

Leitfaden zur Entwicklung von orgAnice-AddOns

**orgAnice Customizer und
orgAnice Installer**

orgAnice Software GmbH

Stand: August 2014

Inhaltsverzeichnis

1	Einführung.....	7
1.1	Was sind der orgAnice Customizer und der orgAnice Installer?.....	7
1.2	Was sind ihre grundlegenden Funktionen?.....	7
1.3	Welche Änderungen werden eingespielt?	7
2	Neuerungen.....	8
2.1	Neuerungen seit der Version 1.0.301.....	8
2.2	Neuerungen seit der Version 1.0.315.....	8
3	Programmierregeln	10
3.1	Löschen.....	10
3.2	Ändern.....	10
4	Was ist zu beachten, wenn Sie ein neues AddOn entwickeln?	11
4.1	Namensregeln bei neuen Objekten	11
4.2	Löschen von Objekten	11
4.3	Ändern von Objekten	12
4.4	Standard-VBA-Schnittstellen im Modul <Präfix>_Macros	13
4.5	Aufteilung in mehrere AddOns	14
4.6	Versionierung von AddOns	14
4.7	AddOn-Eigenschaften	15
5	Was ist zu beachten, wenn Sie ein Update Ihres AddOns entwickeln?.....	22
5.1	Änderungen von „öffentlichem“ Programmcode	22
5.2	Öffentliche Variablen und Eigenschaften	23
6	Was ist zu beachten, wenn Sie Anpassungen an einer Datenbank vornehmen?	24
7	Allgemeine Hinweise & Tipps.....	25
8	Erweiterung der Funktionalität des Installers / Customizers durch eigenen Code	26
9	Bedienung des orgAnice Customizers	27
9.1	Menüpunkte.....	29
9.2	Interpretation der Vergleichsergebnisse.....	30
10	Auslieferung Ihrer Änderungen mit dem orgAnice Installer.....	33

10.1	Voraussetzungen.....	33
10.2	Installationspaket erstellen	33
11	Bedienung der orgAnice Installers.....	36
11.1	Auswahl der Zieldatenbank	36
11.2	Erstellung einer Sicherungskopie.....	36
11.3	Auswahl der einzuspielenden Änderungen.....	37
11.4	Einspielen des Updates.....	41
11.5	Fertigstellung.....	41
11.6	Sicherungskopien	41
12	Steuerung des Installers mit Kommandozeilenparametern.....	43
12.1	Beispielaufrufe.....	44
12.2	Parameterabhängigkeiten	44
12.3	ExitCodes.....	45
13	Installation von zusätzlichen Dateien mit dem Installer	46
13.1	Unterordner „DB-Folder“	46
13.2	Selbstentpackende Datei „DB-Folder.exe“	46
13.3	Beispiel	46
14	Aufruf von weiteren Programmen	48
14.1	Programmaufruf über AddOn-Eigenschaften	48
14.2	Programmaufruf über Kommandozeilenparameter	48
14.3	Übergabe von Parametern	49
15	Übertragen von Datensätzen	50
16	Anhang A – Schichtenmodell der Zugriffe	51
17	Anhang B – Objektbezogene Funktion des Customizers/Installers	52
17.1	Objekte des OrgDbServers.....	52
17.2	Objekte von OrgData	54
17.3	Allgemeine Hinweise zu Änderungen von Tabellenlayouts und Arbeitsbereichen	55
17.4	Allgemeine Hinweise zu Änderungen von Makro-Modulen	55
17.5	Objekte von OrgPrint	55
17.6	Word-Makro-Vorlagen	56

18	Anhang C – VBA Standardprozeduren	57
18.1	Auto-Makros	57
18.2	Installer-Makros	57
18.3	Konfigurations-Makros.....	58
18.4	Dokumentenbehandlung	58
18.5	Fakturierung	59

Abbildungsverzeichnis

Abbildung 1: Versionsnummer des AddOns als Datenbankeigenschaft	14
Abbildung 2: AddOn-Eigenschaften aufrufen	15
Abbildung 3: Dialog AddOn-Eigenschaften.....	16
Abbildung 4: Dialog "Datenbankeinstellungen" mit Registerkarte "AddOns"	17
Abbildung 5: Dialog "Benutzereinstellungen" mit Registerkarte "AddOns"	17
Abbildung 6: Voraussetzungen für AddOns	18
Abbildung 7: Abhängigkeiten zwischen AddOns	19
Abbildung 8: orgAnice Customizer – Lizenzierung	27
Abbildung 9: orgAnice Customizer – Hauptdialog.....	28
Abbildung 10: orgAnice Customizer - Dialog „Optionen“	30
Abbildung 11: orgAnice Customizer – Auswahl des Vergleichsmodus.....	31
Abbildung 12: IExpress starten	34
Abbildung 13: IExpress Wizard.....	34
Abbildung 14: IExpress - Auswahl des Paketumfangs.....	35
Abbildung 15: IExpress - orgInstaller.exe starten	35
Abbildung 16: orgAnice Installer	37
Abbildung 17: orgAnice Installer mit mehreren AddOns.....	38
Abbildung 18: orgAnice Installer mit abhängigen AddOns	38
Abbildung 19: orgAnice Installer - Experteneinstellungen	39
Abbildung 20: orgAnice Installer - Einspielen des Updates.....	41
Abbildung 21: Zusätzliche Dateien installieren.....	47
Abbildung 22: Beispiel-Datensätze kopieren	50
Abbildung 23: Bereiche einer orgAnice-Datenbank	51

Tabellenverzeichnis

Tabelle 1: Kommandozeilenparameter vom orgAnice Installer	43
Tabelle 2: Abhängigkeiten der Kommandozeilenparameter im orgAnice Installer.....	45
Tabelle 3: Objekte des OrgDbServers.....	53
Tabelle 4: Objekte von orgData	55
Tabelle 5: Objekte von OrgPrint.....	55
Tabelle 6: Word-Makro-Vorlagen	56

1 Einführung

1.1 Was sind der orgAnice Customizer und der orgAnice Installer?

Der orgAnice Customizer und der orgAnice Installer sind Tools, die Entwickler oder Administratoren beim Entwickeln von AddOns oder Datenbank Anpassungen (Customizing) unterstützen. Sie wurden erstellt, um solche Entwicklungen zu standardisieren und diese von einer Datenbank in beliebige andere zu übertragen.

Sie können sowohl solche Änderungen erstmalig in eine Datenbank „einbauen“ als auch spätere Updates derselben m.E. realisieren.

1.2 Was sind ihre grundlegenden Funktionen?

Der orgAnice Customizer kann zwei Datenbanken vergleichen, Unterschiede feststellen und teilweise Fehler in der Einhaltung der „Programmierregeln“ (s. 3 „Programmierregeln“, S. 10) feststellen.

Außerdem kann er die Änderungen in die andere Datenbank einspielen.

Der orgAnice Installer dient dem Einspielen der Änderungen in die Kundendatenbank. Er wird vom Entwickler mit an den Kunden ausgeliefert. Der Installer kann über Kommandozeilenparameter gesteuert werden.

1.3 Welche Änderungen werden eingespielt?

Prinzipiell werden nahezu alle Server-Objekte (außer Benutzer und bestimmte binäre Objekte), orgData-Anwendungsobjekte (inkl. VBA-Makros), orgPrint-Objekte und Änderungen an Word-Makro-Vorlagen (*.dot) beim Einspielen oder beim Vergleich berücksichtigt (eine genaue Liste siehe Anhang B – Objektbezogene Funktion des Customizers/Installers, S. 52).

Eine wesentliche Einschränkung beim Übertragen von Änderungen ist die, dass der Customizer / Installer keine Objekte löscht (Eine Ausnahme bilden VBA-Module und einzelne VBA-Prozeduren / Funktionen). Details siehe unten.

2 Neuerungen

Für Entwickler, die bereits den orgAnice Customizer und Installer in einer früheren Version kennen, sind hier die wichtigsten Änderungen, mit den Verweisen auf die entsprechenden Stellen im Dokument, zusammengefasst:

2.1 Neuerungen seit der Version 1.0.301

- Protokollfunktion im Installer und Customizer: s. 11.3.4.1 „Protokoll speichern“, S. 39
- Möglichkeit des Aufrufs von eigenen Programmen nach Durchführung der Installationsroutine mit Übergabe von Parametern: s. 14 „Aufruf von weiteren Programmen“, S. 48
- Steuerung des Installers mit Kommandozeilenparametern: s. 12 „Steuerung des Installers mit Kommandozeilenparametern“, S. 43
- Installation von zusätzlichen Dateien mit dem Installer: s. 13 „Installation von zusätzlichen Dateien mit dem Installer“, S. 46
- Möglichkeit mehrere AddOns gleichzeitig zu installieren: s. 11.3.2 „AddOn-Objekte“, S. 37
- Möglichkeit Voraussetzungen für AddOns zu definieren: s. 4.7.2 „Voraussetzungen für AddOns“, S. 18
- Neuer Customizer-Dialog "AddOn-Eigenschaften" für einfache Bearbeitung von AddOn-Eigenschaften: s. 4.7 „AddOn-Eigenschaften“, S. 15
- Die Größe der Customizer- und Installer-Fenster ist jetzt veränderbar.

2.2 Neuerungen seit der Version 1.0.315

- Hinzufügen von Unterformularen in Registerkarten ist auch für bestehende Formulare zugelassen
 - AddOns können Registerkarten-Controls in Standard-Tabellenlayouts durch eigene Registerkarten erweitern: s. 4.3 „Ändern von Objekten“, Pkt. 7 Besonderheiten bei Registerkarten, S. 13
- Übertragen von Datensätzen: In der Quell-Datenbank können Beispiel-Datensätze mitgeliefert werden, die in die Ziel-Datenbank kopiert werden. S. 15 Übertragen von Datensätzen, S. 50
- Übertragen von Änderungen auf den SQL-Server: Beim Einsatz des MS SQL Servers wird die Übertragung der Datenbankstrukturänderungen auf den SQL-Server vom Installer automatisch gestartet. S. 11.3.4.3 Änderungen auf den SQL-Server übertragen, S. 40

- Erscheinungsbild des Customizers:
 - Der Customizer besitzt jetzt eine Menüleiste
 - Buttons wurden von „unten“ in das Menü verlagert
- Performance-Optimierungen (schnelleres Einspielen von mehreren AddOns, wie bspw. beim Updaten einer CRM-Datenbank auf CRM 2008)
- Die ersetzen Objekte werden als Sicherungskopie im Unterordner „Backup\Installer“ der Ziel-DB abgelegt

3 Programmierregeln

Damit es möglich wird, ein AddOn (einen Teil der Datenbank) in eine beliebige andere Datenbank zu übertragen, ohne dass die möglicherweise vom Kunden angepasste Funktionalität verloren geht, war es notwendig, bestimmte Regeln für das Erstellen solcher AddOns und Anpassungen aufzustellen, sowie die Übertragung solcher Änderungen zu beschränken. Dies war unter anderem notwendig, um Konflikte zwischen Anpassung (Customizing), Updates der Standarddatenbank und Neu-Einspielen bzw. Updates der AddOns auf ein Minimum zu beschränken bzw. völlig zu vermeiden.

Dazu gehört, dass sich Standarddatenbank, Customizing-Änderungen sowie Änderungen durch AddOns eindeutig voneinander unterscheiden lassen, um sie später einzeln aus einer Datenbank isolieren zu können und damit ggf. Konflikte (2 Bereiche wollen dasselbe Objekt ändern) von vorneherein vermieden oder vom Programm erkannt werden können.

Hierfür gibt es nun für Entwickler von AddOns, sowie für Administratoren oder orgAnice-Partner, die beim Kunden Änderungen seiner Datenbank vornehmen, folgende Regeln.

Die Regeln sind im wesentlichen dafür einzuhalten, um die Updatefähigkeit einer Datenbank zu erhalten, d.h. es können auch nach Einspielen Ihres AddOns weitere Updates der Standarddatenbank, Ihres oder anderer AddOns eingespielt werden. Sie können Customizing-Änderungen mit Ihrem AddOn / Update nicht „zerstören“, wenn Sie sich an die Regeln halten

3.1 Löschen

Sie dürfen keine Objekte der Standarddatenbank oder anderer AddOns löschen. Solche Änderungen würden vom Customizer/Installer ohnehin ignoriert werden.

Da der Customizer auch für Updates eingesetzt wird, sollten Sie berücksichtigen, dass auch in einem Ihrer Updates beim Kunden niemals Objekte gelöscht werden – auch nicht die, die zu Ihrem AddOn gehören.

Das gleiche gilt, wenn Sie Anpassungen Ihrer Datenbank vornehmen. Wenn Sie Objekte eines AddOns oder der Standarddatenbank löschen, so würden diese beim nächsten Update ohnehin wieder eingespielt werden.

3.2 Ändern

Sie dürfen nur Objekte der Standarddatenbank und Ihres eigenen AddOns / bzw. der Custom-Objekte verändern.

Bei Änderungen von Objekten gilt die allgemeine Regel „erweitern ja, einschränken nein“ d.h. Sie können z.B. die Länge eines Feldes verlängern, aber nicht verkürzen.

