden. Das Einstellen von Formularvorlagen in diese Bibliothek erfolgt durch Speichern von Formularen in eben dieses Verzeichnis.

Auch diese Funktion verfügt über eine Suchfunktion und über eine Vorschau.

### Neues Design

Der Formulareditor und der Ausfüller erscheinen in einem neuen Design. Die Symbolleisten wurden völlig neu gestaltet und dem Zeitgeist angepasst.

Viele Funktionen, sowohl neue, als auch bestehende, wurden der einfacheren Bedienung halber auf eigene Buttons gelegt.



### Modulbibliotheken

Viele Formulare werden aus Objektgruppen zusammengesetzt, die immer wieder benötigt werden. Ein einfaches Beispiel sind die Eingabefelder für die postalische Adresse, bestehend aus *Name*, *Vorname*, *Straße*, *Postleitzahl* und *Ort*.

Solche Objektgruppen – auch Module genannt – können ab sofort vom Anwender selbständig angelegt und verwaltet werden. Alle diese Module können beim Anlegen neuer Formulare bei Bedarf in ein Formular eingefügt werden.

Bei der Installation wird eine Beispiel-Modulbibliothek mit verschiedenen nützlichen Modulen automatisch angelegt.

Der Anwender kann aber auch eine eigene Modulbibliothek anlegen. Ähnlich der Formularbibliothek wird diese in einem eigenständigen Verzeichnis gepflegt, welches der Anwender bei der Installation vorgeben kann.

Der Editor enthält zwei neue Funktionen:

*“eforms“ Modulbibliothek öffnen...*  
*Eigene Modulbibliothek öffnen...*

Es wird jeweils ein Fenster geöffnet, welches alle Module in hierarchischer Form anbietet. Das Fenster ist nicht-modal, d.h. es bleibt offen, bis es entweder manuell geschlossen wird oder bis die Anwendung selbst beendet wird. Aus diesem Fenster können durch *Drag & Drop* Module in das augenblicklich bearbeitete Formular hinübergezogen werden.

Es können auch häufig benötigte einzelne Objekte mit verschiedenen Eigenschaften als eigenständige Module abgespeichert werden. Dadurch kann jeder Anwender sich seine Lieblingsobjekte zur schnellen Auswahl bereitstellen.

### Erweitertes Raster

Die Einstellungen für das Raster, in dem alle Objekte hinsichtlich Position und Größe einrasten, wurden erweitert.

Zum einen wurden die vorgegebenen Rastermaße erweitert. Insbesondere viele zöllige Maße wurden implementiert.

Außerdem wurde die Möglichkeit vorgesehen, das Raster für horizontale und vertikale Positionen separat vorzugeben.

### Erweitertes Gitternetz

Die Einstellungsmöglichkeiten für das Gitternetz, das zu Positionierung von Objekten im Editor eingeblendet werden kann, wurden erweitert.

Zum einen wurden die vorgegebenen Maße für die Abstände der Gitternetzlinien um einige Werte erweitert. Insbesondere viele zöllige Maße wurden implementiert.

Außerdem wurde die Möglichkeit vorgesehen, den Abstand der Gitternetzlinien für horizontale und vertikale Linien separat vorzugeben

Außerdem wurde eine Vorschau für die Gitternetzlinien implementiert.

### Eigene Symbolleiste für Datenzugriffsfunktionen

Die zahlreichen Funktionen für den Zugriff auf Datenbanken, die bisher als Buttons in der Symbolleiste des Ausfüllers und des Editors verfügbar waren, wurden in eine eigene Symbolleiste ausgelagert. Diese kann separat ein- oder ausgeschaltet werden. Wenn sie eingeschaltet ist, und das Fenster breit genug ist, werden beide Symbolleisten nebeneinander dargestellt. Andernfalls werden die Symbolleisten untereinander dargestellt.

Auf diese Weise wurde die Übersichtlichkeit der Buttons um Editor und im Ausfüller deutlich verbessert.

Im Menü *Anzeige* kann konfiguriert werden, welche Symbolleisten eingeschaltet sein sollen.

### Automatisches Zoomen beim Öffnen

In den Anzeigeeigenschaften eines Formulars kann ab sofort festgelegt werden, ob dieses Formular beim Öffnen oder bei jeder Fenstergrößenveränderung gezoomt werden soll, so dass es sich an die Fenstergröße anpasst.

Folgende Varianten können eingestellt werden:

- nicht Zoomen
- beim Öffnen des Formulars einmalig auf volle Fensterbreite zoomen
- beim Öffnen des Formulars einmalig auf volle Fensterhöhe zoomen
- bei jeder Veränderung der Fenstergröße auf volle Fensterbreite zoomen
- bei jeder Veränderung der Fenstergröße auf volle Fensterhöhe zoomen

Die Einstellungen wirken sowohl im Ausfüller als auch im Editor.

### Buttons einzeln schaltbar beim Datenzugriffsobjekt

In den Anzeigeeigenschaften des Datenzugriffsobjekts wurde die Möglichkeit geschaffen, jeden der Bedienknöpfe einzeln sichtbar oder unsichtbar zu schalten.

Die Sichtbarkeit ist unabhängig von den Funktionen, die für ein Datenzugriffsobjekt erlaubt sind. So kann z.B. die Funktion *Alle selektieren* aktiv geschaltet werden – ja sogar automatisch ausgeführt werden – ohne dass der entsprechende Button sichtbar gemacht werden muss.

## Version 5.4

### Build 5.4.2.1

#### Fehlerbehebungen

Wenn mehrzeilige Objekte in der Sammelbearbeitung von Objekten exportiert und anschließend wieder importiert wurden, trat ein Fehler auf. Dieser wurde behoben.

### Build 5.4.2.0

#### Erweiterung der Anzeigeeinstellungen

Die Anzeigeeinstellungen wurden um verschiedene Punkte erweitert, um die Formularsoftware auch für die Eingabe von Masken- und Datenbankanwendungen flexibler zu machen.

Insbesondere wurde die Möglichkeit implementiert, beim Öffnen eines Formulars die Größe des Hauptfensters automatisch anzupassen und auf die Blattgröße einzustellen.

Alle neu implementierten Bedienelemente befinden sich im Menü *Formular* in der Funktion *Anzeigeeinstellungen*.

### Build 5.4.0.1

#### Export nach PDF wurde erweitert

Beim Export eines Formulars nach PDF kann jetzt optional eingestellt werden, dass Eingabeobjekte nicht als Controls exportiert werden, sondern gezeichnet werden.

### Build 5.4.0.0

#### Erweiterung der Sammelbearbeitung

Die Sammelbearbeitung von Objekten wurde um verschiedene Features erweitert:

- Die Sammelbearbeitung kann ab sofort auf **Alle Objekte** oder auf **Alle markierten Objekte** angewendet werden

- Es wurde ein Filter implementiert, der es erlaubt, jeden einzelnen Objekttyp ein- oder auszublenden. Der zuletzt eingestellte Filter wird gespeichert und beim nächsten Aufruf wieder angewendet.
- Es wurde eine Export-Funktion implementiert, die es erlaubt, die Eigenschaften aller Objekte der Sammelbearbeitung in eine CSV-Datei zu exportieren.
- Es wurde eine Import-Funktion implementiert, die es erlaubt, alle zuvor exportierten Objekteigenschaften wieder aus der CVS-Datei zu importieren.
- Dadurch wird die externe Bearbeitung von Objekteigenschaften ermöglicht

#### Erweiterungen der Statusanzeige

- Bei Eingabeobjekten wird ab sofort in der Statuszeile hinter dem Objekttyp der Index des Objektes in der Eingabereihenfolge eingeblendet.
- Verschiedene Statusfelder - insbesondere das Feld für die Anzeige des Objektnamens - wurden verbreitert.
- Das Statusfeld für die Anzeige des Objektnamens wurde nach vorne geschoben.

## Version 5.3

### Build 5.3.0.3

#### Fehlerbehebung im Page-API

Manche Eigenschaften einer Seite konnten via Page-API nicht gesetzt werden. Der Fehler wurde behoben.

### Build 5.3.0.2

#### Neue Notification-Message

Die Formularsoftware sendet für alle Programme, die eine API-Anbindung haben, ab sofort eine Notification-Message vom Typ:

FORMMSG\_OBJECT\_CONTENT\_CHANGED

sobald sich der Inhalt eines beliebigen Eingabeobjekts ändert. Der Handle des Eingabeobjekts wird der Message als Parameter *IParam* übergeben.

Dadurch hat jede Anwendung, in die ein Formular eingebunden wird, die volle Kontrolle über das Formular.

Näheres ist in der API-Dokumentation *formdev.pdf* beschrieben.

### Build 5.3.0.1

#### Fehlerbehebungen

Es wurde die Bibliothek *freeimage.dll* bei der Installation hinzugefügt.

Es wurde ein Fehler in der Schriftgröße beim Export von Checkboxes nach PDF behoben.

## Build 5.3.0.0

### Neue Schnittstellen zum Setzen und Lesen der Druckparameter

Es wurden zwei neue API-Schnittstellen implementiert, die das Setzen und Auslesen der Druckparameter ermöglichen. Diese lauten:

*Form\_SetXmlPrintProperty*

*Form\_SetXmlPrintProperty*

Die Schnittstellen sind in der API-Dokumentation *formdev.pdf* beschrieben.

### Windows 7 Kompatibilität

[e]forms & more ist ab Version 5.3. vollständig Windows 7<sup>®</sup> Microsoft Corporation kompatibel.

### Fehlerbehebungen

Es wurde die Bibliothek *freeimage.dll* bei der Installation hinzugefügt.

Es wurde ein Fehler in der Schriftgröße beim Export von Checkboxes nach PDF behoben.

## Version 5.2

### Build 5.2.0.10

#### Schreibschutz für Schaltflächen

Es wurde für Schaltflächen die Möglichkeit implementiert, die Option *nicht veränderbar* zu setzen. Schaltflächen mit diesem Attribut, werden beim Export nach PDF so exportiert, dass sie in der PDF-Datei nicht als Schaltflächen, sondern als Bilder erscheinen.

### Build 5.2.0.9

#### Lizenzierung für OEM-Kunden erweitert

Das Lizenzmodell wurde für OEM-Kunden weiterentwickelt.

### Build 5.2.0.8

#### Gestrichelte Linien im PDF-Export

Ab sofort werden gestrichelte Linien auch nach PDF exportiert.

### Build 5.2.0.7

#### Kontext-Menüs wurden erweitert

Das Kontext-Menü für einzelne markierte Objekte wurde um folgende Funktionen erweitert:

- In den Vordergrund
- In den Hintergrund

Das Kontext-Menü für mehrere markierte Objekte wurde um folgende Funktionen erweitert:

- Gruppieren
- Gruppe auflösen

- In den Vordergrund
- In den Hintergrund



Links ausrichten  
Rechts ausrichten  
Oben ausrichten  
Unten ausrichten

Nebeneinander anordnen  
Untereinander anordnen

### PDF-Barriere-Tag für mehrere Objekte gleichzeitig ändern

Es wurde implementiert, dass der Barriere-Tag für den Export nach PDF für mehrere markierte Objekte gleichzeitig verändert werden kann.