Weitere Details siehe Anhang B – Objektbezogene Funktion des Customizers/Installers, S. 52

4 Was ist zu beachten, wenn Sie ein neues AddOn entwickeln?

4.1 Namensregeln bei neuen Objekten

Beachten Sie bei der Erstellung neuer Objekte die folgenden Regeln

1. Lassen Sie sich von der orgAnice Software GmbH ein zwei Zeichen langes Entwickler-Prä-Präfix geben (z.B. „XY“)
2. Hängen an das Prä-Präfix ein eigenes Präfix für Ihr AddOn (z.B. „AB“) an. Somit lautet das gesamte Präfix „XYAB“. Durch dieses Verfahren wird sichergestellt, dass die AddOn-Präfixe eindeutig sind.
3. Alle neuen Objekte tragen dieses Präfix im Namen (oder im Titel bei Objekten ohne Namen). Das Präfix wird durch einen Unterstrich „_“ vom Rest des Namens getrennt.
4. Die Unterobjekte (bspw. Felder und Indizes einer neuen Tabelle) tragen kein Präfix im Namen (außer öffentliche VBA-Routinen bzw. globale Variablen)

Beispiel: AddOn-Präfix „XYAB“

- Neue Tabelle „XYAB_MeineTabelle“
 - Felder: „Beschreibung“, „Typ“, „Datum“ (die Felder tragen als Unterobjekte keine Präfixe)
- Erweiterung der bestehenden Tabelle „Adressen“
 - Neues Feld „XYAB_NeuesFeld“
 - Neuer Index „XYAB_NeuerIndex“
 - Neues Tabelleneigenschaft „XYAB_NeueEigenschaft“
- Neue persistente Variable „XYAB_It_Werteliste“
- Neue Datenbankeigenschaft „XYAB_Eigenschaft“
- Neues Tabellenlayout „XYAB_MeinLayout“
- Neue VBA_Klasse „XYAB_CKlasse“
- Neue Public Sub in Modul XYAB_Macros „XYAB_AutoOpen“
- Neue globale Variable „XYAB_gsVariable“

4.2 Löschen von Objekten

Das Löschen von Objekten ist „verboten“ und wird auch niemals übertragen. Sobald Sie ein AddOn an Ihre Kunden ausgeliefert haben, müssen Sie nämlich davon ausgehen, dass die

Kunden Daten in den neuen Tabellen und Felder erfasst haben. Eine veröffentlichte Erweiterung kann somit nicht zurückgenommen werden.

Ausnahme: Eigene VBA-Module, Prozeduren in Ihren Makro-Modulen,

Dazu siehe: 5 „Was ist zu beachten, wenn Sie ein Update Ihres AddOns entwickeln?“, S. 22

In besonderen Fällen können Sie das Löschen von Objekten mit Hilfe von eigenem Programmcode erzwingen – hierzu müssen Sie den Code in die Prozedur <AddOn-Präfix>_BeforeUpdate platzieren (vgl. Anhang C – VBA Standardprozeduren, S. 57)

4.3 Ändern von Objekten

Das Ändern von Objekten ist nur teilweise erlaubt

1. Nicht bei allen Objekten werden Änderungen übertragen (Einzelheiten s. Anhang B – Objektbezogene Funktion des Customizers/Installers, S. 52)
2. Nur Erweitern anstatt Einschränken
3. Nur eigene Objekte oder Standard-Datenbank-Objekte
 - a. Verwenden Sie niemals Custom-Objekte in Ihrem AddOn
 - b. Ändern sie niemals Custom-Objekte
4. Besonderheiten bei Lookup-Tabellen:

Nur neue Array-Elemente werden eingespielt, nur abhängig vom Namen der Lookup-Tabelle
5. Besonderheiten bei VBA-Makros
 - a. Verändern Sie ausschließlich eigene Module und Formulare (Name mit Präfix)
 - b. Änderungen in anderen Modulen sind verboten, außer in UserMacros durch Hinzufügen oder Ändern eigener Prozeduren (mit Präfix). Diese können dann auch nicht mehr gelöscht werden.
 - c. Beim Aufruf von Prozeduren/Funktionen anderer Module verwenden Sie immer die Schreibweise **<Modulname>.<Prozedurname>**. Denn auch wenn in Ihrem AddOn der Aufruf über **<Prozedurname>** funktioniert, könnte es sein, dass in einem anderen AddOn derselbe Prozedurname in einem anderen Modul verwendet wird. Wenn beide AddOn in eine Datenbank eingespielt werden, ist der Name schon nicht mehr eindeutig und der Aufruf funktioniert nicht.
 - d. Verwenden Sie keine Formulare anderer AddOns oder der Standarddatenbank
6. Besonderheiten bei Relationen

Ein AddOn darf keine neue Tabelle relational „oberhalb“ einer Standard-Tabelle anlegen. Solche Relationen werden vom Customizer/Installer ignoriert.

7. Besonderheiten bei Registerkarten

Ein AddOn darf ein neues Unterformular in einem bestehenden Registerkartensteuerelement eines Standard-Tabellenlayouts hinzufügen. Beim Update des Standard-Tabellenlayouts wird das Unterformular übernommen.

Der Name des Unterformulars muss mit dem Präfix beginnen.

4.4 Standard-VBA-Schnittstellen im Modul <Präfix>_Macros

Jedes AddOn Muss ein Modul mit Namen „<Präfix>_Macros“ (z.B. „XYAB_Macros) enthalten. Dies sollte immer bestimmte Standard-Routinen enthalten.

(s. Anhang C – VBA Standardprozeduren)

1. Verwenden Sie zusätzlich Events, um das Verhalten der Standard-Datenbank zu beeinflussen (z.B.: Änderung von Standardformularen)

Da Sie fremden Code nicht verändern dürfen, ist es im Moment nicht möglich Standardformulare, wie z.B. den Markierungsassistenten zu verändern. Dies sollten Sie i.d.R. auch nicht tun. Falls dringender Änderungsbedarf ist, sprechen Sie mit der orgAnice Software GmbH über einen möglichen Einbau Ihrer Änderungen in die Standarddatenbank.

Für den Fall, dass Ihre Änderungen zu speziell, aber für Ihre Applikation wesentlich sein sollten, wird es in Zukunft die Möglichkeit geben, über standardisierte Aufrufe oder Events die Anzeige von Standardformularen durch die Anzeige eigener Formulare zu ersetzen.

Bis dahin empfehlen wir, durch das Abfangen der Events, die zur Anzeige der Standardformulare führen, dem Benutzer eigene zusätzliche Formulare anzuzeigen, in denen die notwendige Auswahl oder Eingabe durchgeführt werden können.

2. Programmieren Sie Ihr AddOn möglichst allgemeingültig (generisch)
 - a. Tabellennamen nicht hart-codiert verwenden
Wenn Sie eine Funktionalität programmieren, die sich auf eine oder mehrere bestimmte Tabellen bezieht, die sich aber rein theoretisch auch auf andere Tabellen beziehen kann, dann schreiben Sie den Code so, dass er anhand von Tabelleneigenschaften erkennt, welche Tabellen verwendet werden sollen. Achten Sie hierbei darauf, dass diese Tabelleneigenschaften in jedem Fall im Namen das AddOn-Präfix tragen, da sie ggf. auch bei anderen Tabellen gesetzt werden können.
 - b. Geben Sie dem Benutzer die Möglichkeit, Ihr Programm weitgehend konfigurieren zu können, sodass Customizing durch Programmierung seinerseits weitgehend unnötig wird. (siehe auch Standard-Schnittstellen)

4.5 Aufteilung in mehrere AddOns

Denken Sie frühzeitig über eine mögliche Aufteilung Ihres AddOns in kleinere Teilstücke (mehrere AddOns), denn eine spätere Aufteilung ist mit zusätzlichem Aufwand verbunden. Für eine spätere Aufteilung, die mit Umbenennen von Objekten verbunden ist, verwenden Sie am besten den orgAnice Designer. Dieses Tool kann Objekte wie Felder, Tabellen, umbenennen (indem in den meisten Fällen eine Kopie des Objekts mit dem neuen Namen erstellt wird und alle Verweise auf das neue Objekt umgestellt werden. Weitere Details s. Dokumentation zum orgAnice Designer)

4.6 Versionierung von AddOns

Damit Installer und Customizer Ihr AddOn überhaupt identifizieren kann, müssen Sie die spezielle Datenbankeigenschaft

<AddOnPräfix>_Version

als String anlegen, die die Versionsnummer des AddOns enthalten muss:

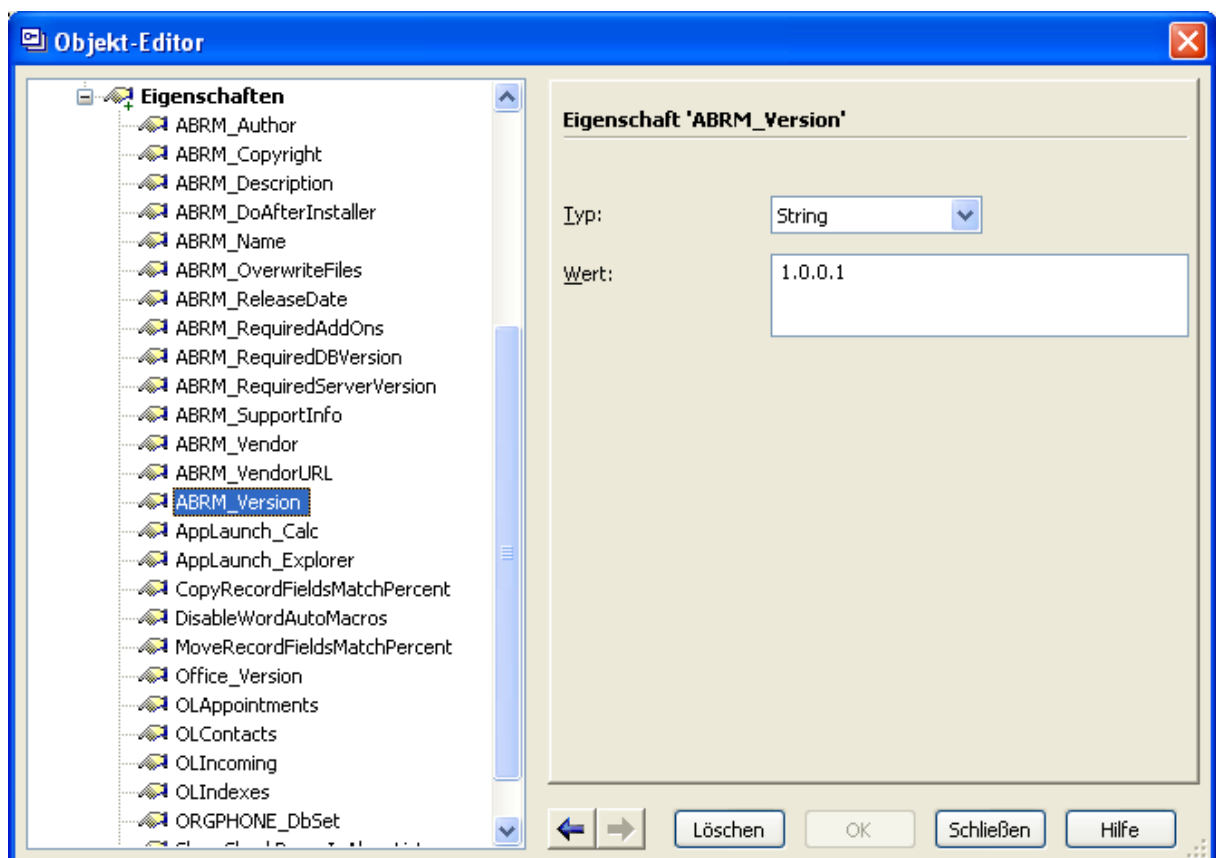


Abbildung 1: Versionsnummer des AddOns als Datenbankeigenschaft

Beispiel: „ABRM_Version“

Der Aufbau der Versionsnummer sollte analog zur orgAnice-Versionsnummer sein, also vier Gruppen von Ziffern, die durch Punkte getrennt sind.

Beispiel: „1.0.0.1“, „2.0.6.17“

Die einzelnen Abschnitte dürfen nicht länger als 12 Stellen sein.

Anhand der Versionsnummer prüft der Installer auch, ob ein Update überhaupt in eine Datenbank eingespielt werden soll (ob nicht schon eine gleiche / neuere Version vorhanden ist).

Achtung: Anhand dieser Datenbankeigenschaft wird ein AddOn als solches erkannt – fehlt diese Datenbankeigenschaft für Ihr AddOn, so wird es als Teil der Standard-Datenbank angesehen – trotz der richtigen Verwendung von Präfixen!

4.7 AddOn-Eigenschaften

AddOn können außer der Versionsnummer über weitere Eigenschaften verfügen. Einige haben einen rein informativen Charakter (z.B. Beschreibung), andere beeinflussen bspw. das Verhalten des orgAnice Installers.

Die weiteren AddOn-Eigenschaften werden als Datenbank-Eigenschaften gespeichert, Sie brauchen für ihre Bearbeitung jedoch nicht unbedingt den Objekt-Editor verwenden und die Eigenschaften selber anlegen. Denn die AddOn-Eigenschaften lassen sich bequem mit dem orgAnice Customizer erfassen. Wählen Sie im Hauptfenster des Customizers aus der Liste der verfügbaren AddOn-Präfixe das gewünschte AddOn aus und betätigen Sie die Schaltfläche „...“ neben der Versionsnummer:

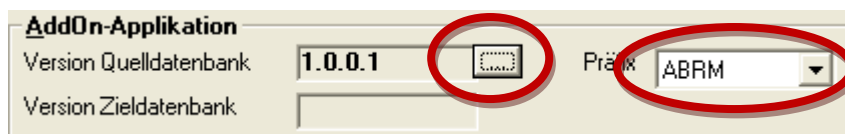


Abbildung 2: AddOn-Eigenschaften aufrufen

Hinweis: damit das AddOn und die Versionsnummer angezeigt werden, müssen Sie in der Datenbank eine Datenbankeigenschaft mit dem Namen < AddOnPräfix>_Version erstellen (s. 4.6 „Versionierung von AddOns“)

In dem Dialog „AddOn-Eigenschaften“ können Sie die definierten Eigenschaften bearbeiten.

Korrespondierende Datenbank-Eigenschaften (zzgl. Präfix):

- Name: `_Name`
- Version: `_Version`
- Autor: `_Author`
- Copyright: `_Copyright`
- Support: `_SupportInfo`
- Freigabedatum: `_ReleaseDate`
- Anbieter: `_Vendor`
- Internet: `_VendorURL`
- Beschreibung: `_Description`
- Vorausgesetzte AddOns: `_RequiredAddOns`
- Vorausgesetzte DB-Version: `_RequiredDBVersion`
- Vorausgesetzte Server-Version: `_RequiredServerVersion`
- Dateien überschreiben: `_OverwriteFiles`
- Ausführen nach Installation: `_DoAfterInstaller`

Abbildung 3: Dialog AddOn-Eigenschaften

Im Folgenden wird die Bedeutung der einzelnen Abschnitte erklärt.

4.7.1 Allgemeine AddOn-Eigenschaften

Die Allgemeinen AddOn- Eigenschaften haben einen informativen Charakter. Sie werden dem Benutzer in den Standard-Dialogen „Datenbankeinstellungen“ und „Benutzereinstellungen“ (ab Datenbankversion 2005-11) auf der Registerkarte „AddOns“ angezeigt:

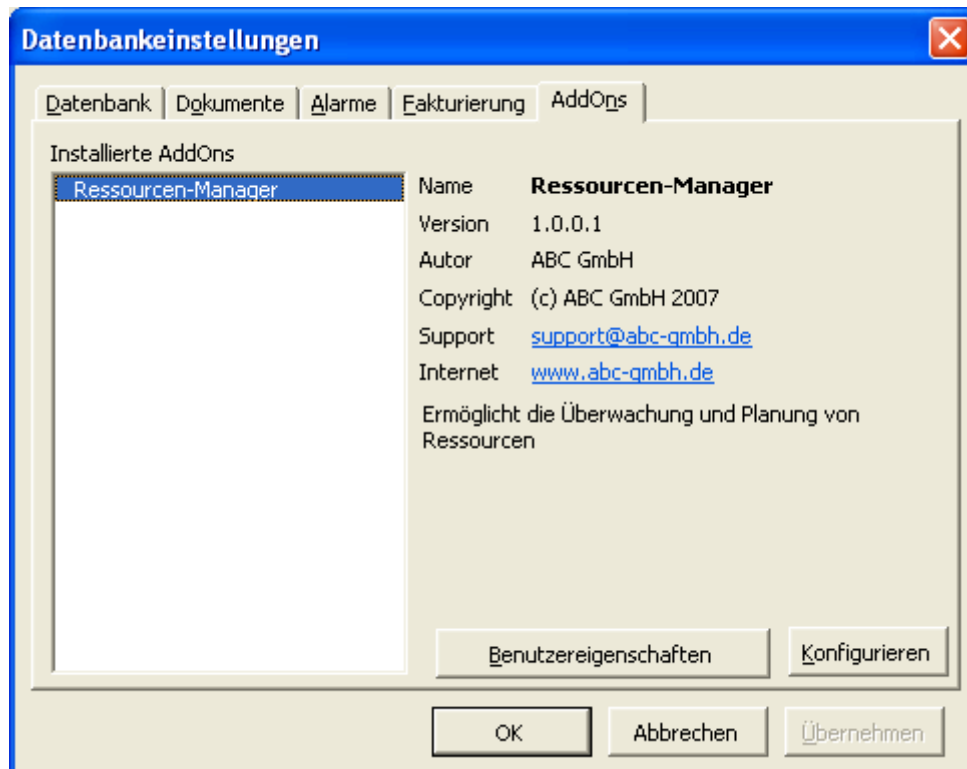


Abbildung 4: Dialog "Datenbankeinstellungen" mit Registerkarte "AddOns"

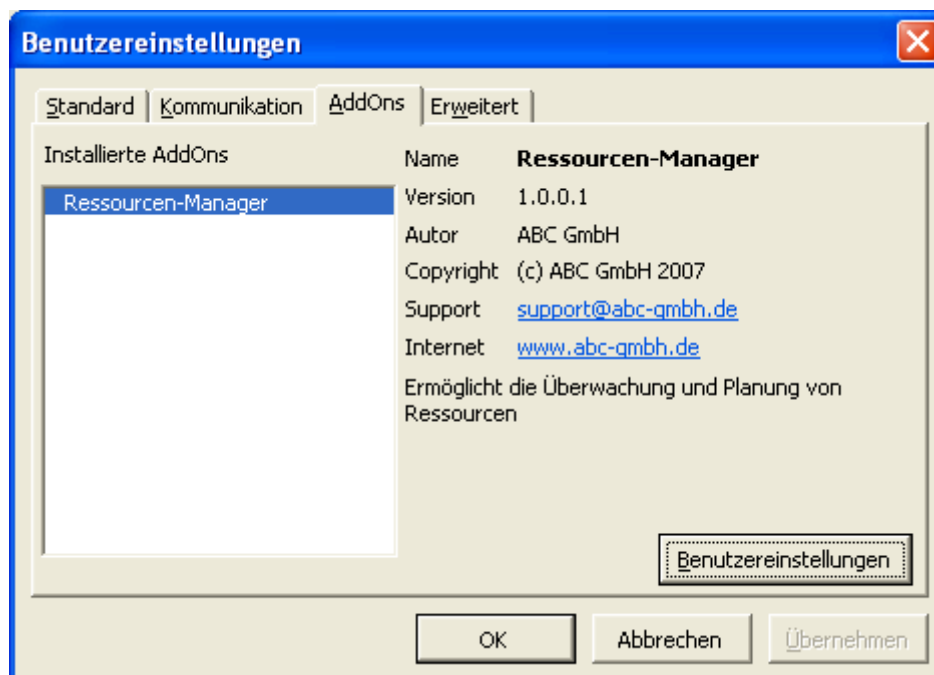


Abbildung 5: Dialog "Benutzereinstellungen" mit Registerkarte "AddOns"

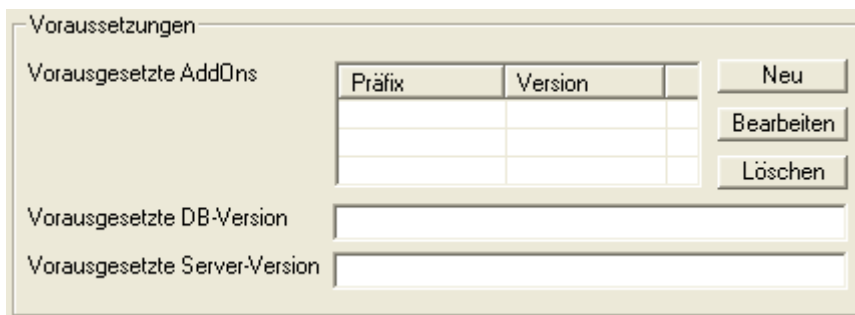
Beachten Sie, dass nicht alle allgemeinen AddOn-Eigenschaften auf der Registerkarte angezeigt werden („Freigabedatum“ und „Anbieter“ fehlen).

4.7.2 Voraussetzungen für AddOns

Sie können bestimmte Voraussetzungen für Ihr AddOn definieren, welche in der Zieldatenbank erfüllt werden müssen. Der orgAnice Installer wird Ihr AddOn nur dann einspielen bzw. updaten wenn die Voraussetzungen erfüllt sind.

Die Voraussetzungen können sich auf die folgenden Punkte beziehen:

- Das Vorhandensein eines oder mehrerer weiterer AddOns (unabhängig von ihren Versionen)
- Das Vorhandensein einer bestimmten Mindestversion eines oder mehrerer weiterer AddOns
- Das Vorhandensein einer bestimmten Mindestversion der Standard-Datenbank
- Das Vorhandensein einer bestimmten Mindestversion von orgAnice (= Version von orgAnice Datenbank Server/Data)



Vorausgesetzte AddOns	Präfix	Version	

Buttons: Neu, Bearbeiten, Löschen

Vorausgesetzte DB-Version:

Vorausgesetzte Server-Version:

Abbildung 6: Voraussetzungen für AddOns

Wie sich die definierten Voraussetzungen im Installer auswirken, sehen Sie in 11.3.2 „AddOn-Objekte“, S. 37.

4.7.2.1 Vorausgesetzte AddOns

Insbesondere mit der Möglichkeit vorausgesetzte AddOns zu definieren, können Sie Abhängigkeiten zwischen AddOns abbilden und damit AddOn-Schichten erstellen. Wenn Sie also zwei AddOns erstellen möchten, die jedoch zum Teil eine ähnliche Funktionalität besitzen, dann können Sie die gemeinsame Funktionalität in ein Basis-AddOn auslagern, welches die Voraussetzung für die zwei weiteren AddOns bildet. Das Basis-AddOn können Sie einzeln oder zusammen mit den anderen AddOns ausliefern, der Installer installiert in dem zweiten Fall das Basis-AddOn automatisch! (s. 11.3 „Auswahl der einzuspielenden Änderungen“, S. 37)

In der Abbildung 7 ist ein Beispiel dargestellt: das Basis-AddOn bildet die Voraussetzung für die AddOns A und B.

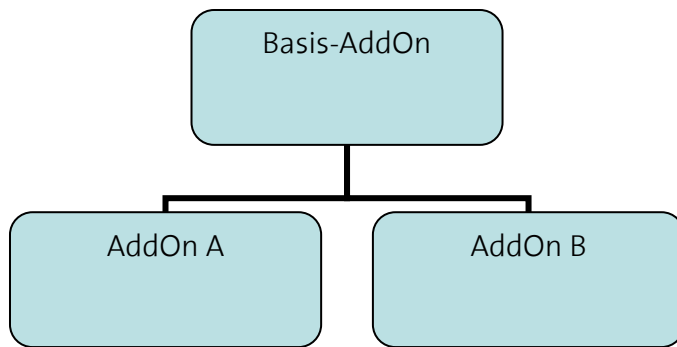
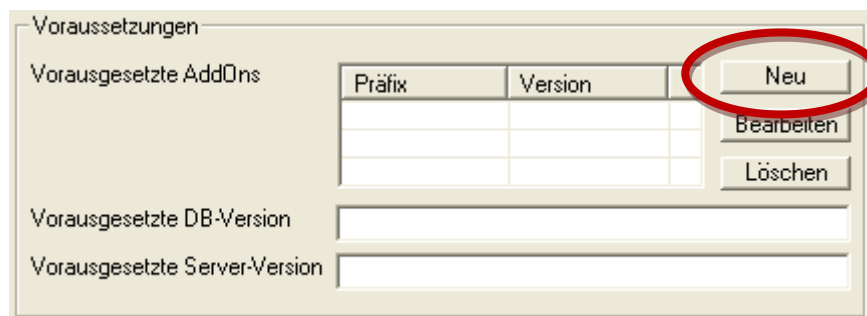


Abbildung 7: Abhängigkeiten zwischen AddOns

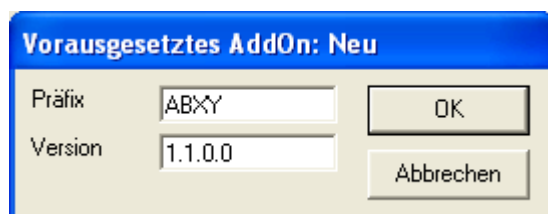
Es ist möglich für ein bestimmtes AddOn mehrere vorausgesetzte AddOns zu definieren (AddOn B ist abhängig außer vom Basis-AddOn auch von einem weiteren AddOn). Es ist des Weiteren möglich mehrstufige Abhängigkeiten zu definieren (AddOn C ist abhängig vom AddOn B, welches wiederum von dem Basis-AddOn abhängig ist), auch wenn dieser Fall in der Praxis selten vorkommen dürfte.

Um ein vorausgesetztes AddOn zu definieren, gehen Sie folgendermaßen vor:

Betätigen Sie die Schaltfläche „**Neu**“ in dem Rahmen „Voraussetzungen“ des Dialogs „AddOn-Eigenschaften“:



Geben Sie in dem folgenden Dialog „Vorausgesetztes AddOn: Neu“ das Präfix des vorausgesetzten AddOns. Die Versionsnummer ist optional, wenn sie weggelassen wird, dann reicht es, wenn das AddOn in der Zieldatenbank vorhanden ist, unabhängig von seiner Version. Wird die Versionsnummer angegeben, dann muss das AddOn in der Zieldatenbank mindestens in der vorgegebenen Version existieren.



Fügen Sie ggf. weitere vorausgesetzte AddOns hinzu.

The screenshot shows a dialog box titled 'Voraussetzungen'. It contains a table for 'Vorausgesetzte AddOns' with columns 'Präfix' and 'Version'. The table lists two entries: 'ABXY' with version '1.1.0.0' and 'OSAMG' with version '2.0.5.6'. To the right of the table are three buttons: 'Neu', 'Bearbeiten', and 'Löschen'. Below the table are two input fields: 'Vorausgesetzte DB-Version' and 'Vorausgesetzte Server-Version', both currently empty.

Bestehende Einträge können bearbeitet und gelöscht werden (Schaltflächen „Bearbeiten“ und „Löschen“)

4.7.2.2 Vorausgesetzte Datenbank-Version

Wenn Ihr AddOn die Funktionalitäten der Standard-Datenbank verwendet, die erst ab einer bestimmten Datenbank-Version zur Verfügung stehen, dann können Sie die vorausgesetzte Datenbank-Version ebenfalls definieren. Geben Sie die vorausgesetzte Datenbank-Version in dem Rahmen „Voraussetzungen“ des Dialogs „AddOn-Eigenschaften“ ein:

This screenshot is identical to the previous one, but the input field for 'Vorausgesetzte DB-Version' now contains the text '2007-07' and is circled in red.

4.7.2.3 Vorausgesetzte Server-Version

Wenn Ihr AddOn die Funktionalitäten verwendet, die erst ab einer bestimmten orgAnice- (Data/Server)-Version zur Verfügung stehen, dann können Sie diese Voraussetzung ebenfalls definieren. Geben Sie die Server-Version in dem Rahmen „Voraussetzungen“ des Dialogs „AddOn-Eigenschaften“ ein:

This screenshot is identical to the previous one, but the input field for 'Vorausgesetzte Server-Version' now contains the text '4.0.2.690' and is circled in red.

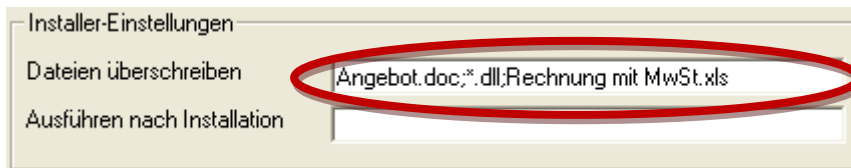
4.7.3 Installer-Einstellungen

Die Installer-Einstellungen betreffen die erweiterten Möglichkeiten der Steuerung des orgAnice Installers.

4.7.3.1 Dateien überschreiben

Die Option „Dateien überschreiben“ bezieht sich auf die Möglichkeit, beim Einspielen eines AddOns mitgelieferte Dateien in den Ziel-Datenbank-Ordner zu kopieren. Einzelheiten s. 13 „Installation von zusätzlichen Dateien mit dem Installer“, S. 46.