### PDF-Lesezeichen

Es werden beim PDF-Export ab sofort automatisch PDF-Lesezeichen erzeugt. Diese werden automatisch aus den Barriere-Tags *Überschrift 1* bis *Überschrift n* abgeleitet.

## Build 5.2.0.6

### Verbesserung der Installation für den PDF-Import

Das Importmodul wird ab sofort in bestehende Acrobat-Installationen integriert.

## Build 5.2.0.5

### Erweiterungen beim PDF-Import

Die vertikale Ausrichtung importierter Eingabefelder wird jetzt standardmäßig auf *mittig* gesetzt.

Wenn mehrere Fehler den gleichen Namen haben, werden diese ab sofort alle importiert und nicht nur das erste.

Die PDF-Eigenschaft *QuickInfo* (*kurzer Text*) wird als *Status-Text* importiert.

## Build 5.2.0.4

### Fehlerbehebungen

Beim Speichern von Formularen als XML-Datei wurden RTF-Objekte fehlerhaft exportiert, wenn diese eine bestimmte Größe überschritten. Dies führte zu Fehlern beim Lesen der exportierten Objekte. Der Fehler wurde behoben.

## Build 5.2.0.3

### Neues Bibliotheksformular für Millimeterpapier

Es wurde ein neues Bibliotheksformular zum Drucken eines Din-A4-Blattes mit Millimeterpapier eingestellt. Das Formular befindet sich in der Rubrik *Schule* im Menüpunkt *Öffnen aus Vorlagenbibliothek*.

### Neue API-Funktion zum Setzen von PDF-Export-Parametern

Der automatisierte Export von Formularen nach PDF mittels API-Routinen kann ab sofort durch das Setzen von *Flags* besser konfiguriert werden. Es wurde eine neue Schnittstelle:

*Form\_AddPdfExportFlag (HWND hwndForm, longflag)*

implementiert, die es erlaubt eines oder mehrere der folgenden Flags zu setzen:

FORM_EXPFLAG_DRAWFIELDS	Eingabeobjekte werden in Druckform exportiert und nicht als Eingabeobjekte
FORM_EXPFLAG_ONLYPRINTABLE	Es werden nur Objekte exportiert, die die Eigenschaft <i>drucken</i> eingeschaltet haben

### Neue API-Funktion zum Drücken eines Knopfes

Es wurde eine neue API-Funktion implementiert, die es erlaubt, einen Knopf zu drücken, während das Formular im Ausfüllmodus ist:

*Form\_PerformButton (HWND hwndForm, const char \*ButtonName)*

Beim Aufruf der Funktion passiert dasselbe, wie wenn der Knopf durch Interaktion des Anwenders gedrückt wurde (z.B. durch Anklicken mit der Maus).

### Neue API-Funktionen zum Löschen von Objekten

Es wurde eine neue API-Funktion implementiert, die es erlaubt, alle Objekte eines bestimmten Typ zu löschen, die sich innerhalb eines vorgegebenen Rechtecks befinden:

*Form\_DeleteObjectsInRect (HWND hwndForm, long objectTypes, long left, long top, long right, long bottom, long flags)*

Ferner wurde eine neue API-Funktion implementiert, die es erlaubt, Objekte anhand ihres Namens auf allen Seiten, die einen bestimmten Namen haben, zu löschen:

*Form\_DeleteObjectsByName (HWND hwndForm, char \*PageName, char \*ObjectName, long Flags)*

### Neue Eigenschaft zum Unterdrücken von Fehler-Dialogboxen

Es wurde eine neue Eigenschaft implementiert, mit der das Formular angewiesen wird, keinerlei Dialogboxen im Fehlerfall zu öffnen. Die Eigenschaft kann mittels:

*Form\_SetXmlProperty (hwndForm, FORM\_PROP\_DISABLE\_WARNING\_BOXES, "1")*

gesetzt werden. Dadurch können Stapelprozesse optimiert werden.

### Neue API-Funktion zum Setzen von Seiteneigenschaften

Es wurde eine neue API-Funktion zum Setzen von Eigenschaften einer einzelnen Seite implementiert:

*Form\_SetPageXmlProperty (HWND hwndForm, const char \*pageName, const char \*attrName, const char \*data)*

Mittels dieser Funktion können einzelne Seiteneigenschaften als XML-Attribute gesetzt werden.

---

## Build 5.2.0.2

### Neues Bibliotheksformular für Ordnerrücken

Das Bibliotheksformular für die Gestaltung von Ordnerrücken wurde vollständig überarbeitet und gestattet jetzt, dynamisch mehrere Etiketten anzulegen. Eine gute Demonstration der neuen Makrobefehle, die in Version 5.2.0.1 eingeführt wurden.

### Fehlerbehebungen

Beim Export nach PDF wurde ein Exportfehler behoben, der immer dann auftrat, wenn man ein Textobjekt mit Teilungsstrichen exportiert hat, welches nicht leer war.

## Build 5.2.0.1

### Neue Makrofunktionen für dynamische Formulare

Für die Umgestaltung von Formularen zur Laufzeit wurden weitere Makrobefehle implementiert:

- AppendObjectSetExt*
- DeleteObjectsByName*
- GetObjectsEnvelope*
- GetPageHeight*
- GetPageWidth*
- PostFormMacro*
- PostObjectMacro*

Die Befehle erweitern die Möglichkeiten, dynamisch Objekte zur Laufzeit hinzuzufügen oder aus dem Formular zu entfernen.

Alle Funktionen sind in der Dokumentation *Formular Makrobefehle* erläutert.

### Neue allgemeine Makrofunktionen

Die Bibliothek von allgemeinen (also nicht formularspezifischen) Makrofunktionen wurde erweitert. Es wurden folgende Funktionen hinzugefügt:

- StripLeft* (entfernt Zeichen am Anfang eines Textes)
- StripRight* (entfernt Zeichen am Ende eines Textes)
- StripMid* (entfernt Zeichen in der Mitte eines Textes)
- FindString* (sucht eine Zeichenfolge in einer anderen Zeichenfolge)

*HtmlToRgb* (konvertiert einen HTML-Farbcode in einzelne Farbwerte)

*RgbToHtml* (konvertiert einzelne Farbwerte zu einem HTML-Farbcode)

Diese Funktionen können in allen Anwendungen verwendet werden, die die Makrosprache *Wincula* verwenden. Sie sind in der Dokumentation *Allgemeine Makrobefehle* beschrieben.

## Build 5.2.0.0

### Neue Kommandozeilen-Schalter im Ausfüller

Für den Start des Ausfüllers über Kommandozeile wurden weitere Schalter implementiert:

<code>/WINDOWWIDTH=xxx</code>	Definieren der Fensterbreite in Pixeln
<code>/WINDOWHEIGHT=xxx</code>	Definieren der Fensterhöhe in Pixeln
<code>/RESIZE</code>	Fenster immer in Normalgröße öffnen

Die Schalter zur Vorgabe der Fenstergröße müssen immer beide gesetzt werden, damit die Werte übernommen werden.

Wenn die Fenstergröße als Schalter gesetzt wird, wird automatisch auch der Schalter `/RESIZE` impliziert.

### Auslösen von Schaltflächen durch eine Shortcut-Taste

Schaltflächen können ab sofort mit einem sogenannten *Shortcut*-Buchstaben assoziiert werden. Wenn dann die Tastenkombination STRG+UMSCHALT+Buchstabe gedrückt wird, wird die erste Schaltfläche, die mit dem Buchstaben als *Shortcut* assoziiert ist, ausgelöst.

## Version 5.1

### Build 5.1.0.13

#### Inhalt eines Feldes beim Betreten nicht markieren

Bisher wurde beim Betreten eines Eingabefeldes der Inhalt markiert. Dies ist das übliche Verhalten unter Windows-Anwendungen.

Ab sofort kann dieses Verhalten für jedes Feld dahingehend konfiguriert werden, dass der Feldinhalt entweder beim Betreten wie bisher markiert wird oder dass der Inhalt unmarkiert bleibt und der Cursor ans Ende des Feldinhaltes gesetzt wird.

#### Fehlerbehebungen

Wenn sich der Cursor in einem mehrzeiligen Eingabefeld befand, wurden die Impulse des Rollrades der Maus an das Eingabefeld anstelle an das Formular fokussiert. Dadurch konnte das Formular mit der Maus in diesem Fall nicht mehr gerollt werden. Das Verhalten wurde korrigiert. Die Rollimpulse werden jetzt an das Formular weitergegeben.

### Build 5.1.0.12

#### Fehlerbehebungen

Beim Export von RTF-Texten nach PDF wurde beim Einzug des rechten Randes ein Fehler behoben.

Beim Export von Ankreuzfeldern mit Label nach PDF wurden Texte zwar korrekt mit Tags versehen, aber es gab keine Alternativtexte mit Leerzeichen. Der Fehler wurde behoben.

### Build 5.1.0.11

#### Eingabepflicht für Auswahlfelder

Für Auswahlfelder wurde die Option *Eingabepflicht* implementiert. Diese ist nur beim Export nach PDF wirksam und wird in die gleichbedeutende PDF-Eigenschaft transformiert.

## Fehlerbehebungen

### Doppelte Bilder

In bestimmten Fällen wurden Bilder beim Export nach PDF fehlerhaft exportiert. Bilder, die im Original-Formular verschieden waren, erschienen in der PDF-Datei als identische Kopien des ersten Bildes. Der Fehler wurde behoben.

### Export von PDF-Dateien nach PDF

Die Tabulator-Einrückung am Anfang von Zeilen in RTF-Dateien wurde beim Export nach PDF manchmal fehlerhaft exportiert. Der Algorithmus wurde verbessert.

## Build 5.1.0.10

### Anzeige der Schriftart für Ankreuzfeld-Labels

Im Eigenschaften-Dialog wird nun die Schriftart angezeigt, mit der das Label ausgegeben wird.

## Fehlerbehebungen

### Doppelter Menüeintrag

Die neue Funktion *Sammelbearbeitung* stand versehentlich zweimal im Menü. Der Fehler wurde behoben.

### Refresh nach Sammelbearbeitung

Nach Änderungen, die in der Sammelbearbeitung durchgeführt wurden, erfolgte keine Neuanzeige. Dadurch wurden die Änderungen nicht sichtbar. Der Fehler wurde behoben.

## Build 5.1.0.9

### PDF-Export Barriere-Tag für gedrehte Texte

Für gedrehte Texte wurde implementiert, dass die Barriere-Tags beim PDF-Export in die PDF-Datei geschrieben werden.

### Barriere-Tag für Label von Checkboxes

Für Ankreuzfelder mit einem Label wurde implementiert, dass der Barriere-Tag des Labels immer vor dem Barriere-Tag des Ankreuzfeldes ausgegeben wird – unabhängig davon ob das Label links oder rechts vom Ankreuzfeld steht.

### Quadratform von Ankreuzfeldern

Für Ankreuzfelder wurde die Eigenschaft *quadratisch* implementiert, da zunehmend Ankreuzfelder mit einem Label versehen werden und diese dann natürlich nicht mehr quadratisch sind. Dies führte immer wieder zu Verwirrungen. Bestehende Formulare mit quadratischen Ankreuzfeldern ohne Label bleiben selbstverständlich unverändert.

## Build 5.1.0.8

### Neue Funktionen zur Sammelbearbeitung von Objekten

Es wurde eine Funktion implementiert, die es ermöglicht, bestimmte Eigenschaften aller Eingabeobjekte oder aller markierten Objekte in einer gemeinsamen Tabelle zu bearbeiten. Die Funktionen heißen:

*Eingabeobjekte ändern (Sammelbearbeitung)*

*Markierte Objekte ändern (Sammelbearbeitung)*

Es wird jeweils eine Tabelle eröffnet, in der die Eigenschaften der entsprechenden Objekte zeilenweise angeordnet sind und bearbeitet werden können.

## Build 5.1.0.7

### Custom-String für alle Objekte

Es wurde die Möglichkeit implementiert, eine anwenderspezifische Zeichenfolge für jedes Objekt zu hinterlegen. Dies erleichtert die Integration der Formularsoftware in fremde Umgebungen. Es können individuelle Zustände und Konfigurationen gespeichert werden.

Die anwenderspezifische Zeichenfolge wird von der Software mit in die Formularvorlagen geschrieben und aus diesen auch wieder eingelesen. Über eine Schnittstelle kann die Zeichenfolge während der Laufzeit ausgelesen und eingespeichert werden.

## Build 5.1.0.6

### Fehlerbehebungen

#### Export Globale JavaSkripte nach PDF

Beim Export globaler JavaSkripte nach PDF wurde ein Fehler im Acrobat umgangen.



## Build 5.1.0.5

### Fehlerbehebungen

#### Tag-Reihenfolge in Gruppen

Es wurde ein Fehler in der Reihenfolge der Ausgabe von Barriere-Tags in Gruppen beim Export nach PDF behoben.

## Build 5.1.0.4

### Fehlerbehebungen

#### RTF-Darstellung und PDF-Export

Es wurden Fehler bei der RTF-Darstellung und beim PDF-Export behoben.

## Build 5.1.0.3

### Neue Schnittstelle für Refresh

Es wurde eine neue API-Funktion implementiert:

*Form\_Invalidate (HWND hwndForm)*

Der Aufruf dieser Funktion führt zu einem umgehenden Refresh des Formularfensters.

### Fehlerbehebungen

#### Im Makrobefehl *SendToHttpServer* wurde ein Fehler behoben

Im Fehlerfall wurde der Puffer der Fehlermeldung nicht korrekt vorbesetzt. Der Fehler wurde korrigiert. Die Fehlerbehandlung wurde verbessert.

## Build 5.1.0.2

### Option für die Reihenfolge der Eingabefelder beim PDF-Export

Beim PDF-Export wurde eine Option implementiert, die es ermöglicht, zu wählen, ob alle Objekte in geometrischer Anordnung (oben nach unten und links nach rechts) exportiert werden, oder ob die Eingabereihenfolge verwendet werden soll.

## Build 5.1.0.1

### Einfügen von Bildern aus der Zwischenablage

Die Funktion *Einfügen (aus der Zwischenablage)* im Editor wurde erweitert. Bisher konnten nur native Formularobjekte eingefügt werden, die zuvor markiert und in die Zwischenablage kopiert wurden. Ab sofort können auch Bilder, die aus anderen Quellen in die Zwischenablage kopiert wurden, eingefügt werden. IN diesem fall wird automatisch ein Bild-Objekt erzeugt und das Bild aus der Zwischenablage in das Bildobjekt eingebettet.

### Fehlerbehebungen

#### API-Funktion *SForm\_Open\_Expanded* wurde implementiert

Die mit Version 5.1 neu überarbeitete API-Dokumentation beschreibt die API-Funktionen *Form\_Open\_Expanded* und *SForm\_Open\_Expanded*. Die letztere der beiden Funktionen wurde aber nicht implementiert. Dies wurde nachgeholt.

## Build 5.1.0.0

### Vollständig überarbeitete API-Dokumentation

Die API-Dokumentation *formdev.pdf* wurde vollständig erneuert. Es wurden alle neuen Routinen dokumentiert. Die Beschreibungen für obsoletere Routinen wurde entfernt.

### Labels für Ankreuzfelder

Die Ankreuzfelder wurden dahingehend erweitert, dass sie automatisch mit einem Label versehen werden können. Dadurch braucht nicht mehr jedes Ankreuzfeld mit einem separaten Textobjekt beschriftet zu werden.

Die Box zum ankreuzen kann wahlweise links vor dem Labeltext oder rechts dahinter stehen. Links ist die Voreinstellung. Wenn die Box hinter dem Text steht, so wird sie rechtsbündig im Objekt ausgerichtet und der Labeltext ist rechtsbündig gegen die Box gesetzt.

Die Erweiterung ist aufwärtskompatibel zu bisherigen Versionen. Ankreuzfelder ohne Label verhalten sich exakt wie bisher.

Die Voreinstellung *quadratisch* ist allerdings nicht mehr gesetzt, da ein Ankreuzfeld mit einem Label keine quadratische Form hat. Die Form umfasst immer das ganze Objekt – also in diesem Fall Box und Label. Die Eigenschaft *quadratisch* kann allerdings wie bisher eingeschaltet werden. Sie sollte nicht in Verbindung mit einem Label benutzt werden.

### Automatische Objektgrößenanpassung konfigurierbar

Die Funktion, dass der Umriss von Objekten beim Edieren automatisch an die Textlänge des Objekts angepasst wird, kann ab sofort im Menü *Extras* unter *Einstellungen* ein- oder ausgeschaltet werden.

Erweiterung des Dialogs für die Eigenschaften mehrerer Objekte

Die Funktion *Eigenschaften* im Menü *Objekte* die einen Dialog eröffnet, der es erlaubt, die Eigenschaften mehrerer Objekte gleichzeitig zu verändern, wurde erweitert. Ab sofort ist es auch möglich folgende Eigenschaften von Ankreuzfeldern zu verändern:

- Name der Gruppe von gegenseitig auslösenden Ankreuzfeldern
- Eigenschaft *Ankreuzfelder der gleichen Gruppe auslösen*
- Eigenschaft *Mindestens ein Feld muss angekreuzt sein*

### Vertikales Rollen in mehrzeiligen Eingabefeldern

Für mehrzeilige Eingabefelder wurde eine Option implementiert, die das vertikale Rollen im Text ermöglicht, sobald dieser länger wird als es das Eingabefeld darstellen kann.

### Neue Schnittstellen zum Setzen und Lesen von Eigenschaften

Es wurden folgende neue Schnittstellen zum Setzen und Lesen von Eigenschaften des Formulars implementiert:

```
Form_GetXmlProperty  
Form_SetXmlProperty  
Form_GetXmlPropertyByAttrName  
Form_SetXmlPropertyByAttrName
```

Es wurden folgende neue Schnittstellen zum Setzen und Lesen von Eigenschaften von einzelnen Seiten implementiert:

```
FormPage_GetXmlProperty  
FormPage_SetXmlProperty  
FormPage_GetXmlPropertyByAttrName  
FormPage_SetXmlPropertyByAttrName
```

Es wurden folgende neue Schnittstellen zum Setzen und Lesen von Eigenschaften einzelner Objekte implementiert:

```
FormObj_GetXmlProperty  
FormObj_SetXmlProperty  
FormObj_GetXmlPropertyByAttrName  
FormObj_SetXmlPropertyByAttrName
```

### Fehlerbehebungen

Beim automatischen Rollen (mit Mauszeiger bei gedrückter Maustaste gegen den Fenster-  
rand stoßen), wurde der Fehler beseitigt, dass horizontal zwar nach rechts, nicht aber wie-  
der nach links gerollt werden konnte.

Ferner wurde beim automatischen Rollen die vertikale Rolldistanz verkleinert, so dass jetzt  
*smooth* gerollt werden kann. Vorher sprang das Formular immer gleich um eine halbe Seite  
weiter.

## Version 5.0

### Build 5.0.1.8

#### Verbesserungen beim PDF-Export

Es wurden weitere Verbesserungen beim PDF-Export implementiert.

### Build 5.0.1.7

#### Barrierefreiheit-Tags für RTF-Objekte

Die Tags für die Barrierefreiheit im PDF-Export wurden zusätzlich auch in den RTF-Objekten implementiert.

#### Verbesserungen im PDF-Export

Es wurden weitere Fehlerbehebungen im PDF-Export durchgeführt.

### Build 5.0.1.6

#### Barrierefreiheit-Tags für PDF-Export

Es wurde für Textobjekte eine Erweiterung implementiert, die es erlaubt, den Tag auszuwählen, der für die Barrierefreiheit nach PDF exportiert wird.

#### Gruppen bei der Erzeugung der Barrierefreiheit-Tags

Bei der Festlegung der Reihenfolge für die Erzeugung für Barrierefreiheit-Tags im PDF-Export wurde implementiert, dass Gruppen vorrang vor der geometrischen Anordnung haben.

#### Verbesserungen im PDF-Export

Es wurden weitere Fehlerbehebungen im PDF-Export durchgeführt.

## Build 5.0.1.5

### Fehlerbehebungen

Bei der Bearbeitung von Makros entstanden gelegentlich Leerzeilen am Anfang des Makrocodes. Der Fehler wurde behoben.

## Build 5.0.1.4

### Implementation der Tags für Barrierefreiheit beim PDF-Export

Die Tags für die Unterstützung der Barrierefreiheit in PDF-Dateien werden ab sofort beim PDF-Export angelegt.

### Verbesserungen im PDF-Export

Es wurden weitere Fehlerbehebungen im PDF-Export durchgeführt.

## Build 5.0.1.3

### Verbesserungen im PDF-Export

Es wurden weitere Fehlerbehebungen im PDF-Export durchgeführt.

## Build 5.0.1.2

### Verbesserungen im PDF-Export

Farbige 3D-Rechtecke wurden nicht korrekt exportiert. Der Fehler wurde behoben.  
Farbige Seitenhintergründe wurden nicht nach PDF exportiert. Der Fehler wurde behoben.  
In Javascripten wurden die Aufrufe der Events *OnValidate* und *OnBlurnicht* korrekt behandelt. Der Fehler wurde ebenfalls behoben.