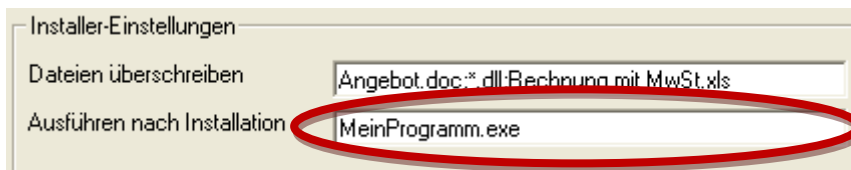
Da standardmäßig keine vorhandenen Dateien ersetzt werden, können hier Namen von Dateien angegeben werden, die auf jeden Fall (auch bei Vorhandensein) kopiert werden sollen. Wildcards sind erlaubt, die einzelnen Einträge sind durch Semikolon („;“, ohne Leerzeichen) zu trennen. Bsp: „Angebot.doc;*.dll;Rechnung mit MwSt.xls“



4.7.3.2 Ausführen nach Installation

Die Option „Ausführen nach Installation“ bezieht sich auf die Möglichkeit, nach dem Einspielen des AddOn ein weiteres Programm zu starten. Weitere Details s. 14.1 “Programmaufruf über AddOn-Eigenschaften“, S. 48.

Geben Sie hier den Dateinamen der ausführbaren Datei an, die nach der Installation des AddOns gestartet werden soll.



5 Was ist zu beachten, wenn Sie ein Update Ihres AddOns entwickeln?

Wenn sie Ihr AddOn weiterentwickeln, gelten natürlich grundsätzlich dieselben Regeln wie beim Neu-Entwickeln eines AddOns. Hierbei gibt es jedoch eine Reihe von Dingen, die Sie zusätzlich beachten müssen.

Dies ergibt sich hauptsächlich aus der Tatsache, dass Sie damit rechnen müssen, dass ein Anwender Ihre Entwicklung bereits im Rahmen eigener Anpassungen verwendet, dass er bereits Anpassungen vorgenommen hat oder dass es irgendwann auch weitere AddOns gibt, die Objekte oder Programmteile Ihres AddOn verwenden.

5.1 Änderungen von „öffentlichem“ Programmcode

Darunter verstehen wir

- VBA
 - Public Prozeduren
 - Public Funktionen
 - Klassen
 - Formulare
- Server (Parser)
 - benutzerdefinierte Funktionen

Öffentlichen Programmcode dürfen Sie nur noch eingeschränkt ändern, da dieser bereits vom Anwender in eigene Applikationen eingebunden sein kann. Daher ergeben sich folgende **Regeln**:

1. Löschen von öffentlichem Programmcode ist verboten
(VBA Prozeduren oder Funktionen werden bei Updates durch den Installer/Customizer tatsächlich gelöscht, da dieser nämlich VBA-Module immer als Ganzes einspielt)
2. Löschen von Parametern oder Ändern der Bedeutung oder des Typs der Parameter ist verboten. Weitere Parameter dürfen Sie nur als Optional hinzufügen (VBA).
3. Natürlich dürfen Sie bei Funktionen auch nicht den Typ des Rückgabewertes verändern.
4. Das Ändern der erwarteten Funktionalität sollte ebenfalls vermieden werden, also wenn z.B. eine Funktion eine Adresse mehrzeilig formatiert zurückgibt, sollte sie diese nach dem Update nicht einzeilig zurückgeben.
5. Verschieben Sie keine öffentlichen Prozeduren oder Funktionen in andere Module, da Anwender diese ebenfalls mit <Modulname>.<Prozedurname> aufrufen.

6. Gehen Sie also grundsätzlich sparsam mit öffentlichem Programmcode um, damit Sie bei eventuellen Updates keine Probleme bekommen.

5.2 Öffentliche Variablen und Eigenschaften

1. VBA
 - a. Globale Variablen / Konstanten
 - i. Löschen Sie keine globalen Variablen/Konstanten
 - ii. Ändern Sie nicht den Typ von globalen Variablen/Konstanten
 - iii. Ändern Sie nicht die Bedeutung von globalen Variablen/Konstanten
2. orgDB-Server
 - a. Datenbankeigenschaften
 - i. Prinzipiell gilt dasselbe, wie für globale Variablen und Konstanten
 - ii. Im Unterschied zu globalen Variablen, persistenten Variablen und Tabelleneigenschaften werden die Inhalte der Datenbankeigenschaften bei einem eventuellen Update Ihrer Anwendung vom Installer/Customizer überschrieben.
 - iii. Hinweis: Allgemeine Konfigurationsdaten, die vielleicht auch indirekt vom Administrator verändert werden können, sollten Sie in persistenten Variablen oder in Tabelleneigenschaften speichern (s.u.).
 - b. Tabelleneigenschaften werden nicht aktualisiert
Der Customizer aktualisiert grundsätzlich keine Tabelleneigenschaften, es werden lediglich neue eingespielt. Ihre Änderungen sind also wirkungslos. Tabelleneigenschaften sollten Konfigurationseinstellungen der Tabelle enthalten. Diese muss vom Benutzer durchgeführt werden können, also dürfen diese nicht überschrieben werden. (Sehen Sie hierzu: 4.2 Löschen von Objekten *und* Anhang C – VBA Standardprozeduren – <AddOn-Präfix>_BeforeUpdate(), S. 58)
 - c. Persistente Variablen werden nicht aktualisiert
(verhalten sich wie Tabelleneigenschaften). Sonderfall: Lookup-Tabellen
Der Customizer fügt Lookuptabellen Einträge hinzu, die bestehenden Einträge bleiben erhalten. (Sehen Sie hierzu: 4.2 Löschen von Objekten *und* Anhang C – VBA Standardprozeduren – <AddOn-Präfix>_BeforeUpdate(), S. 58).

6 Was ist zu beachten, wenn Sie Anpassungen an einer Datenbank vornehmen?

Wenn Sie Ihre Datenbank anpassen, aber gleichzeitig ihre Updatefähigkeit und der ggf. installierten AddOns erhalten wollen oder mit dem Customizer Ihre Änderungen auf andere Datenbanken übertragen wollen (oder einfach die Änderungen verfolgen wollen), müssen Sie im Prinzip dieselben Regeln einhalten, die für die Entwicklung eines AddOns gelten.

Einzige Besonderheit ist die, dass für solche Anpassungen (die wir Customizing nennen) das Präfix ‚CSTM‘ vorgesehen ist (siehe *Neue Objekte: Namensregeln beachten*). Der Unterschied zu einem AddOn besteht darin, dass Customizing nur für eine spezielle Datenbank gilt, AddOns dagegen für viele Datenbanken veröffentlicht werden.

Sie können natürlich auch ein anderes Präfix wählen, z.B. *CS<Kundenname>*, z.B. um verschiedene Änderungen unterscheiden zu können, oder um aus den Änderungen später ein Standard-AddOn zu machen. Hierfür erhält jeder interessierte Entwickler ein zwei Zeichen langes Prä-Präfix (z.B. „XY“), dem er dann für seine Erweiterungen um weitere Buchstaben und Zahlen erweitern kann. Somit kann das gesamte Präfix „XYAA“, „XYAB“ usw. lauten.

Wenn Sie nicht das Präfix „CSTM“ verwenden, dann müssen Sie allerdings in jedem Fall eine Versionsnummer für Ihre Änderungen anlegen (Siehe 4.6 „Versionierung von AddOns“, S. 14) und Sie sollten sich von der orgAnice Software GmbH einen Prä-Präfix geben lassen, damit vermieden wird, dass bereits ein weiteres AddOn mit diesem Präfix existiert.

Außerdem sollten Sie beachten, dass alle Änderungen an Objekten, die bei einem Update der Standard-Datenbank oder eines AddOns überschrieben werden, ggf. nachgeführt werden müssen. Hierzu sehen Sie bitte die Tabellen im Anhang B – Objektbezogene Funktion des Customizers/Installers, S. 52, und achten insbesondere auf die Objekte mit einem Kreuz in der Spalte Änderung / Update.

Vermeiden Sie also Änderungen an solchen Objekten, indem Sie einfach ggf. über den orgDBDesigner Kopien dieser Objekte anlegen und dann diese für Ihre Anpassungen verwenden.

7 Allgemeine Hinweise & Tipps

Verwenden Sie unbedingt den orgAnice Customizer zum Überprüfen Ihres AddOns. Hierfür empfehlen wir:

1. Entwickeln Sie möglichst auf Basis der aktuellen Standard-Datenbank
2. Sichern Sie die Ausgangsdatenbank bevor Sie mit Änderungen beginnen
3. Verwenden Sie den Customizer zum Vergleich Ihres aktuellen Standes mit der unveränderten Ausgangsdatenbank.
Der Customizer kann dann sehr leicht die meisten Abweichungen von den o.g. Regeln feststellen.
4. Bei neuen Updates der Standarddatenbank, können Sie – sofern Sie keine Regeln missachtet haben- auch mit dem Customizer Ihre Entwicklungsdatenbank updaten. Diese können Sie dann zukünftig mit dem neusten Stand der Standarddatenbank vergleichen.
5. Wenn Sie zwischendurch wissen wollen, was Sie zuletzt verändert haben, dient Ihnen der Customizer ebenfalls als Hilfe. Sichern Sie einfach Zwischenzustände Ihrer Entwicklungsdatenbank und vergleichen Sie dann diese mit dem aktuellen Stand.

8 Erweiterung der Funktionalität des Installers / Customizers durch eigenen Code

Für den Fall, dass die Funktionalität des orgAnice Installers / Customizers nicht ausreicht, um ein Update Ihrer Applikation einzuspielen, weil z.B. Änderungen vorgenommen werden müssen, die der Installer nicht durchführt (siehe Anhang), haben wir eine einfache Schnittstelle geschaffen, über die Sie einfach VBA-Code mitliefern können, der beim Update aufgerufen wird. Hierzu betrachten Sie bitte Anhang C – VBA Standardprozeduren – <AddOn-Präfix>_BeforeUpdate() und <AddOn-Präfix>_Update(). Mit *BeforeUpdate* können Sie notfalls einfach Objekte löschen, die standardmäßig nicht aktualisiert werden: sie werden dann neu eingespielt.

9 Bedienung des orgAnice Customizers

Beim ersten Start des orgAnice Customizers werden Sie aufgefordert, Ihre Lizenzinformationen einzugeben:

- Firmenname: Geben Sie hier Ihren Firmennamen ein. Er erscheint später in der Titelzeile des orgAnice Customizers.
- OFFICE-Lizenznummer: Geben Sie hier Ihre orgAnice-Lizenznummer ein, die Sie für Ihre orgAnice CRM-Applikation nutzen.
- OrgCustomizer- Lizenznummer: Geben Sie hier die Lizenznummer ein, die Sie für den orgAnice-Customizer erhalten haben. Die Lizenznummern für den OrgCustomizer und für die orgAnice CRM-Applikation müssen in der Seriennummer (hier „123456“) übereinstimmen.

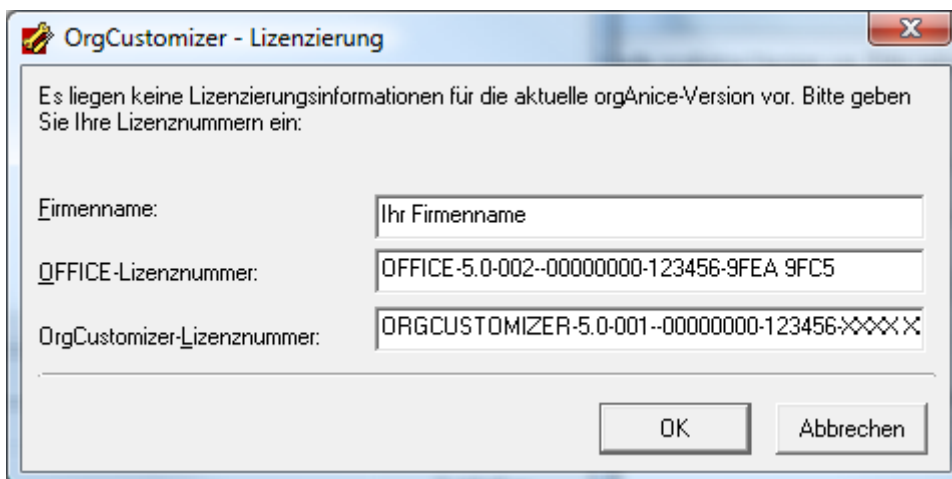


Abbildung 8: orgAnice Customizer – Lizenzierung

Nach der Eingabe der Lizenzinformationen erscheint das Hauptfenster des orgAnice Customizers. Im Folgenden werden die einzelnen Bereiche und Menüpunkte beschrieben.

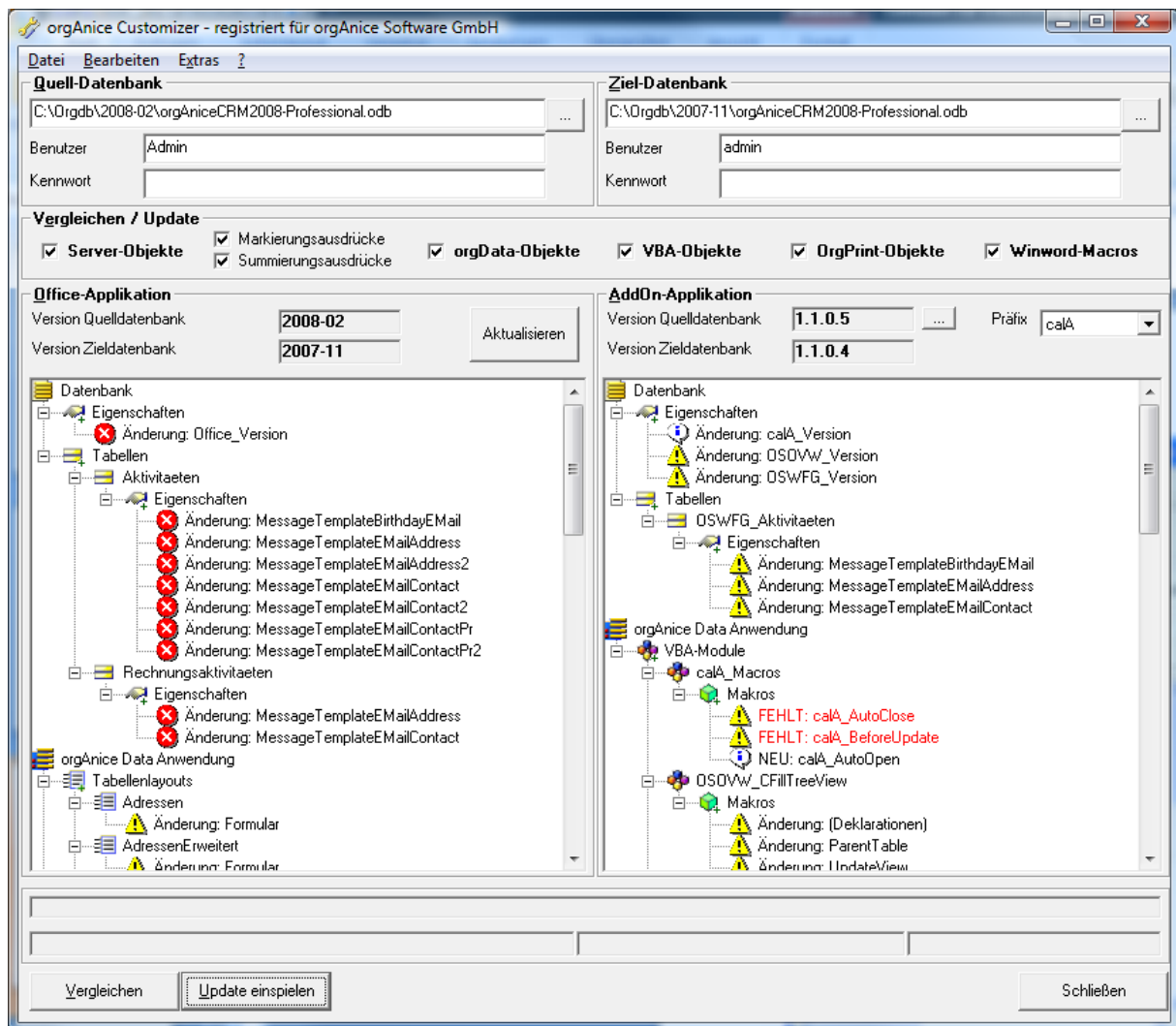


Abbildung 9: orgAnice Customizer – Hauptdialog

Quell-Datenbank: Geben Sie hier den Pfad und die Zugangsdaten zu der Quell-Datenbank (Datenbank mit den Änderungen) ein.

Ziel-Datenbank: Geben Sie hier den Pfad und die Zugangsdaten zu der Ziel-Datenbank (Datenbank, in die die Änderungen eingespielt werden sollen) ein.

Vergleichen/Update: Hier können Sie bestimmen, welche Bereiche verglichen oder eingespielt werden sollen. Mit Hilfe der Checkboxes „Server-Objekte“, „orgData-Objekte“, „Markierungsausdrücke“, „Summierungsausdrücke“, „VBA-Objekte“, „OrgPrint-Objekte“ und „Winword-Macros“ können Sie den Umfang des Vergleichs / des Updates einschränken. Deaktivieren Sie die Checkboxes, wenn die Änderungen an den entsprechenden Objekten nicht eingespielt werden sollen.

Durch das Deaktivieren von „orgData-Objekten“ können Sie bspw. verhindern, dass die Änderungen an Tabellenlayouts eingespielt werden.

Sie können diese Option auch verwenden, um Zeit zu sparen – wenn Sie wissen, dass Sie nur innerhalb der Server-Objekte Änderungen vorgenommen haben, dann brauchen Sie die zeitaufwändige Überprüfung der orgData-Objekte und der VBA-Makros nicht vorzunehmen.

Achtung: auch beim Ausschließen bestimmter Objekte wird beim Updaten die Versionsnummer des AddOns hochgesetzt – es ist im Nachhinein nicht mehr zu erkennen, ob das AddOn vollständig oder nicht eingespielt wurde. Unter Umständen wird das AddOn auch nicht richtig funktionieren, wenn nicht alle Bestandteile vorhanden sind.

Office-Applikation: im oberen Bereich des Rahmens können Sie die Version der Standard-Datenbank (Standard- Applikation) in der Quell- und der Ziel-Datenbank ablesen. Die Versionen werden angezeigt, sobald in den Bereichen „Quell-Datenbank“ bzw. „Ziel-Datenbank“ der richtige Pfad und Zugangsdaten eingetragen wurden – um das erneute Einlesen zu erzwingen betätigen Sie die Schaltfläche „**Aktualisieren**“.

Im unteren Bereich sehen Sie die Ergebnisse des Vergleichs oder des Updates, die die Standard-Datenbank und den Customizing-Bereich betreffen. Änderungen der Standard-Datenbank sind dem Hersteller, der orgAnice Software GmbH, vorbehalten – Sie können hier die Änderungen sehen, die zwischen zwei Datenbankversionen stattgefunden haben.

AddOn-Applikation: im oberen Bereich des Rahmens können Sie die Version des in der Combobox „Präfix“ ausgewählten AddOns in der Quell- und der Ziel-Datenbank ablesen. Die Versionen werden angezeigt, sobald in den Bereichen „Quell-Datenbank“ bzw. „Ziel-Datenbank“ richtige Pfad und Zugangsdaten eingetragen wurden. In der Combobox „**Präfix**“ werden alle in der Quell-Datenbank vorhandenen AddOns angezeigt, wählen Sie einen Eintrag aus, um seine Versionsnummern zu sehen. Wenn das AddOn in der Ziel-Datenbank noch nicht vorhanden ist, dann bleibt das entsprechende Anzeigefeld leer.

Betätigen Sie die Schaltfläche „...“, um die Eigenschaften eines AddOns zu bearbeiten (s. 4.7 „AddOn-Eigenschaften“, S. 15)

Im unteren Bereich sehen Sie die Ergebnisse des Vergleichs oder des Updates, die das ausgewählte AddOn betreffen. Die Interpretation der Ergebnisse ist weiter unten unter 9.2 (S. 30) beschrieben.

9.1 Menüpunkte

Datei - Vergleichen: Startet den Vergleich der Ziel- und der Quelldatenbank. Es werden die Objekte verglichen, die in dem Rahmen „Vergleichen / Update“ ausgewählt sind. Die Ergebnisse des Vergleichs werden in den beiden Baumansichten dargestellt und in einer Protokolldatei gespeichert (s. „**Optionen**“).

Datei - Update einspielen: Spielt die Erweiterungen aus der Quell- in die Ziel-Datenbank ein. Es werden die Objekte eingespielt, die in dem Rahmen „Vergleichen / Update“ ausgewählt sind. Die Ergebnisse des Vergleichs werden in den beiden Baumansichten dargestellt und in einer Protokolldatei gespeichert (s. „**Optionen**“)

Bearbeiten - UDFs bereinigen: Diese Funktion vereinheitlicht die Schreibweise der Variablennamen im Quellcode der angepassten Funktionen (=UDFs – user defined functions) in

der Quelldatenbank. Dadurch wird sichergestellt, dass keine Änderungen erkannt werden, die nur auf unterschiedlicher Groß-/ Kleinschreibung beruhen.

Fehler korrigieren: Zurzeit ohne Funktion.

Extras - Optionen: Zeigt den folgenden Dialog mit Optionen an:

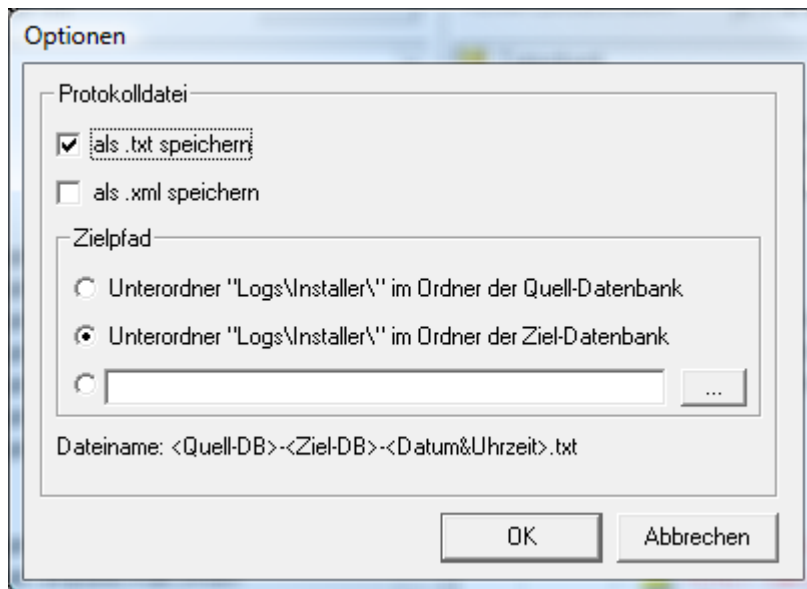


Abbildung 10: orgAnice Customizer - Dialog „Optionen“

Hier können Sie Einstellungen bzgl. der Protokollerstellung festlegen:

- Legen Sie fest, ob das Protokoll im txt-, im xml-Format oder in beiden Formaten erstellt werden soll
- Legen Sie fest, ob das Protokoll
 - in dem Unterordner „Logs/Installer“ der Quell-Datenbank
 - in dem Unterordner „Logs/Installer“ der Ziel-Datenbank
 - in einem frei wählbaren Ordnererstellt werden soll.

Extras – Aktualisieren: Liest die Versionsnummern der Quell- und Ziel-Datenbank erneut ein.

? - **Lizenz updaten:** Hier können Sie eine neue Lizenznummer für den Customizer eintragen.

? - **Info:** Zeigt die Versionsnummer des orgAnice Customizers an.

9.2 Interpretation der Vergleichsergebnisse

Die Interpretation der Ergebnisse hängt davon ab, welchen Vergleichsmodus Sie nach dem Betätigen der Schaltfläche „Vergleichen“ gewählt haben.

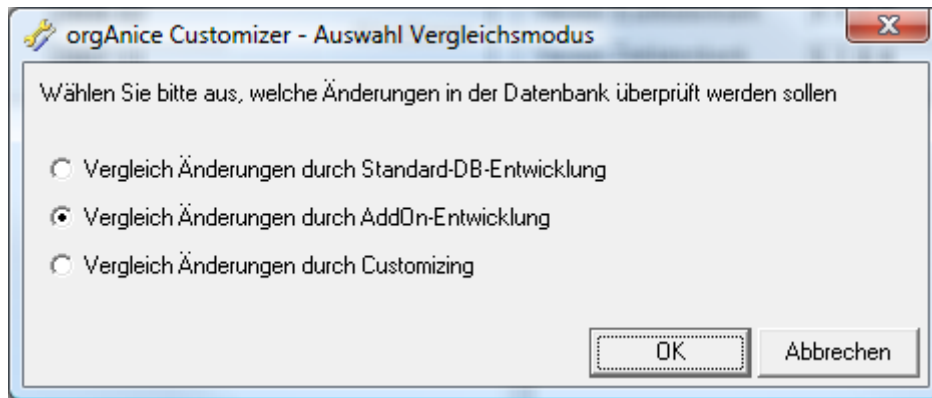


Abbildung 11: orgAnice Customizer – Auswahl des Vergleichsmodus

Es stehen drei Modi zur Verfügung:

- Vergleich Änderungen durch Standard-DB-Entwicklung: wählen Sie diesen Punkt, wenn Sie zwei unterschiedliche Versionen der Standard-Datenbank vergleichen wollen
- Vergleich Änderungen durch AddOn-Entwicklung: wählen Sie diesen Punkt, wenn Sie zwei unterschiedliche Versionen eines AddOns vergleichen wollen
- Vergleich Änderungen durch Customizing: wählen Sie diesen Punkt, wenn Sie zwei unterschiedliche Versionen einer Customizing-Entwicklung (Präfix CSTM) vergleichen wollen.

Wie wirkt sich diese Einstellung aus? Es werden in allen drei Fällen **alle** Änderungen angezeigt – der Unterschied liegt darin, welche Änderungen für den ausgewählten Fall als **erlaubt** und welche als **unerlaubt** gelten. Was erlaubt ist und was nicht, können Sie in 3 „Programmierregeln“, S. 10, nachlesen. So können Sie, wenn Sie z.B. ein AddOn entwickeln, keine neuen Objekte zur Standard-Datenbank hinzufügen – die Namen aller neuen Objekte müssen mit dem AddOn-Präfix beginnen. Solche Änderungen würden später über die Funktion „Update einspielen“ oder mit dem Installer nicht eingespielt werden.

Der orgAnice Customizer verwendet verschiedene Symbole für die Kennzeichnung der unterschiedlichen Zustände:



mit dem Informationszeichen werden Änderungen gekennzeichnet, die erlaubt sind und damit die auch in die Ziel-Datenbank eingespielt werden (z.B. einem AddOn wird neues Feld hinzugefügt, der Name des Feldes beginnt mit dem richtigen Präfix)



mit dem Ausrufezeichen werden Änderungen gekennzeichnet, die zwar innerhalb der Programmierregeln gültig sind, aber bei der ausgewählten Option nicht eingespielt werden (z. B. neue AddOn-Objekte des AddOn B beim Vergleich der Änderungen durch AddOn-Entwicklung des AddOns A). Mit dem Ausrufezeichen werden auch fehlende Objekte markiert, die allerdings für das Funktionieren des AddOn nicht notwendig sind, bspw. das Module <AddOnPräfix>_Macros.



mit dem weißen Kreuz auf rotem Hintergrund werden Änderungen gekennzeichnet, die ungültig im Sinne der Programmierregeln sind und nicht in die Ziel-Datenbank eingespielt werden (z.B. Löschungen von Tabellen, Hinzufügen von Objekten der Standard-Datenbank bei einem AddOn).

10 Auslieferung Ihrer Änderungen mit dem orgAnice Installer

10.1 Voraussetzungen

Damit Ihre Änderungen beim Kunden eingespielt werden, liefern Sie einfach eine entsprechende Datenbank aus, die alle relevanten Objekte Ihres AddOns und der Standarddatenbank enthält.