## Build 5.0.1.1

### Automatische Limitierung der Länge von typbehafteten Feldern

Typbehaftete Eingabeobjekte erhalten ab sofort eine automatische Maximallänge, die dem internen Genauigkeitsbereich des Datentyps entspricht. Wenn eine explizite Maximallänge definiert ist, gilt die explizite Länge. Die typspezifischen Minimallängen sind:

Ganzzahl:	9 Zeichen
Gleitkommazahl:	15 Zeichen
Datum:	20 Zeichen
Uhrzeit:	20 Zeichen

### Fehlerbehebungen im PDF-Export

Folgende Fehler beim PDF-Export wurden behoben:

- Die Linien bei Texten mit farbiger Umrahmung wurden immer schwarz
- Durchgehende Linien wurden manchmal gestrichelt dargestellt

## Build 5.0.1.0

### Freigabe PDF Export ohne Acrobat

Der Export von Formularen nach PDF erfordert ab Version 5.0.0.0 nicht mehr das Produkt *Adobe Acrobat* oder ein adäquates Produkt, welches PDF-Dateien in Form eines Druckertreiber erstellen kann. Vielmehr können PDF-Dateien nunmehr direkt aus *eforms& more* erzeugt werden.

Diese Funktion wurde nun endgültig für alle Anwender freigegeben.

### Eigenschaft nicht veränderbar für mehrere Objekte setzen

Die Eigenschaft *nicht veränderbar* die bisher nur für einzelne Eingabeobjekte definiert werden konnte, kann ab sofort für alle markierten Objekte gleichzeitig ein- oder ausgeschaltet werden.

### Fehlerbehebungen

Es wurden verschiedene Fehler im neu implementierten PDF-Export, der den Adobe Acrobat nicht mehr benötigt, beseitigt.

## Build 5.0.0.1 bis 5.0.0.6

### Fehlerbehebungen

Es wurden verschiedene Fehler im neu implementierten PDF-Export, der den Adobe Acrobat nicht mehr benötigt, beseitigt.

Bei RTF-Objekten wurden Schriftgrößen bei Zoom-Faktoren ungleich 100% teilweise nicht korrekt dargestellt. Der Fehler wurde behoben.

Es wurde ein Anzeigefehler im RTF-Editor behoben.

## **Build 5.0.0.0**

### **HTML und PDF Export jetzt im Standardlieferumfang**

Der Export von Formularen nach HTML und PDF ist ab sofort im Standardlieferumfang des Editors enthalten und braucht nicht mehr separat lizenziert zu werden.

### **PDF Export erfordert nicht mehr den Adobe Acrobat**

Der Export von Formularen nach PDF erfordert ab sofort nicht mehr das Produkt *Adobe Acrobat* oder ein adäquates Produkt, welches PDF-Dateien in Form eines Druckertreiber erstellen kann. Vielmehr können PDF-Dateien nunmehr direkt aus *eforms& more* erzeugt werden.



## Version 4.0

### Build 4.3.0.4

#### Tooltips für Buttons in Datenzugriffsobjekten

Für die Buttons in Datenzugriffsobjekten wurden Tooltips implementiert.

### Build 4.3.0.3

#### Pufferverlängerung für Datenbankzugriff

Beim Datenbankzugriff waren bisher alle Puffer auf 1000 Zeichen begrenzt. Dieser Wert wurde für Datenbankfelder, die Langtexte beinhalten können, auf 10.000 Zeichen verlängert.

### Build 4.3.0.2

#### Unterdrückung des HTML-Exports für nicht angezeigte Objekte

Ausgabeobjekte, die den Status *Anzeigen* nicht gesetzt haben, werden ab sofort auch nicht mehr nach HTML exportiert. Eingabeobjekte (Eingabefelder, Auswahlfelder, Ankreuzfelder und Buttons) hingegen werden auf jeden Fall exportiert und erhalten in dem Fall, dass der Status *Anzeigen* nicht gesetzt ist das Attribut *visibility:hidden*.

### Build 4.3.0.1

#### Makrobefehl zum modifizieren des Tooltip-Textes von Buttons

Es wurde ein neuer Makrobefehl implementiert, der es ermöglicht, den Tooltip-Text eines Buttons in der Buttonleiste des Ausfüllers zu ändern.

Die Befehle zur Modifikation der Buttons beginnen alle mit „*Abb...*“. Dieser Term steht für *Application Button Bar*.

Der neu implementierte Befehle lautet: *AbbSetButtonPopupText*. Eine detaillierte Beschreibung findet man in der Dokumentation *Interne Makrobefehle*.

## Build 4.3.0.0

### Export einer Seite als Grafikdatei

Es wurde ein neue Funktion implementiert, die die aktuelle Seite als Grafikdatei (Bitmap, JPG, etc.) in eine Datei exportiert. Die Farbtiefe kann frei definiert werden. Die Auflösung basiert auf dem aktuell eingestellten Standarddrucker.

### Neues Makro ListBoxDialog

Es wurde ein neuer Makrobefehl *ListBoxDialog* implementiert. Dieser löst die bisherigen Makrobefehle *ListDialog*, *ListDialogMultiSelect*, *ListDialogAnsi* und *ListDialogAnsiMultiSelect* ab. Dabei wurden unter anderem verschiedene Fehlfunktionen behoben.

## Build 4.2.1.0

### Dateieingabefelder für PDF-Export

Für den Export nach PDF wurde ein Schalter implementiert, der als Eigenschaft in die exportierte PDF-Datei übernommen wird und der ein Eingabefeld als *Dateinamen-Eingabefeld* kennzeichnet. Der Schalter hat für den nativen Betrieb keine Bedeutung.

### Import von PDF aus Acrobat 7

Das Import-Modul wurde für den Import aus Acrobat 7 umgestellt.

## Build 4.2.0.0

### Definition der Bedeutung von Eingabefeldern

Für die Eingabeobjekte *Eingabefeld* und *Auswahlfeld* wurde eine Erweiterung implementiert, die es ermöglicht, die semantische Bedeutung des Feldes zu definieren. In der ersten Implementationsstufe kann eine der folgenden Bedeutungen ausgewählt werden:

*Nicht spezifiziert*

*Name*

*Vorname*

*Email-Adresse*

*Telefonnummer privat*

*Telefonnummer beruflich*

*Telefonnummer mobil*

Die Bedeutung der Felder findet zur Zeit nur beim Export einer Visitenkarte Verwendung. Bestehende Formulare und deren Felder erhalten automatisch die Bedeutung *nicht spezifiziert*.

#### Export einer Visitenkarte

Sowohl im Ausfüller als auch im Editor wurde die Funktion *Visitenkarte exportieren* implementiert. Es wird ein Dateiname angefragt, und dann aus dem aktuellen Formular eine Visitenkartendatei erzeugt (VCF-Datei). Dabei wird erwartet, dass die Felder (Eingabefelder oder Auswahlfelder), die in die Visitenkarte exportiert werden sollen, mit der entsprechenden semantischen Bedeutung markiert worden sind.

Zusätzlich wurde auch der zugehörige Makrobefehl *CreateVisitCardFile* implementiert, der diese Leistung als Makro abbildet.

### Build 4.1.0.9

#### Fehlerbehebungen

Es wurden diverse Fehler beim PDF-Export behoben.

### Build 4.1.0.8

#### Fehlerbehebungen

Wenn der Anzeigestatus eines Datenzugriffsobjekts mittels des Makrobefehls *SetDisplayState* verändert wurde, blieb dies ohne Wirkung. Der Fehler wurde behoben.

Es wurden verschiedene Fehler beim Senden an einen HTTP-Server behoben. Ferner wurden Verbesserungen beim PDF-Export implementiert.

### Build 4.1.0.7

#### Senden an HTTP-Server

Es wurde die Möglichkeit implementiert, aus dem Ausfüller heraus, ein Formular direkt an einen HTTP-Server zu versenden.

### Verbesserungen im PDF-Import

- Verbesserter Import von Rechtecken mit runden Ecken
- Import von Unicode-Zeichen nach ANSI wurde verbessert
- Das Import-Protokoll wurde verbessert
- Es wurden Fehler beim Erkennen der Hintergrundfarbe behoben
- Verbesserungen bei der Schriftartenerkennung der PDF-Fonts

### Fehlerbehebungen

Es wurde ein Fehler behoben, der bei Auswahlfeldern (nur Liste) zum Absturz führte, falls keine Vorbesetzung angegeben war.

## Build 4.1.0.5

### Fehlerbehebungen

Es wurden zahlreiche Fehler beim PDF-Export behoben.

Beim Speicher von Formularen in XML-Dateien traten Fehler auf, wenn Objektnamen das Zeichen # enthalten. Der Fehler wurde behoben.

## Build 4.1.0.4

### Runde Ankreuzfelder

Die Darstellung von Ankreuzfeldern kann ab sofort nicht nur eckig, sondern auch rund erfolgen.

## Build 4.1.0.3

### Neuer Makrobefehl zum Anzeigen eines ListView-Dialogs

Es wurde eine Neue Makrofunktion *ListViewDialog* implementiert. Die Funktion ist in der Dokumentation *intfunc.pdf* beschrieben.

### Fehlerbehebungen

Beim Anlegen neuer Objekte sollen diese am oberen Bildschirmrand erscheinen. In Wirklichkeit wurden die Objekte an einer beliebigen, aber fehlerhaften Position erstellt. Der Fehler wurde behoben.

Beim Ändern von mehrzeiligen Textobjekten wurden diese immer um einen halben Zentimeter breiter, wenn man die Eigenschaft *mehrzeilig* eingeschaltete. Der Fehler wurde behoben. Mehrzeilige Objekte werden in ihrer Größe nicht mehr verändert.

## Build 4.1.0.2

### Fehlerbehebungen

Beim Speichern in XML-Dateien wurden die Namen benutzerdefinierter Makro-Ereignisse nicht gespeichert. Der Fehler wurde behoben.

Beim Anbinden von Ankreuzfeldern an eine Datenbank wurden die Dateninhalte nicht korrekt gelesen oder gespeichert. Der Fehler wurde behoben.

## Build 4.1.0.1

### Neue Makrobefehle

Es wurden folgende neuen Makrobefehle implementiert:

*GetAssociatedComboValue*  
*GetAssociatedComboColor*  
*GetChangeFlag*  
*NoCloseOnEscape*  
*HideMenu*

Alle Befehle sind in der Dokumentation *formmac.pdf* beschrieben.

## Build 4.1.0.0

### Speichern von Formularen als XML-Datei

Das Dateiformat XML wurde als Speichermedium für Formularvorlagen implementiert. Auf diese Weise können alle Komponenten eines Formulars auch klarschriftlich eingesehen werden. Die Codierung der XML-Dateien erfolgt in UTF-8.

### Neues Formular für die Gestaltung von Ordnerrücken

Die Vorlagenbibliothek wurde um das Verzeichnis *Ordnerrücken* erweitert. In diesem Verzeichnis wurde insbesondere ein Formular für das Ausgeben farbcodierter Ordnerrücken implementiert.

## Build 4.0.0.1

### Neues globales Makroereignis

Es wurde das Makroereignis *Beim Klicken auf einen Button* in die Liste der globalen Formular-Makroereignisse aufgenommen. Damit ist es möglich an zentraler Stelle darauf zu reagieren, wenn irgendein Button des Formulars gedrückt wird.

### Neue Makrobefehle

Es wurden folgende neuen formularspezifischen Makrobefehle implementiert:

*GetPointedObjectHandle*  
*GetObjectDisplayRect*  
*GetPageDisplayRect*

Die Befehle sind in der Dokumentation für formularspezifische Makrobefehle *formmac.pdf* beschrieben.

Es wurden folgende neuen allgemeinen Makrobefehle implementiert:

*SendMail*  
*GetCommandLineCount*  
*GetCommandLineParam*  
*DeleteProfileItem*  
*DeleteProfileSection*  
*DeleteRegistryKey*  
*DeleteRegistryItem*  
*CreateHash*

Die Befehle sind in der Dokumentation für allgemeine Makrobefehle *intfunc.pdf* beschrieben.

#### Farbige Einträge in Auswahlfeldern

In Auswahlfeldern kann ab sofort die Farbe jedes Eintrags vorgegeben werden. Die Voreinstellung ist wie bisher schwarz. Die Farbe kann als R,G,B-Wert in der Liste der Einträge eingegeben werden.

#### Gestaltung der Farbe des Balkencursors in Auswahlfeldern

Damit der Balkencursor sich bei der Verwendung von Farben in Auswahlfeldern besser abhebt, kann dessen Farbe ebenfalls vorgegeben werden. Hierfür ist eine Schaltfläche im Eigenschaften-Dialog vorgesehen.

Die Schriftfarbe innerhalb des Balkencursors wird automatisch auf *weiß* umgestellt, wenn der Balkencursor eine zu dunkle Farbe erhält.

## Build 4.0.0.0

#### Verfügbarkeit der Anwendung in Englisch und Deutsch

Beide Anwendungen (*Ausfüller* und *Editor*) der Formularsoftware *[e] forms & more* stehen ab sofort zweisprachig zur Verfügung. Es kann im laufenden Betrieb zwischen den Sprachen *Englisch* und *Deutsch* umgeschaltet werden. Die Umschaltung erfolgt im Menü *Extras*.

#### Installation in Englisch und Deutsch

Die Installation der Anwendung ist ebenfalls mehrsprachig gestaltet worden. Vor dem Beginn der Installation kann der Anwender die gewünschte Landessprache auswählen (*Englisch* oder *Deutsch*).

Die für die Installation ausgewählte Sprache wird auch als Voreinstellung der Landessprache der Anwendungen verwendet.

Ungeachtet dieser Voreinstellung kann der Anwender später zur Laufzeit jederzeit die Landessprache wieder umschalten.

### Kontextmenüs für Seiten und Objekte

Es wurden spezifische Kontextmenüs für folgende Klicks mit der rechten Maustaste implementiert. Folgende Operationen können auf diese Weise aufgerufen werden:

- Neue Seite anlegen und anhängen
- Neue Seite anlegen und vor der aktuellen Seite einfügen
- Aktuelle Seite löschen
- Alle markierten Objekte löschen
- Die Schriftart aller markierten Objekte bearbeiten
- Die Makros eines Objektes bearbeiten
- Die Eigenschaften eines Objektes bearbeiten
- Die gemeinsamen Eigenschaften aller markierten Objekte bearbeiten

Die Funktionen sind in verschiedenen Kontextmenüs untergebracht. Welches Kontextmenü erscheint, hängt davon ab, wohin der Anwender klickt. Es wird unterschieden zwischen:

- Klicken auf eine Seite (in den Leerraum zwischen den Objekten)
- Klicken auf ein Objekt
- Klicken auf eine markierte Gruppe von Objekten

### Benutzerdefinierte Makroereignisse

Die Verwaltung der Makroereignisse wurde dahingehend erweitert, dass zu den vordefinierten Ereignissen, an denen Makros ausgeführt werden, jetzt auch benutzerdefinierte Ereignisse hinzugefügt werden können.

Benutzerdefinierte Ereignisse können durch einen entsprechenden Makrobefehl angestoßen werden (*CallFormMacro*, *CallObjectMacro*).

### Sammeleditor zur Bearbeitung aller Makros auf einmal

Es wurde eine Funktion implementiert, die es erlaubt, den Makro-Code aller Makroereignisse eines Objektes oder des gesamten Formulars auf einmal zu bearbeiten. Diese Funktion spart Zeit, wenn zu einem Objekt sehr viele Makroereignisse angelegt sind und bearbeitet werden sollen.



### Makrobefehle zur Gestaltung der Buttonzeile des Ausfüllers

Es wurde eine Reihe von Makrobefehlen implementiert, die es erlauben, die Buttonzeile des Ausfüllers dynamisch zu modifizieren oder auch völlig neu zu gestalten. Dadurch kann die Buttonzeile in Abhängigkeit des Formulars gestaltet werden, das gerade bearbeitet wird.

Die Befehle beginnen alle mit „*Abb...*“. Dieser Term steht für *Application Button Bar*. Folgende Befehle wurden implementiert:

<i>AbbAppendButton</i>	Erzeugen und anhängen eines neuen Buttons
<i>AbbDeleteButton</i>	Löschen eines Buttons
<i>AbbDeleteAllButtons</i>	Löschen aller Buttons der Buttonleiste
<i>AbbLoadButtonsFromProfile</i>	Laden der gesamten Buttonleiste aus einer Profile-Datei
<i>AbbSaveButtonsToProfile</i>	Schreiben der gesamten Buttonleiste in eine Profile-Datei
<i>AbbSetButtonAction</i>	Zuweisen einer <i>benannten Aktion</i> zu einem Button
<i>AbbSetButtonBitmapFile</i>	Zuweisen einer <i>Bitmap-Datei</i> zu einem Button
<i>AbbSetButtonMacroFile</i>	Zuweisen einer <i>Makro-Datei</i> zu einem Button
<i>AbbSetButtonSignum</i>	Zuweisen eines <i>Beschriftungstextes</i> zu einem Button

Die Makrobefehle zur Gestaltung der Buttonzeile in der Dokumentation für *Allgemeine Makrobefehle* beschrieben – nicht in der Dokumentation der Makrobefehle für *[e] forms & more*. Diese Dokumentation sich in der Datei *intfunc.pdf*.

### Makrobefehle zum Lesen und Setzen Makrocodes

Es wurde folgende Makrobefehle implementiert, die es erlauben den Code eines jeden Makroereignisses eines Objektes oder eines globalen Formularmakros zu lesen oder zu setze:

*GetMacroCode*  
*SetMacroCode*

Die Befehle sind in der Dokumentation für Makros der Formularsoftware *formmac.pdf* beschrieben.

### Makrobefehle für Farben in der Registry

Es wurde folgende Makrobefehle implementiert, die es erlauben, Farben in der Registry abzulegen und wieder auszulesen:

*ReadColorFromRegistry*  
*WriteColorToRegistry*

Die Befehle sind in der Dokumentation *formmac.pdf* für allgemeine Makrobefehle beschrieben.

#### Versenden eines Formulars als Mail

Es wurde eine Funktion implementiert, die es erlaubt, das Formular als Email zu versenden. Dabei kann ausgewählt werden, ob der Dateityp *fvl* oder *fml* erzeugt wird.

Wenn der Export nach HTML lizenziert ist, ist auch ein Versenden des Formulars als HTML-Datei möglich.

Ferner kann auch ein Extrakt aller Eingabefelder als Profile-Datei versendet werden. Der Versand des Emails erfolgt mittels des Microsoft-MAPI-Interfaces. Dieses erfordert, dass beim Versenden der bevorzugte Mail-Client (z.B. Outlook) gestartet ist. Ansonsten ist der Versand ohne Wirkung.

#### Fehlerbehebungen

##### **Absturz in der Vorlagenbibliothek**

Es wurde ein Fehler behoben, der beim Bearbeiten der Vorlagenbibliothek zum Absturz führte.

## Version 3.5

### Build 3.5.0.2

#### Auswahl des Postscript-Druckers beim PDF-Export

Beim Eingeben der Einstellungen für den PDF-Export wird für die Eingabe des Druckertreibers für die Umwandlung in *PostScript* ab sofort eine Liste der installierten Druckertreiber angezeigt, so dass der Anwender einen Treiber auswählen kann. Auf diese Weise werden Eingabefehler vermieden.

Wenn ein neues, leeres Formular angelegt wird, wird automatisch der zuletzt ausgewählte Druckertreiber selektiert.

#### Prüfung der Eindeutigkeit von Objektnamen beim PDF-Export

Beim Exportieren eines Formulars nach PDF kann ab sofort eine Option gewählt werden, die dafür sorgt, dass für den Fall dass mehrere Eingabeobjekte verschiedenen Typs den gleichen Namen haben, automatisch eindeutig benannt werden. Auf diese Weise wird beispielsweise vermieden, dass ein Eingabefeld und ein Ankreuzfeld mit gleichem Namen existieren. Letzteres führt nämlich im PDF-Formular zu Verwirrungen.

#### Farbige Texte in RTF-Objekten

Es wurden Fehler bei der Darstellung von farbigen Texten in RTF-Objekten behoben.

#### Auflösung der Bilder in PDF

Beim Export von Formularen nach PDF wurde die Einstellung für die Auflösung von Bildern in der Adobe Konfigurationsdatei *form.joboptions* von 75 auf 600 erhöht. Diese Datei wird bei der Installation aufgespielt und konfiguriert alle Parameter für die Erzeugung der PDF-Datei. Dadurch wird der Effekt vermieden, dass manche Bilder sehr verschwommen exportiert wurden.

#### Prüfung beim Speichern, ob das Medium voll ist

Beim Speichern von Formularen wird ab sofort geprüft, ob das Zielmedium genügend Platz bietet, um das zu speichernde Formular aufzunehmen. Falls dies nicht der Fall ist, wird das Speichern abgelehnt, anstatt die voriger Version der Formulardatei zu überschreiben und damit (weil aus Platzmangel unvollständig) zu beschädigen.

### Einschränkung der Javaskripte für Eingabeobjekte

Für die Objekte *Button* und *Ankreuzfeld* wurde der Javascript-Event *Berechnen abhängiger Felder* entfernt, da dieser in PDF-Dateien nicht hinreichend unterstützt wird. Formulare, in denen dieser Event mit Code unterlegt ist, werden automatisch so modifiziert, dass der Code für dieses Event entfernt wird.