Beim Einspielen geht der orgAnice Installer von folgenden Voraussetzungen aus:

1. Die Quelldatenbank (mit dem aktuellen AddOn) befindet sich im selben Ordner wie die aufgerufene orgInstaller.exe
2. Die Datenbank trägt den Namen „**orgAniceX.Y.odb**“, wobei X.Y für die Versionsnummer von orgAnice steht:
 - **orgAnice4.0.odb** für orgAnice SQL/CRM (4.0)
 - **orgAnice5.0.odb** für orgAnice CRM 2008 (5.0)

Aus Kompatibilitätsgründen mit älteren Installer-Versionen kann die Datenbank auch OfficeSQL.odb heißen, wenn orgAnice SQL/CRM installiert ist:

3. Der Zugang zur Datenbank erfolgt über Benutzer „**Admin**“ mit leerem Passwort

Sollte eine dieser Bedingungen nicht erfüllt sein, so erscheint die Fehlermeldung, dass das Update nicht gefunden werden konnte.

Achten Sie also im Interesse Ihrer Anwender darauf, dass diese Bedingungen erfüllt sind.

Es können sich im Installationsordner auch **mehrere Datenbanken** für unterschiedliche orgAnice-Versionen befinden. Damit können Sie Installationspakete zusammenstellen, die mehrere orgAnice-Versionen unterstützen. orgAnice Installer öffnet dann je nach installierter orgAnice-Version die passende Datenbank.

Hinweise zu der Bedienung des orgAnice Installers finden Sie in im nächstem Kapitel. Der Inhalt des Kapitels ist auch als ein gesondertes Dokument erhältlich, welches Sie als AddOn-Entwickler Ihren Kunden zur Verfügung stellen können.

10.2 Installationspaket erstellen

Für die Auslieferung aller relevanten Dateien können Sie diese Dateien mit Hilfe eines Dritttools zu einer einzigen Datei zusammenpacken. Hierfür bietet sich der Einsatz von Tools, die selbstentpackende .exe-Dateien erstellen können. Wir schlagen das Tool IExpress vor, welches die folgenden Eigenschaften besitzt:

- IExpress befindet sich im Lieferumfang von Windows (2000, XP, Vista)
- es ist einfach zu bedienen
- es ist kostenlos
- die Anzeige einer Hinweisdatei ist möglich
- die Erstellung des Auslieferungspaketes wird mit Hilfe eines Assistenten schrittweise vorbereitet
- Einstellungen können für Wiederverwendung in einer Konfigurationsdatei gespeichert werden

Starten Sie IExpress über „Start – Ausführen– IExpress“:

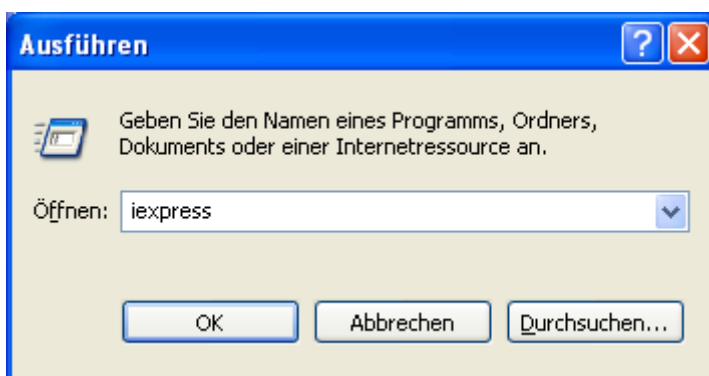


Abbildung 12: IExpress starten



Abbildung 13: IExpress Wizard

Fügen Sie dem Paket die folgenden Dateien hinzu:

- orgInstaller.exe

- die Quell-Datenbank (z.B. orgAnice4.o.odt, orgAnice5.o.odt oder mehrere)
- weitere Dateien, wie z.B. die „DB-Folder.exe“ (s. 13 „Installation von zusätzlichen Dateien mit dem Installer“, S. 46)

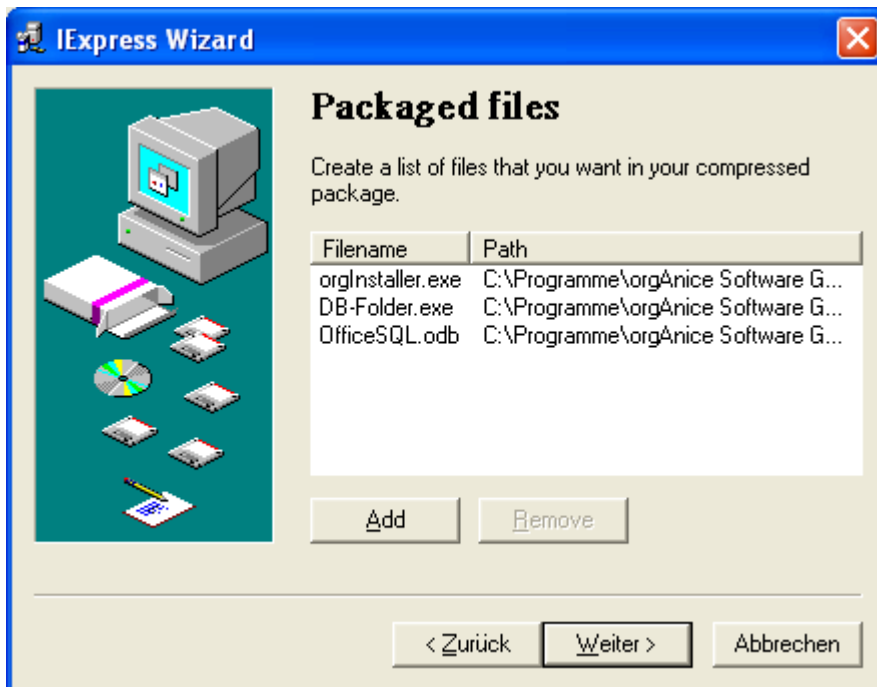


Abbildung 14: IExpress - Auswahl des Paketumfangs

Wählen Sie dann als die zu startende Datei die „orgInstaller.exe“ aus:

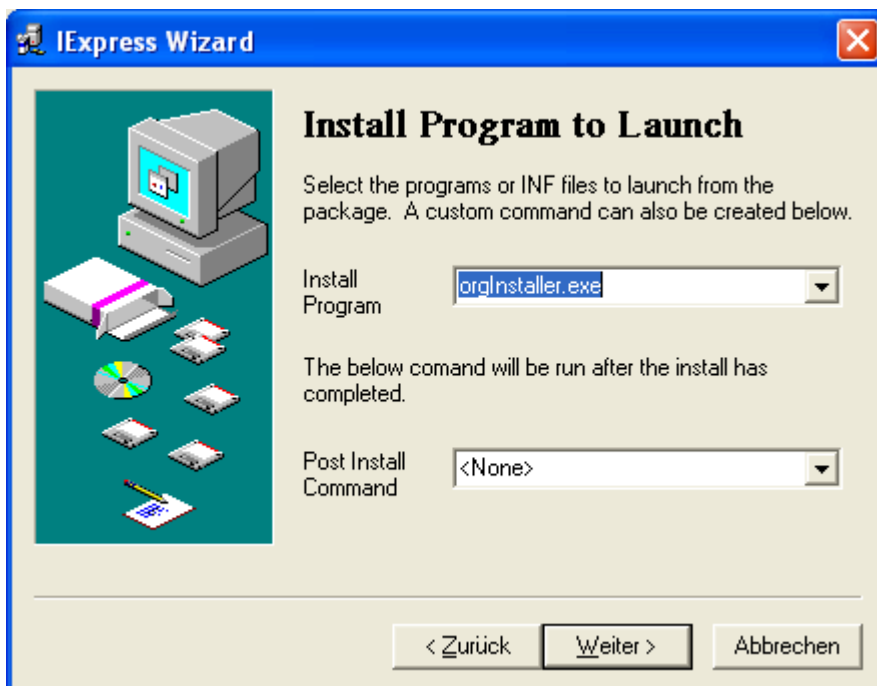


Abbildung 15: IExpress - orgInstaller.exe starten

Als Ergebnis wird eine selbstentpackende .exe-Datei erstellt, die Sie veröffentlichen (zum Download bereitstellen, Ihren Kunden schicken) können.

11 Bedienung der orgAnice Installers

11.1 Auswahl der Zieldatenbank

Nach dem Start des orgAnice Installers (orgInstaller.exe) erscheint ein Dialog, der Sie nach dem Zugang zu Ihrer Datenbank fragt, in die Sie das Update einspielen möchten.

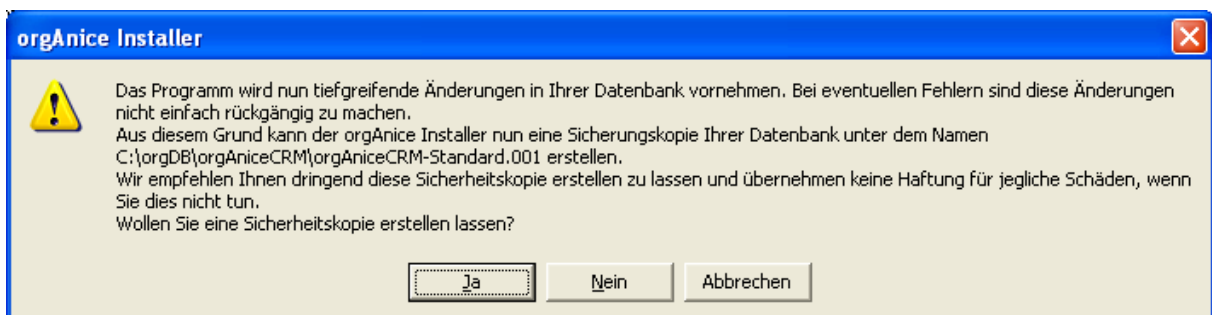
Geben Sie bitte den Pfad an und die Zugangsdaten für einen Datenbankadministrator.



11.2 Erstellung einer Sicherungskopie

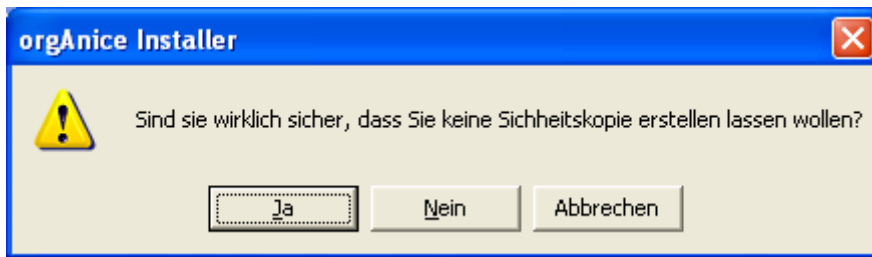
Darauf bietet Ihnen das Programm die Option an, eine Sicherheitskopie Ihrer Datenbank zu erstellen. Dies sollten Sie unbedingt tun, da nicht sichergestellt ist, dass alle Ihre Änderungen an der Datenbank nach dem Update noch verfügbar sind bzw. funktionieren.

Ebenso für den Fall, dass das Update fehlerhaft sein sollte und das Arbeiten mit Ihrer Datenbank erschwert oder unmöglich macht, empfehlen wir Ihnen dringend diese Sicherheitskopie anfertigen zu lassen.



Der Installer erstellt eine Datei im selben Pfad wie Ihre Datenbank, die den gleichen Namen trägt, allerdings mit der Dateiendung .001, bzw., falls diese bereits existiert, .002, u.s.w.

Für den Fall, dass Sie diese Frage verneinen, erscheint die erneute Abfrage...



11.3 Auswahl der einzuspielenden Änderungen

Als nächstes ermittelt der Installer die möglichen und sinnvollen Updates für Ihre Datenbank und zeigt den folgenden Dialog an:

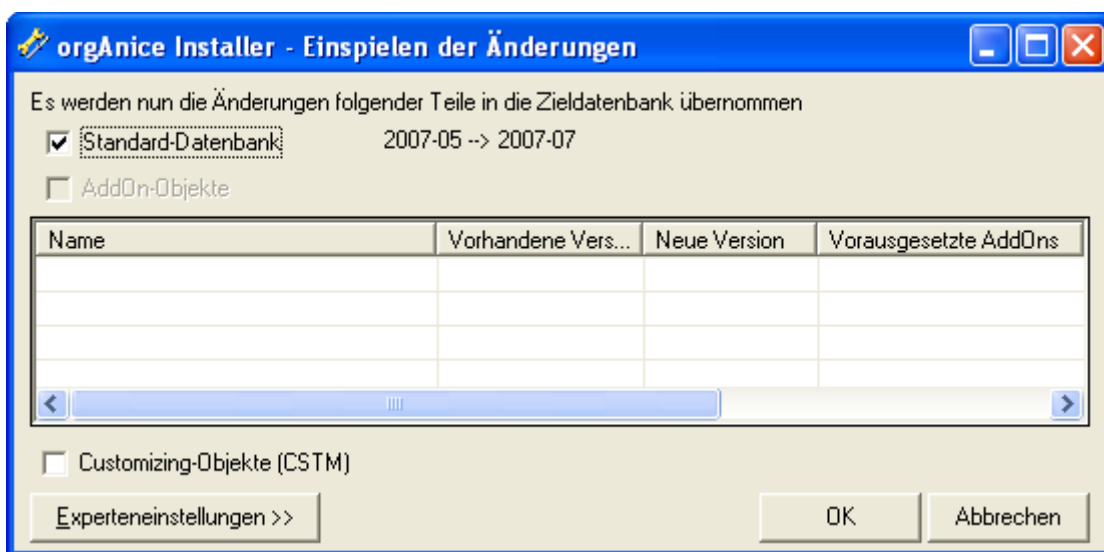


Abbildung 16: orgAnice Installer

Sie haben hier die Möglichkeit aus drei Bereichen auszuwählen.

11.3.1 Standard-Datenbank

Die orgAnice Software GmbH liefert regelmäßig Updates für die Standard- Datenbank aus.

Sollten Sie eine ältere Version der orgAnice Standard-Datenbank verwenden, so können Sie hier ggf. diese updaten. Dies hängt natürlich von der Version im gelieferten Update ab.

Der Installer vergleicht dabei die Versionsnummern in Ihrer Datenbank mit der im vorliegenden Update. Sollte letztere eine höhere Versionsnummer tragen, so schlägt er ein Update vor. Sollte Ihre Standard-Datenbank bereits auf dem neusten Stand sein, so ist diese Option nicht angekreuzt.

11.3.2 AddOn-Objekte

Sie können bei einem Updatevorgang mehrere AddOns (soweit mitgeliefert) in Ihre Datenbank einspielen oder dieses updaten. Der Installer zeigt alle in der Quell-Datenbank verfügbaren

AddOns in einer Liste an. Befinden sich darunter neuere Versionen oder noch nicht vorhandene AddOns, so schlägt der Installer diese vor und aktiviert die Option AddOn-Objekte.

Zu jedem AddOn werden die definierten Voraussetzungen angezeigt – nur wenn Voraussetzungen erfüllt sind, lässt sich das jeweilige AddOn einspielen. Sind bestimmte Voraussetzungen nicht erfüllt, dann erscheinen sie betreffenden Zellen in rot und der AddOn-Listeneintrag ist deaktiviert.



Abbildung 17: orgAnice Installer mit mehreren AddOns

Sollten die Voraussetzungen für ein AddOn in der Ziel-Datenbank nicht erfüllt, aber in der Quell-Datenbank in den passenden Versionen enthalten sein, dann erkennt der Installer die Abhängigkeiten selbständig und setzt die entsprechenden Optionen automatisch.

Beispiel: das AddOn A ist die Voraussetzung das AddOn B. In der Zieldatenbank sind weder AddOn A noch AddOn B vorhanden. Wenn in der Quell-Datenbank sowohl AddOn A als auch AddOn B mitgeliefert werden, dann lässt der Installer die Installation des AddOns B zu und installiert automatisch AddOn A mit:

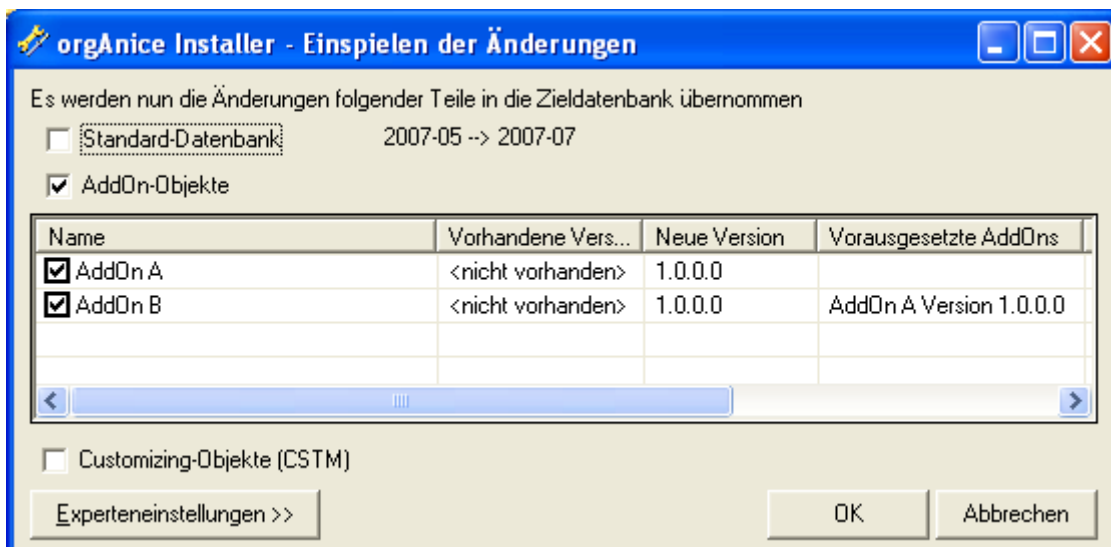


Abbildung 18: orgAnice Installer mit abhängigen AddOns

Die einzelne Installation des AddOns B ohne AddOn A wird vom Installer nicht zugelassen (sobald der Benutzer AddOn A deaktiviert, wird automatisch auch AddOn B deaktiviert)

11.3.3 Customizing-Objekte

Diese Option dient dem Übertragen von eigenen Anpassungen (Erweiterungen mit dem Präfix „CSTM“). Die Customizing-Objekte verfügen über keine Versionierung.

11.3.4 Experteneinstellungen

Mit Hilfe der Experteneinstellungen können Sie weitere Einstellungen vornehmen.

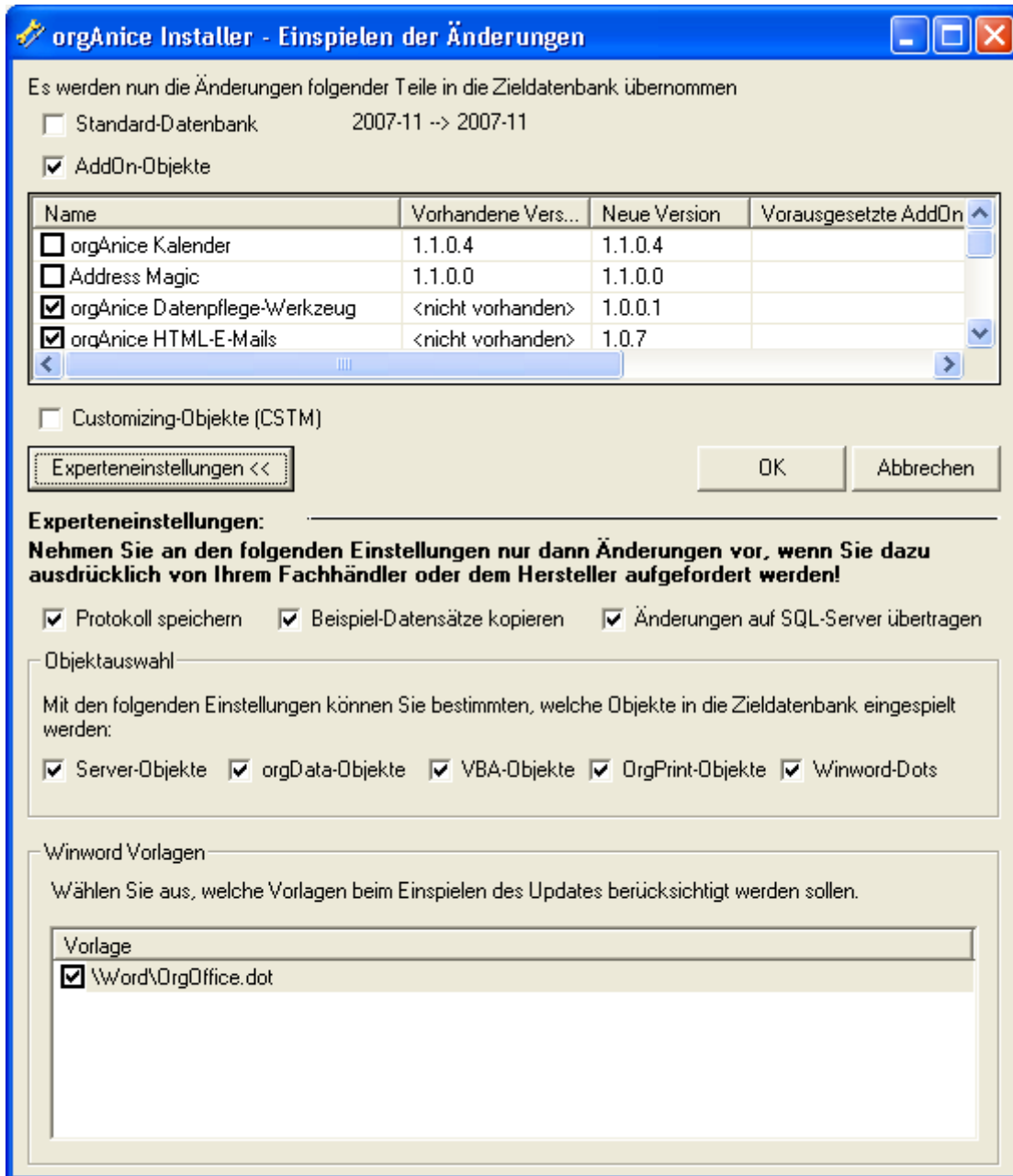


Abbildung 19: orgAnice Installer - Experteneinstellungen

11.3.4.1 Protokoll speichern

Der orgAnice Installer erstellt ein Protokoll, in welchem der Verlauf des Updates dokumentiert wird. Das Protokoll wird dem Unterordner Logs\Installer des Ziel-Datenbankordners erstellt und

trägt den Namen "<Ziel-DB> yyyy-MM-dd hh-mm-ss.txt". Deaktivieren Sie die Option „Protokoll speichern“, wenn kein Protokoll erstellt werden soll.

11.3.4.2 Beispiel-Datensätze kopieren

Bestimmen Sie, ob Datensätze aus der Quell-Datenbank in die Ziel-Datenbank kopiert werden sollen. Die Datensätze werden nur dann übertragen, wenn in der Zieltabelle noch keine Datensätze enthalten sind. Details: 15 Übertragen von Datensätzen, S. 50.

11.3.4.3 Änderungen auf den SQL-Server übertragen

Beim Einsatz des MS SQL Servers kann die Übertragung der Datenbankstrukturänderungen auf den SQL-Server vom Installer automatisch gestartet. Deaktivieren Sie die Option, falls Sie die Übertragung der Änderungen manuell, z.B. aus der Oberfläche von orgAnice Data starten möchten. Das kann z.B. dann sinnvoll sein, wenn Sie nach dem Einspielen der Änderungen weitere Änderungen an der Datenbankstruktur vornehmen und erst im Nachhinein alle Änderungen gemeinsam übertragen möchten.

11.3.4.4 Objektauswahl und Winword-Vorlagen

Mit Hilfe der Checkboxes „Server-Objekte“, „orgData-Objekte“, „VBA-Objekte“, „OrgPrint-Objekte“ und „Winword-Dots“ sowie der Liste der Word-Vorlagen können Sie den Umfang der einzuspielenden Änderungen einschränken. Deaktivieren Sie die Checkboxes, wenn die Änderungen an den entsprechenden Objekten nicht eingespielt werden sollen.

Durch das Deaktivieren von „orgData-Objekten“ können Sie bspw. verhindern, dass die Änderungen an Tabellenlayouts eingespielt werden.

Achtung: auch beim Ausschließen bestimmter Objekte wird die Versionsnummer des AddOns hochgesetzt – es ist im Nachhinein nicht mehr zu erkennen, ob das AddOn vollständig oder nicht eingespielt wurde. Unter Umständen wird das AddOn auch nicht richtig funktionieren, wenn nicht alle Bestandteile vorhanden sind.

Verwenden Sie die Option nur, wenn das Einspielen von bestimmten – möglicherweise nicht notwendigen – Objekttypen Probleme bereitet und Sie trotzdem das AddOn installieren möchten.

11.4 Einspielen des Updates

Nach Bestätigung dieses Dialoges beginnt das Einspielen des AddOns und der folgende Dialog erscheint:

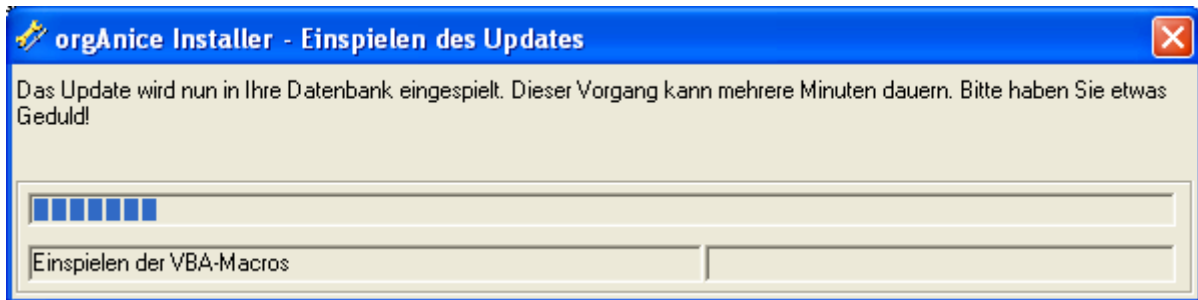


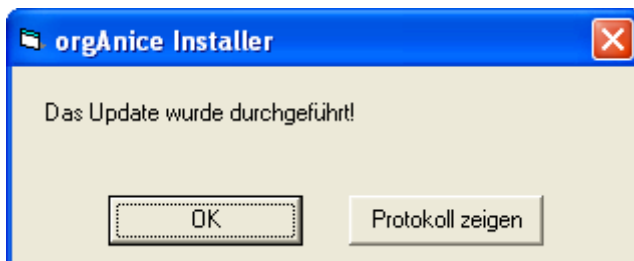
Abbildung 20: orgAnice Installer - Einspielen des Updates

Nach dem Einspielen der neuen VBA-Makros und nachdem das gesamte Update eingespielt ist, kann der Hersteller eines AddOns eigene Dialoge anzeigen lassen. Hierzu lesen Sie bitte die entsprechende Dokumentation.

Der Vorgang kann je nach Umfang des AddOns einige Minuten dauern, eine längere Dauer ist also kein Grund zur Beunruhigung.

11.5 Fertigstellung

Nach dem erfolgreichen Einspielen der Änderungen erscheint der folgende Dialog:



Mit Hilfe der Schaltfläche „Protokoll zeigen“ können Sie das während des Updates erstellte Protokoll anzeigen.

Überprüfen Sie nach Beenden des Installationsvorgangs bitte sofort Ihre Datenbank.

11.6 Sicherungskopien

Der orgAnice Installer legt bei Bedarf folgende Unterverzeichnisse unterhalb Ihres Datenbankverzeichnis an:

\Icons\

Darin werden Sie alle Icons und Abbildungslisten finden, die beim Einspielen des AddOns verwendet wurden.

\Modules\

Hier finden Sie alle VBA-Module, die der Installer neu eingespielt hat, sowie eine Kopie aller beim Update überschriebenen Module. Hier können sie also notfalls Ihre eigenen Änderungen, die überschrieben wurden, wiederfinden.

Außerdem sichert der Installer überschriebene Tabellenlayouts und Arbeitsbereiche innerhalb der Datenbank, indem er diese nach dem Schema

000<Bisheriger Name> , bzw., falls schon eine solches existiert,

001<Bisheriger Name>, u.s.w.

umbenennt.