## Build 3.5.0.1

### Neue Makrobefehle

Es wurden folgende neue Makrobefehle implementiert. Diese lauten:

<b>LoadFile</b>	Einlesen einer ganzen Datei in einen Speicherbereich
<b>ScanFileForRegExp</b>	Durchsuchen einer Datei nach einem regulären Ausdruck
<b>ScanBufferForRegExp</b>	Durchsuchen eines Puffers Datei nach einem regulären Ausdruck

Eine detaillierte Beschreibung steht in der Dokumentation *intfunc.pdf* zur Verfügung. Die Dokumentation kann von unserer Webseite [www.waimea.de](http://www.waimea.de) unter der Rubrik *Downloads* und *Dokumentationen für Makros* heruntergeladen werden. Die Dokumentation heißt *Allgemeine Makrobefehle*.

## Build 3.5.0.0

### Neuimplementation des Tabellenobjektes

Die Tabellenobjekte wurden vollständig neu implementiert. Dadurch stehen ab sofort verschiedene Erweiterungen zur Verfügung, Außerdem wurde der Weg geebnet, um weitere Verbesserungen zu implementieren.

Alle existierenden Formulare mit Tabellenobjekten können aufwärtskompatibel weiterverwendet werden.

Die einzelnen Verbesserungen werden in den folgenden Kapiteln beschrieben.

### Automatisches Anlegen der Spalten in Tabellenobjekten

Wenn in einem Tabellenobjekt die Spalten automatisch auf Basis der Spalten der angebenen Datenbank angelegt werden, werden ab sofort folgende Eigenschaften der Datenbankspalten ausgewertet und für die Spalten des Tabellenobjektes übernommen:

- der Datentyp wird übernommen
- numerische Spalten automatisch auf *rechtsbündig* gesetzt.
- das Zahlformat für ganze Zahlen wird auf eine neutrale Form normiert.
- die Eingabelänge wird auf die maximale Zeichenzahl der Datenbankspalte begrenzt.
- Spalten, die in der Datenbank das Attribut *readonly* haben (z.B. automatisch hochzählende Spalten), erhalten auch im Tabellenobjekt das Attribut *readonly*.

#### Drucken von Tabellenobjekten

Tabellenobjekte werden ab sofort auch gedruckt. Es wird nur der jeweils sichtbare Teil der Spalten und Zeilen ausgedruckt.

#### Farbige Darstellung negativer Zahlen

Die farbige Darstellung negativer Zahlen wurde implementiert.

## Version 3.4

### Build 3.4.0.3

#### Farbige Absetzung der geraden Zeilen in Tabellenobjekten

In Tabellenobjekten wurde die Möglichkeit implementiert, die geraden Tabellenzeilen farbig abzusetzen, so dass die Zeilen immer abwechselnd eingefärbt sind. Dadurch wird die Lesbarkeit großer Tabellen erhöht.

### Build 3.4.0.2

#### Fehlerbehebungen

##### Zugriff auf ODBC-Datenbanken

Beim Zugriff auf ODBC-Datenbanken erfolgte der Zugriff auf manche Felder nicht korrekt. Der Fehler wurde behoben.

##### Export von API-Funktionen für die Methode *STDCALL*

Der Export der folgenden Funktionen für die Aufrufmethode *STDCALL* wurde ergänzt bzw. bereinigt:

- SFormObj\_GetShortComment
- SFormScript\_GetItem
- SFormScript\_SetItem

##### Lesen der Formularfelder aus INI-Dateien

Beim Lesen von Formularfeldern vom Typ *Datum* aus INI-Dateien (Profiles) wurde die früher funktionierende Variante, dass eine sechstellige Zahl als korrektes Datum der Form TTMMJJ interpretiert wird, wieder implementiert.

### Build 3.4.0.1

#### Verbesserte automatische Spaltenauswahl in Tabellenobjekten

Die Funktion *Spalten neu aufbauen*, im Dialog für die Eigenschaften eines Tabellenobjektes wurde dahingehend erweitert, dass alle hierdurch automatisch angelegten Spalten den sel-

ben Datentyp erhalten, den sie auch in der Datenbank haben. Bisher wurden immer nur Spalten vom Typ *Text* erzeugt.

### Fehlerbehebungen

#### **Einfärbung von negativen Zahlen in Tabellenobjekten**

Wenn für negative Zahlen in Tabellen eine besondere Farbe definiert wurde, blieb diese ohne Verwendung. Der Fehler wurde behoben. Negative Zahlen werden jetzt in der entsprechend definierten Farbe ausgegeben.

#### **Programmabsturz beim Bestätigen des Selektionsmodus in Tabellen**

Wenn eine Tabelle an ein ODBC-Datenzugriffsobjekt angebunden wurde und dann für diese Zugriffsobjekt die Eingabe von Selektionsbegriffen (Matchcodes) eingeschaltet und bestätigt wurde, geriet die Software in eine endlose Schleife. Das Programm konnte nur noch mit dem Taskmanager abgebrochen werden. Der Fehler wurde behoben.

## **Build 3.4.0.0**

### Reorganisation des ODBC-Zugriffs

Der Zugriff auf ODBC-Datenbanken mittels des entsprechenden Zugriffsobjekts wurde vollständig erneuert. Dabei wurden die im folgenden beschriebenen Verbesserungen implementiert und es müssen die nachstehenden Besonderheiten berücksichtigt werden:

#### **Spaltennamen ohne Qualifizierung**

Der durch das Zugriffsobjekt selektierten Tabellenspalten werden ab sofort als unqualifizierter Name (also ohne dass die Tabelle durch einen Punkt getrennt vorangestellt wird) gehandhabt. Dies gilt auch dann, wenn die Selektion aus mehreren Tabellen erfolgt. Wenn mehrdeutige Spaltennamen selektiert werden, müssen diese durch einen Aliasnamen eindeutig gemacht werden.

Durch diese Maßnahme können nunmehr alle Datenbanktypen unterstützt werden. Es gab bisher immer wieder Schwierigkeiten mit Datenbanktreibern für Oracle und für Texttreiber, die auf einer CSV-Datei aufsetzen. Leider verhalten sich die einzelnen Datenbanken hier nicht konform. Daher war diese Maßnahme nötig.

#### **Spaltennamen unterscheiden nicht die Groß- und Kleinschreibung**

Ab sofort können alle Bezüge auf Spaltennamen in Groß- oder Kleinschreibung erfolgen. Es wird nicht mehr unterschieden. Damit wird auf die Tatsache Rücksicht genommen, dass die Unterscheidung auch in den Datenbanken selbst nicht getroffen wird.

**Unterstützung von Text-Treibern**

Der ODBC-Zugriff auf einfache Textdateien (z.B. CSV-Dateien oder TAB-separierte Dateien) wird ab sofort ausdrücklich unterstützt. Microsoft liefert einen solchen Texttreiber zusammen mit der ODBC-Installation aus.

**ACHTUNG:**

Formulare, deren Spaltenbezüge qualifiziert sind (durch Punkt vorangestellter Tabellename), müssen geändert werden. Der Falls ist aller Wahrscheinlichkeit nach selten, stellt aber eine mögliche Inkompatibilität dar.



## Version 3.3

### Build 3.3.0.19

#### Neue Makrofunktionen

Es wurden folgenden neuen Formular-Makrofunktionen implementiert:

GetLongFormatted  
SetLongFormatted  
GetFloatFormatted  
SetFloatFormatted  
CopyText

Diese Funktionen sind in der Dokumentation *formmac.pdf* beschrieben.  
Ferner wurden folgender neue allgemeine Makrobefehl implementiert:

MakeTextInt

Dieser ist in der Dokumentation *intfunc.pdf* beschrieben.

### Build 3.3.0.18

#### Implementation der Zoomfunktionen *Zoom In* und *Zoom Out*

Im Formulareditor sowie im Ausfüller wurde jeweils das Funktionenpaar *Zoom In* und *Zoom Out* implementiert. Die beiden Funktionen erlauben es, ein Formular durch Tastendrucke (Strg +, Strg -) größer oder kleiner zu zoomen, ohne den Zoomdialog aufrufen und bestätigen zu müssen.

#### Erweiterungen im Objekt *Tabelle*

In Eingabeobjekten vom Typ *Tabelle* wurde die Möglichkeit implementiert, mehrere Spalten gleichzeitig zu markieren. In Anlehnung an die Erweiterung wurde auch ein Button *Alle Spalten markieren* implementiert.

Die Funktion *Spalte löschen* wurde dahingehend erweitert dass nicht nur eine Spalte, sondern auch alle markierten Spalten gleichzeitig gelöscht werden können.

Die Spaltenbreiten, die im Einstellungsdialog vorgewählt werden, werden gespeichert und als Voreinstellung verwendet.

Das Sortieren einer Tabelle nach einer bestimmten Spalte im Ausfüll- oder testmodus kann durch Doppelklicken des Spaltenkopfes der gewünschten Tabellenspalte erfolgen.

#### Fehlerbehebungen im Objekt *Tabelle*

In Eingabeobjekten vom Typ *Tabelle* wurden diverse Fehler behoben. Insbesondere wurden die *Hänger* beim Klicken auf einen Spaltenkopf beseitigt.

#### Fehlerbehebung beim Drucken auf vorgegebene Drucker

Beim Drucken auf explizit vorgegebene Drucker trat das Problem auf, dass manchmal gar nicht gedruckt wurde. Der Fehler wurde behoben.

## Build 3.3.0.17

### Keine Größenanpassung von mehrzeiligen Textobjekten

Das automatische Anpassen der Größe eines Textobjektes beim Speichern der Eigenschaften oder Ändern des Textes wird ab sofort bei mehrzeiligen Texten unterdrückt, da dies hier unsinnig ist.

### Eingabeobjekte in den Vordergrund heben

Es wurde eine neue Funktion implementiert, die alle Eingabeobjekte automatisch in den Vordergrund hebt, ohne dass diese zuvor markiert werden müssen. Die Funktion ist im Menü *Objekte* untergebracht.

Da dieser Zustand vermutlich immer der gewünschte ist, kann im Dialog *Einstellungen* im Menü *Extras* eine Option gewählt werden, die diese Funktion immer beim Speichern eines Formulars automatisch ausführt.

## Build 3.3.0.16

### Neue Schnittstelle zum Drucken in Dateien

Es wurde eine neue Schnittstelle implementiert, die das Drucken in Dateien ermöglicht, wenn die Formularsoftware als Komponente eingebunden wird.

## Build 3.3.0.15

### Fehlerbehebungen

#### Formatierung von Datumsfeldern

Datumsfelder enthielten seit der vorhergehenden Version nach dem Lesen aus einer ODBC-Datenbank immer den Wert 1.1.2000. Der Fehler wurde behoben.

#### Setzen von Bedingungen mittels des Makros *AccObjSetWhereClause*

Beim Setzen von Abfragebedingungen in Datenbanken trat ein sporadischer Fehler auf. Dieser wurde behoben.

### Build 3.3.0.14

#### Export nach PDF

Die Reihenfolge der Ausgabe von Rand und Innenfläche bei Rechtecken wurde vertauscht. Dadurch ergibt sich eine bessere Darstellung im exportierten PDF-Dokument.

#### Anzeige der Namen aller existierenden Eingabeobjekte

Es wurde im Menü *Formular* eine neue Funktion implementiert, die es erlaubt, eine Liste der Namen aller bisher existierenden Eingabeobjekte anzuzeigen.

### Build 3.3.0.13

#### Export nach PDF

Es wurden diverse Fehler beim PDF-Export behoben.

### Build 3.3.0.12

#### Automatisches Aufklappen von Auswahlfeldern

Für Auswahlfelder wurde eine Option implementiert, die es ermöglicht, daß diese sich beim Betreten mit dem Cursor automatisch aufklappen.

#### Fehlerbehebungen

Wenn ein Auswahlfeld vom Typ *nur Liste* mittels des Makros *SetTextoder* durch eine API-Funktion mit einem neuen Wert besetzt wurde, während der Cursor sich in diesem Objekt befand, wurde der Wert erst wirksam, wenn das Objekt verlassen wurde. Der Fehler wurde behoben.

### Build 3.3.0.11

#### Implementation einer modalen Dialogform

Es wurde eine neue API-Funktion *Form\_ModalFormDialog* implementiert, die es gestattet, ein beliebiges Formular in Form eines modalen Dialogs abzuwickeln.

Es wurden neue Formulareigenschaften hinzugefügt, die im Menü *Formular* mittels der Funktion *Betrieb als modaler Dialog* eingestellt werden können. Hier können die Dialoggröße und weitere Dialogeigenschaften eingestellt werden.

## Build 3.3.0.10

### Implementation des Exports von HTML-Links

Objekte vom Typ *Text* wurden um die Eigenschaft *HTML-Link* erweitert. Wenn zu einem Textobjekt ein HTML-Link definiert wird, wird das entsprechende Objekt beim Export nach HTML als Link anstelle eines normalen Textes exportiert.

## Build 3.3.0.9

### Erweiterte Bibliothek für reguläre Ausdrücke

Die Bibliothek für reguläre Ausdrücke für die syntaktische Prüfung von Eingabefeldern wurde um einige Vorgaben erweitert.

### Fehlerbehebungen

Es wurde ein Fehler beim Export von sich gegenseitig auslösenden Ankreuzfeldern nach PDF behoben.

## Build 3.3.0.8

### Neue Makros zum Setzen des aktuellen Rollausschnitts

Es wurde ein neues Makros *SetScrollOffset* implementiert, mittels dessen die aktuelle vertikale und/oder horizontale Rollposition gesetzt werden kann.

## Build 3.3.0.7

### Neue Makros zur Unterstützung dynamischer Eingabemasken

Es wurden neue Makros implementiert, die es unterstützen, im laufenden Betrieb Formularobjekte oder Gruppen von Objekten in ein Formular einzubinden und wieder zu entfernen. Auf diese Weise können Formulare während des Ausfüllens wesentlich modifiziert werden.

Der Makrobefehl *AppendObjectSet* fügt alle Objekte der ersten Seite eines namentlich angegebenen Formulars in das augenblicklich bearbeitete Formular ein. Die Objekte werden immer an die vorhandenen Objekte angehängt, so daß die neu eingefügten Objekte in der Eingabereihenfolge hinter den bereits bestehenden Objekten stehen.

Ferner wurde der Befehl *DeleteObjectsInRect* bereitgestellt. Dieser entfernt alle Objekte eines Formulars, welche sich in einem Rechteck befinden, dessen linker, oberer, rechter und unterer Rand dem Makrobefehl in Mikrometern mitgeteilt wird. Auf diese Weise lassen sich Objekte oder Objektgruppen auf Basis ihrer Position aus einem Formular entfernen – auch während des Ausfüllens. Wenn das Objekt entfernt wird, in dem sich augenblicklich der Cursor befindet, wird der Cursor ins erste Eingabeobjekt des Formulars gestellt.

#### Neue Makros zum Setzen von Fenstertiteln

Es wurden die beiden Makrobefehle *SetTitleMain* und *SetTitleMdi* implementiert. Mittels dieser können die Titel des Hauptfensters und des aktuellen MDI-Fensters gesetzt werden. Diese Makros sind nur im Ausfüller verwendbar.

#### Neuer Makrobefehl zum Ausführen eines Dateinamen-Dialogs

Es wurde ein neuer Makrobefehl *FilenameDialog* zum Abwickeln eines Dateinamen-Dialogs implementiert. Der Befehl ist in der Dokumentation *Allgemeine Makrobefehle* beschrieben. Es handelt sich nicht um einen formularspezifischen Befehl.

## Build 3.3.0.6

#### Unterstützung von Synonymen und Alias-Tabellen beim ODBC-Zugriff

Beim ODBC-Zugriff auf Datenbanken, wird in dem Assistenten zum Erzeugen von SELECT-Befehlen neben der Anzeige von Tabellen und Abfragen jetzt auch die Anzeige von Systemtabellen, Synonymen und Alias-Namen unterstützt. Damit kann bei vollwertigen SQL-Datenbanken besser zwischen den Objekten unterschieden werden.

#### Neuer Makrobefehl

Es wurde ein neuer Makrobefehl implementiert, der es gestattet, in Strings Zeichenfolgen durch andere zu ersetzen. Der Befehl heißt *ReplaceString*.

## Build 3.3.0.5

#### Unterstützung von Datenbankschemata bei ODBC-Zugriff

Beim ODBC-Zugriff auf Datenbanken, die sogenannte *Datenbank-Schemata* unterstützen, wurde eine Oberfläche zur Eingabe der Schemata bei der Generierung von SQL-Befehlen implementiert.

## Build 3.3.0.4

### Seriennummern werden in der Registry abgelegt

Sowohl der Ausfüller als auch der Editor legen die auf sie registrierten Seriennummern in der Untersektion *Install* der Registry ab. Dadurch können die Nummern gegebenenfalls von Makros ausgewertet und in Formulare eingedruckt werden, um den registrierten Anwender kenntlich zu machen.

### Fehlerbehebungen

#### **Fehler beim Einfügen von Objekten aus der Zwischenablage**

Wenn ein Raster von mehr als 2 mm eingestellt war (z.B. 5 mm) und Objekte in die Zwischenablage kopiert und dann wieder eingefügt wurden, konnte es zum Absturz kommen. Der Fehler wurde behoben.

#### **Fehler beim Einfügen von Objekten aus der Zwischenablage**

Wenn mehrere Objekte exakt die gleiche Größe und die gleichen Koordinaten hatten – wenn diese also praktisch genau übereinander lagen – konnte es vorkommen daß beim Kopieren in die Zwischenablage und Wiedereinfügen ein Absturz erfolgte. Der Fehler wurde behoben.

## Build 3.3.0.3

### Belegte Makros und Skripte werden markiert

In den Dialogen zur Bearbeitung von Makros und Skripten werden diejenigen Ereignisse, die wirklich mit einem Skript hinterlegt sind, markiert. Dadurch kann sofort eine Übersicht gewonnen werden, ob (und wenn ja, wie viele) Skripte vorhanden sind, ohne durch die ganze Liste laufen zu müssen.

### Mehrzeilige Textobjekte werden automatisch vergrößert

Beim Anlegen oder Ändern eines mehrzeiligen Textobjekts wird ab sofort die Höhe des Objekts vergrößert, falls der Text nicht mehr in das Objekt paßt. Die Breite bleibt bei mehrzeiligen Textobjekten immer unverändert.

### Fehlerbehebungen

#### **Export von Bildern nach HTML**

Beim Exportieren eines Formulars nach HTML wurden in Abhängigkeit der Exporteinstellungen manchmal keine Bilder exportiert. Der Fehler wurde behoben.

### Rückgängig machen

Wenn die Funktion *Rückgängig* machen aufgerufen wurde, wurde gleichzeitig die Seitenposition nach oben geschoben. Der Fehler wurde behoben.

## Build 3.3.0.2

### Schriftgrößen in Zehntelpunkten

Die Unterstützung von Schriftgrößen in Zehntelpunkten wurde implementiert. Dazu können ein oder mehrere Objekte selektiert werden. Durch Aufruf der Funktion *Eigenschaften* im Menü *Objekte* gelangt man in den Dialog zum Bearbeiten der Eigenschaften der selektierten Objekte. Hier wurde ein zusätzliches Eingabefeld implementiert, welche die Vorgabe der Schrifthöhen in Zehntelpunkten ermöglicht.

Die Eingabe der Schrifthöhe in Zehntelpunkten im Schriftendialog ist leider nicht möglich, da dies von Windows nicht unterstützt wird.

### Neue Kommandozeilenparameter

Im Formularausfüller wurden folgende neue Kommandozeilenparameter implementiert:

**/HideButtons**      Der Ausfüller wird ohne Buttonzeile gestartet.

Damit kann zusammen mit dem Schalter */HideMenu* der gesamte Fensterkopf ausgeblendet werden. Dies ist sinnvoll, wenn der Ausfüller benutzt wird, um ein bestimmtes, festes Formular darzustellen, welches die notwendigen Steuerelemente bereits innerhalb des Formulars bereitstellt. Dies ist bei Datenbankanwendungen häufig der Fall.

### Neue Makroereignisse

Es wurden die folgenden neuen Makroereignisse implementiert:

- OnAccessStatusChange
- OnModalConfirmBeforeCreateOutput
- OnRecordUnselect

*OnModalConfirmBeforeCreateOutput* wird immer dann aufgerufen, bevor ein Formular in einem modalen Dialog die Formularfelder in eine Ausgabedatei schreibt. Auf diese Weise kann die Übernahme von Formulardaten in andere Anwendungen (z.B. TexAssOffive XP) manipuliert werden. Das Ereignis steht nur in der Liste der Formularmakros und hat den Beschreibungstext *Vor dem Erzeugen der Ausgabedatei eines Modalen Formulars*.



*OnAccessStatusChange* wird immer dann aufgerufen wenn sich etwas am Status des Zugriffs auf eine Datenquelle ändert. Auf diese Weise kann z.B. eine Statusinformation generiert und angezeigt werden. Das Ereignis steht in der Liste der Ereignisse eines Datenzugriffsobjektes zur Verfügung. Es hat den Beschreibungstext *Bei Änderung des Zugriffsstatus*.

*OnRecordUnselect* wird immer dann aufgerufen, wenn eine Datensatzselektion wieder aufgehoben wird. Bisher wurde hier das Ereignis *Nach dem Zugriff auf einen Datensatz aufgehoben*. Das Ereignis steht sowohl in der Liste der Formularereignisse als auch in der Liste der Ereignisse eines Datenzugriffsobjektes zu Verfügung. Es trägt den Beschreibungstext *Nach dem Aufheben des Zugriffs auf Datensätze*.

#### Neuer API-Aufruf zum Aufrufen von eingebetteten Makros

Es wurde ein neuer API-Aufruf *Form\_CallFormMacro* implementiert. Dieser gestattet es, alle im Formular eingebetteten Makros, die bei bestimmten Ereignissen automatisch aufgerufen werden, auch explizit aufzurufen.

### Build 3.3.0.1

#### Hervorhebung des aktiven Eingabefeldes

Es wurde die Option implementiert, das augenblicklich aktive Eingabefeld optisch hervorzuheben. Die Option kann im Dialog *Anzeigeeinstellungen* im Menü *Formular* aktiviert werden.

#### Neue Funktion für den ODBC-Datenbankzugriff

Der Zugriff auf ODBC-Datenbanken wurde dahingehend erweitert, daß eine Funktion implementiert wurde, die alle Datensätze einer Tabelle selektiert, diese automatisch nach dem aktuellen Eingabefeld sortiert und auf den ersten Datensatz positioniert, der größer oder gleich der Eingabe im aktuellen Eingabefeld ist.

Nach dem Ausführen dieser Funktion sind im Gegensatz zu einer selektiven, gefilterten Selektion alle Datensätze selektiert. Allerdings wird gleichzeitig auf einen bestimmten Satz positioniert, so daß von hier aus weitergeblättert werden kann.

## Fehlerbehebungen

### Linien verändern

Beim Verändern von Linien mit der Tastatur (gedrückte Shift-Taste und Pfeiltasten) konnten Linien in einen Zustand geraten, in dem sie nie wieder angeklickt werden konnten. Diese Linien konnten daher auch nie wieder entfernt werden. Der Fehler wurde beseitigt.

## Build 3.3.0.0

### Import von PDF-Dokumenten

Für die Übernahme bestehender Formularvorlagen anderer Produkte in *[e] forms & more* wurde ein Import-Modul für den Adobe-Acrobat Version 5 erstellt, welches es erlaubt, PDF-Dateien in Formularvorlagen zu exportieren. Auf diese Weise kann eine PDF-Datei in eine Formularvorlage umgewandelt werden. Voraussetzung ist, daß eine Lizenz des Adobe Acrobat 5 vorliegt.

Im Acrobat erscheint nach korrekter Installation des Plug-In unter dem Menüpunkt *Werkzeuge* im Untermenü *PDF Consultant* der Eintrag *[e] Forms & More Converter*.

Sollte der Eintrag nicht erscheinen, so muß der Acrobat-Plug-In *PdfToForm.api* von Hand aus dem Verzeichnis *AcrobatPlugins* im Programmverzeichnis von *[e] forms & more* in das Plugin-Verzeichnis des Acrobat kopiert werden. Normalerweise erfolgt der Transfer also aus dem Verzeichnis:

C:\Programme\Waimea Software\eforms& more\AcrobatPlugins

in das Verzeichnis:

C:\Programme\Adobe\Acrobat 5.0\Acrobat\Plug\_ins

### Elegantere Bearbeitung von Makros in der Ausfüller-Buttonleiste

In dem Dialog zur Bearbeitung von Buttonleisten wurde die Bearbeitung von Makros, die einem Button zugeordnet sind, wesentlich verbessert. Die Makros können ab sofort mit einem Zeileneditor bearbeitet werden, der direkt in die Buttonleiste eingebunden ist. Ferner können für Buttons, denen noch kein Makro zugeordnet ist, automatisch leere Makros erzeugt werden.

Der Dialog zur Gestaltung der Buttonleiste wurde bei dieser Gelegenheit vollständig überarbeitet.

## Version 3.2

### Build 3.2.0.3

#### Neue Makrobefehle

Es wurden die neue Makrobefehle

**GetCurrentTime (Hour, Minute, Second)**

**GetCurrentDate (Year, Month, Day)**

zum Auslesen der aktuellen Urzeit und des aktuellen Datums implementiert.

### Build 3.2.0.2

#### Fehlerbehebung

Wenn Ausfüller oder Editor gestartet wurden, in dem im Explorer doppelt auf ein Formular oder eine Formularvorlage geklickt wurde, erschien im Dokumententitel der Dateiname im 8-zu-3-Format. Der Fehler wurde behoben.

### Build 3.2.0.1

#### Neue Makrobefehle

Es wurden folgende neue Makrobefehle implementiert:

<b>ReadIntFromRegistry</b>	Liest ganze Zahlen aus der Registry
<b>WriteIntToRegistry</b>	Schreibt ganze Zahlen in die Registry
<b>ShellExecute</b>	Startet die zu einem Dokument gehörige Anwendung
<b>FindExecutable</b>	Sucht die zu einem Dokumententyp gehörige Anwendung
<b>GetCurrentPath</b>	Liefert das Verzeichnis zurück, aus dem die Anwendung gestartet wurde

Als Konstanten für die Registry-Schlüssel können jetzt auch die Abkürzungen

**HKCU** anstelle von **HKEY\_CURRENT\_USER**

---

**HKLM** anstelle von **HKEY\_LOCAL\_MACHINE**

verwendet werden.

### Neue Kommandozeilenparameter

Im Formularausfüller wurden folgende neue Kommandozeilenparameter implementiert:

- |                         |  |
|-------------------------|--|
| <b>/HideMenu</b>        | Der Ausfüller wird ohne Menüleisten gestartet.   |
| <b>/NoCloseOnEscape</b> | Die Taste ESC schließt nicht wie sonst üblich das aktuelle Formularfenster.  |
| <b>/ButtStat=file</b>   | Der Ausfüller liest die Buttons und die Statuszeile nicht aus der vordefinierten Datei sondern aus der hier angegebenen. Die Datei ist strukturell eine Profile-Datei. |
| <b>/Title=xxx</b>       | Der Titel des Hauptfensters wird mit der angegebenen Zeichenfolge überschrieben.   |

Eine vollständige Dokumentation aller Kommandozeilenparameter finden Sie in der API-Dokumentation *formdev.pdf*.

### Überarbeitung der Installation

Die gesamte Installation wurde erneut überarbeitet. Insbesondere wurden allgemeine Standards berücksichtigt. Insbesondere wurden folgende wesentlichen Änderungen vorgenommen:

- Dynamische Bibliotheken (DLLs) werden ab sofort nicht mehr in das Systemverzeichnis des Betriebssystems sondern in dasjenige Verzeichnis installiert, in dem auch die ausführbaren Programme liegen. Durch diese Maßnahme wird die Installation mehrerer Produkte, die identische DLLs verwenden bzw. die Installation desselben Produktes in verschiedenen Versionen gegeneinander entkoppelt. Der Nachteil, daß ggf. dynamische Bibliotheken mehrfach installiert werden, ist bei den heutigen Plattenkapazitäten nicht mehr wirklich von Nachteil.
- Alle Demodaten, d.h. Formulare, Beispieltex te, Beispielbilder und auch die Beispieldatenbank werden nicht mehr in das Verzeichnis installiert, in dem die Anwendung selbst liegt, sondern in das *Anwenderdatenverzeichnis*. Dieses Verzeichnis ist benutzerspezifisch, d.h., wenn mehrere Benutzer die Anwendung installieren, sind ihre Daten gegeneinander entkoppelt.

- Die installierten Beispielformulare haben ab sofort lange, sprechende Dateinamen. Dies gilt nicht für die in sich geschlossene Formularbibliothek, die nach wie vor als geschlossene Leistungseinheit zur Verfügung steht.
- Alle Beispielbilder werden ab sofort als JPG-Dateien installiert und nicht mehr als Bitmaps. Dies verkleinert sowohl die Setup-Dateien als auch die fertigen Installationen. Durch die stark verbesserten Möglichkeiten des Imports und Exports von Grafikformaten, die in den letzten Versionen entwickelt wurden, ist das JPG-Format nicht mehr nachteilig gegenüber dem Bitmap-Format – aber viel kleiner.

#### Export nach HTML

Objekte, die die Eigenschaft *unsichtbar* (die Eigenschaft *anzeigen* ist ausgeschaltet) haben, werden jetzt auch mit dieser Eigenschaft nach HTML übertragen, d.h. sie sind dann auch in der HTML-Datei unsichtbar, aber dennoch vorhanden.

#### Export nach PDF

Der Export von Formularen in PDF-Dateien wurde in vielen Punkten verbessert. Folgende wesentliche Verbesserungen wurden implementiert:

- Schaltflächen werden jetzt als PDF-Objekte in die PDF-Datei übertragen. Dadurch lassen sich Eigenschaften, wie *Drucken* oder *Anzeigen* mit in die PDF-Datei übertragen und sind dann beim Ausfüllen des PDF-Formulars wirksam. Außerdem verhalten sich die Schaltflächen in der PDF-Datei jetzt genau wie solche, die mittels des Acrobatens selbst angelegt wurden. Schaltflächen lassen sich jetzt auch 3D-technisch herunterdrücken.
- Die Schrifthöhe in Auswahlfeldern (Comboboxen) wird jetzt in die PDF-Datei übernommen und hat dort die selbe Größe, wie in der Formularvorlage.
- Die Eingabe von Datumsangaben in Form einer sechsstelligen Zahl, wie z.B. 030165 für den 3. Januar 1965, kann jetzt auch in PDF-Dateien angewendet werden.
- Gegenseitig auslösende Ankreuzfelder (sogenannte Radio-Buttons) werden jetzt korrekt nach PDF abgebildet. Wichtig ist nach wie vor, daß alle zu einer Gruppe gehörenden Ankreuzfelder, die sich gegenseitig auslösen sollen einen Unterschiedlichen Eintrag für das Feld *Text wenn angekreuzt* haben müssen. Hingegen dürfen Ihre Namen identisch sein. Letzteres ist manchmal sogar unbedingt nötig. Dies gilt insbesondere, wenn das Formular nicht nur nach PDF sondern auch nach HTML exportiert werden soll.

## Fehlerbehebungen

### **Umwanden von Rechtecken in Texte oder Eingabefelder**

Wenn ein Rechteck in ein Textobjekt oder in ein Eingabefeld umgewandelt wurde und anschließend der Eigenschaften-Dialog des Textobjekts oder Eingabefeldes aufgerufen wurde (rechte Maustaste oder Doppelklick), dann konnte der Dialog nicht mehr mittels der Buttons *OK* oder *Abbruch* beendet werden. Dieser Fehler wurde behoben.

### **Begrenzung von mehrzeiligen Texten auf 9999 Zeichen**

Die Länge der Texte in mehrzeiligen Eingabefeldern war auf 9999 Zeichen begrenzt. Diese Schranke wurde auf 30.000 Zeichen erhöht.