12 Steuerung des Installers mit Kommandozeilenparametern

Der orgAnice Installer kann über Kommandozeilenparameter gesteuert werden, um bestimmte Einstellungen beim Anzeigen der Dialoge voreinzustellen oder das automatisierte Einspielen von Updates ohne Benutzereingriff zu ermöglichen. Es stehen die folgenden Parameter zur Verfügung, die alle Möglichkeiten der Oberfläche abbilden:

Parameter	Bedeutung
Database=<Pfad>	Pfad der Zieldatenbank
User=<Username>	Name des Benutzers mit Admin-Rechten
Password=<Password>	Passwort des o.g. Benutzers
InstallStandard	Objekte der Standard-Datenbank sollen upgedatet werden
InstallAddOn=<AddOnPräfix>	Präfix des zu installierenden AddOns
InstallCustom	Customizing (CSTM)-Erweiterungen sollen upgedatet werden
NoOrgDbServer	OrgDbServer-Objekte sollen nicht eingespielt werden
NoOrgData	OrgData-Objekte sollen nicht eingespielt werden
NoOrgVBA	VBA-Objekte sollen nicht eingespielt werden
NoOrgPrint	OrgPrint-Objekte sollen nicht eingespielt werden
NoWinwordTemplates	Winword-Vorlagen sollen nicht eingespielt werden
NoLog	keine Protokollerstellung
NoDialog	keine Anzeige von Benutzerdialogen, automatischer Ablauf
NoBackup	keine Erstellung von Sicherungskopieren
Silent	keine MsgBoxen, Fehlermeldungen nur im Protokoll
EnforceUpdate	Bestimmt, dass jedes Update installiert wird, egal, ob in der Zieldatenbank die Version gleich oder neuer ist. Das gilt für die Standard-DB und AddOns.
DoAfter=<Programmaufruf>	Name des Programms, welches nach der Durchführung des Updates aufgerufen werden soll. Die Eigenschaften „Ausführen nach Installation“ aller AddOns werden ignoriert. Details s. 14.2 „Programmaufruf über Kommandozeilenparameter“, S. 48

Tabelle 1: Kommandozeilenparameter vom orgAnice Installer

Die einzelnen Parameter müssen durch Leerzeichen und Schrägstrich („/“) getrennt werden.

12.1 Beispielaufufe

Bsp.: OrgInstaller.exe /Database=C:\orgdb\orgAniceCRM.odt /User=ADMIN
/Password=geheim /InstallStandard /InstallAddOn=XYAB /NoDialog

- die Datenbank C:\orgdb\orgAniceCRM.odt wird mit dem Benutzer ADMIN, Passwort „geheim“ geöffnet
- es werden die Änderungen der Standard-Datenbank und das AddOn XYAB automatisch eingespielt

Bsp.: OrgInstaller.exe /Database=C:\orgdb\orgAniceCRM.odt /User=ADMIN
/Password=geheim /InstallAddOn=XYAB /EnforceUpdate / NoBackup /NoDialog
/DoAfter=start.exe

- die Datenbank C:\orgdb\orgAniceCRM.odt wird mit dem Benutzer ADMIN, Passwort „geheim“ geöffnet
- es wird das AddOn XYAB unabhängig von der vorhandenen Version eingespielt (es wird also möglicherweise eine neuere Version durch eine ältere ersetzt)
- es wird keine Sicherungskopie erstellt
- nach dem Einspielen wird das Programm start.exe ausgeführt

Bsp.: OrgInstaller.exe /Database=C:\orgdb\orgAniceCRM.odt /User=ADMIN
/Password=geheim

- die Datenbank C:\orgdb\orgAniceCRM.odt wird mit dem Benutzer ADMIN, Passwort „geheim“ geöffnet, der Benutzer sieht sofort den Dialog „Einspielen der Änderungen“, muss die Auswahl der Änderungen jedoch eigenständig vornehmen.

12.2 Parameterabhängigkeiten

Zwischen den Parametern bestehen Abhängigkeiten, die in der folgenden Tabelle dargestellt sind:

Parameter	mit Parametern	ohne Parameter
Database	<ul style="list-style-type: none"> • NoDialog: die Dialoge „Öffnen der Zieldatenbank“, „Sicherungskopie“ und „Einspielen der Änderungen“ werden nicht angezeigt. • Silent: wie NoDialog, es werden zusätzlich weder MsgBoxen (z.B. Schlussmeldung) noch Fehlermeldungen angezeigt 	<ul style="list-style-type: none"> • User: Name des Datenbankbenutzer ist Pflicht, Fehlermeldung bei fehlender Angabe

NoDialog	<ul style="list-style-type: none"> • NoBackup: keine Sicherungskopie. 	<ul style="list-style-type: none"> • EnforceUpdate: wenn die Version der Datenbank oder des AddOns nicht neuer ist, dann muss bei fehlendem EnforceUpdate der Benutzer entscheiden, ob das Update eingespielt werden soll → Benutzereingriff notwendig • Database und User: Datenbankpfad und Benutzername sind Pflicht, Fehlermeldung bei fehlenden Angaben
Silent	<ul style="list-style-type: none"> • EnforceUpdate: auch ältere und gleiche Versionen werden installiert • NoBackup : keine Sicherungskopie 	<ul style="list-style-type: none"> • EnforceUpdate: wenn die Version der Datenbank oder des AddOns nicht neuer ist, dann keine Installation → kein Benutzereingriff möglich • Database und User: Datenbankpfad und Benutzername sind Pflicht, Fehlermeldung bei fehlenden Angaben
NoDialog / Silent		<p>Mindestens einer der Parameter</p> <ul style="list-style-type: none"> • InstallStandard • InstallAddOn • InstallCustom <p>muss angegeben sein.</p>

Tabelle 2: Abhängigkeiten der Kommandozeilenparameter im orgAnice Installer

12.3 ExitCodes

Der Installer wurde mit Rückgabewerten ausgestattet, welche von der aufrufenden Komponente ausgewertet werden können.

ExitCode	Ergebnis
0	Erfolgreich
1	Erfolgreich, mit Fehlern
9	Nicht erfolgreich

13 Installation von zusätzlichen Dateien mit dem Installer

Mit Hilfe des orgAnice Installers ist es möglich, mitgelieferte Dateien in den Ordner der Ziel-Datenbank zu kopieren. Wenn Ihr AddOn z.B. eigene Vorlagendateien oder externe Konfigurationsdateien beinhaltet, so können Sie auf diese Weise diese Dateien ausliefern; sie werden automatisch in den Ordner der Ziel-Datenbank kopiert.

13.1 Unterordner „DB-Folder“

Wenn im Ordner der orgInstaller.exe ein Unterordner namens "DB-Folder" existiert, dann wird der Inhalt dieses Ordners in den Ordner der Ziel-Datenbank kopiert. Der Ordner „DB-Folder“ kann weitere Unterordner beinhalten, die Unterordnerstruktur wird in dem Ziel-Datenbank-Ordner abgebildet (nicht vorhandene Ordner werden erstellt).

Dateien, die bereits existieren, werden nicht überschrieben, damit bspw. Benutzeranpassungen an den Vorlagen nicht überschrieben werden. Liefern Sie Dateien mit, die von Benutzer nicht verändert werden können und die upgedatet werden sollen, dann können Sie die Namen dieser Dateien als eine Eigenschaft Ihres AddOns bestimmen. Die Eigenschaft bearbeiten Sie mit dem orgAnice Customizer, Details s. 4.7.3.1 „Dateien überschreiben“.

13.2 Selbstentpackende Datei „DB-Folder.exe“

Wenn der Unterordner „DB-Folder“ nicht existiert, dann sucht der Installer nach der Datei „DB-Folder.exe“ im eigenen Ordner. Wenn diese Datei existiert, dann wird sie gestartet. Sie sollte eine selbstentpackende-EXE sein, deren Inhalt sich automatisch in den Unterordner „DB-Folder“ extrahiert. Damit ist es möglich ein Installations-Paket mit IExpress.exe zu erstellen, welches die „orgInstaller.exe“, die Datenbankdatei (z.B. „orgAnice4.o.odb“) und die „DB-Folder.exe“ enthält (IExpress.exe unterstützt keine Unterordner). Nach der Extraktion der EXE-Datei wird der Inhalt des Ordners, wie oben beschrieben, in den Ziel-Datenbankordner kopiert.

13.3 Beispiel

Ihr AddOn liefert zusätzliche Aktivitäten-Vorlagen und Vorlagen für Urlaubsplanung mit. Erstellen Sie im Ordner des Installers einen neuen Unterordner „DB-Folder“ und erstellen Sie dort zwei Unterordner „Vorlagen für Aktivitäten“ und „Vorlagen für Urlaubsplanung“.

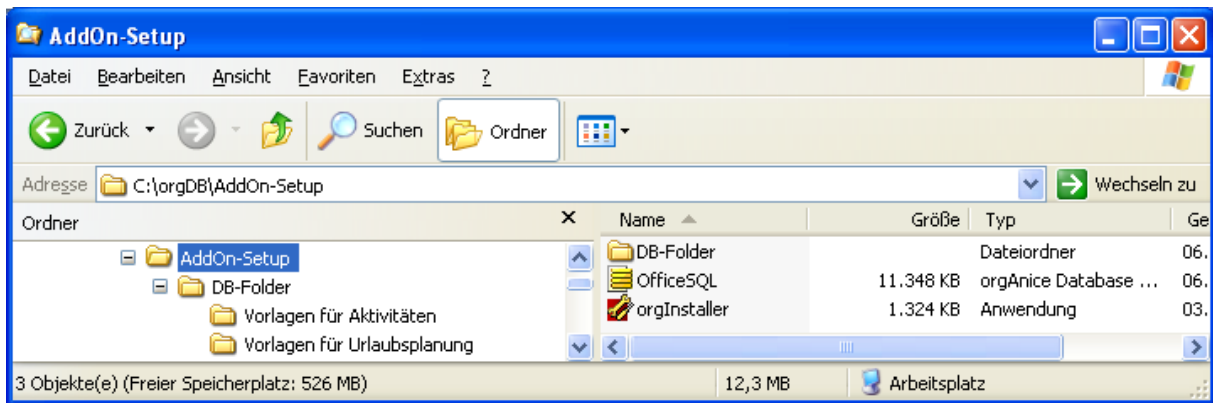


Abbildung 21: Zusätzliche Dateien installieren

Fügen Sie in diese Ordner die gewünschten Vorlagen hinzu. Beim Einspielen in die Ziel-Datenbank wird der Inhalt von „Vorlagen für Aktivitäten“ in der (normalerweise) bestehenden Ordner „Vorlagen für Aktivitäten“ kopiert, der Ordner „Vorlagen für Urlaubsplanung“ wird neu erstellt und sein Inhalt kopiert.

14 Aufruf von weiteren Programmen

Der orgAnice Installer verfügt zwar über vielfältige Funktionalitäten, mit denen sich Erweiterungen in eine Ziel-Datenbank einspielen lassen, trotzdem kann es AddOns geben, für die diese Möglichkeiten nicht ausreichen. In solchen Fällen können Sie ein eigenes Installationsprogramm schreiben, welches den Installer allerdings nicht zu ersetzen braucht, sondern nur zu erweitern. orgAnice Installer ist nämlich in der Lage, nach dem Durchführen der Installation ein weiteres Programm aufzurufen.

14.1 Programmaufruf über AddOn-Eigenschaften

Den Namen des zu startenden Programms können Sie als Eigenschaft Ihres AddOns definieren. Die Eigenschaft bearbeiten Sie mit dem orgAnice Customizer, s. 4.7.3.2 „Ausführen nach Installation“, S. 21.

Die zu startende Anwendung muss sich im Ordner des orgAnice Installers befinden.

14.2 Programmaufruf über Kommandozeilenparameter

Der Aufruf über die AddOn-Eigenschaft sollte dann verwendet werden, wenn Sie ein bestimmtes Installationsprogramm immer mit Ihrem AddOn ausliefern möchten. Es steht Ihnen aber auch die Möglichkeit zur Verfügung, dem Installer über einen Kommandozeilenparameter das zu startende Programm mitzuteilen. Damit können Sie für dasselbe AddOn unterschiedliche Installationsroutinen ausliefern. Diese Möglichkeit bietet sich dann an, wenn:

- Sie nur in bestimmten Fällen (z.B. für eine bestimmte Kundengruppe) ein zusätzliches Programm starten möchten,
- Sie unterschiedliche Installationsprogramme für unterschiedliche Zielgruppen verwenden möchten,
- Sie mehrere AddOns in einem Installer-Paket ausliefern und ein gemeinsames Installationsprogramm aufrufen möchten.

Der Kommandozeilenparameter für den Aufruf der Anwendung lautet:

DoAfter=<Programmname> (S. 12. „Steuerung des Installers mit Kommandozeilenparametern“, S. 43)

Bsp.: OrgInstaller.exe /Database=C:\orgdb\orgAniceCRM.odt /User=ADMIN
/Password=geheim /InstallAddOn=XYAB /EnforceUpdate / NoBackup /NoDialog
/DoAfter=start.exe

Hinweis: Die Verwendung des Kommandozeilenparameters setzt die eventuell vorhandenen AddOn-Eigenschaften (s. 14.1 „Programmaufruf über AddOn-Eigenschaften“) außer Kraft. D.h.

es wird nur das Programm aufgerufen, das über den Kommandozeilenparameter übergeben wurde, die Aufrufe aus den AddOn-Eigenschaften werden nicht ausgeführt.

14.3 Übergabe von Parametern

An den Aufruf (unabhängig davon, ob er über die AddOn-Eigenschaft oder Kommandozeilenparameter erfolgt ist) werden alle aktuell geltenden Einstellungen des Installers als Kommandozeilenparameter angehängt, sodass das aufgerufene Programm entsprechend reagieren kann. Es werden u.a. der Pfad und die Zugangsdaten zur Ziel-Datenbank und die Auswahl der einzuspielenden Änderungen übergeben. Die vollständige Liste der möglichen Parameter sieht folgendermaßen aus:

```
/Database=<Zieldatenbankpfad> /User=<Zieldatenbankbenutzer>  
/Password=<Zieldatenbankkennwort> /InstallStandard  
/InstallAddOn=<ADDON> /InstallCustom /NoOrgDbServer /NoOrgData  
/NoOrgVba /NoOrgPrint /NoWinwordTemplates /NoLog /NoDialog /NoBackup  
/Silent /EnforceUpdate
```

Die Parameter entsprechen denen, die für den Installer definiert sind, s. 12. „Steuerung des Installers mit Kommandozeilenparametern“, S. 43.

Die Auswahl der übergebenen Parameter ergibt sich aus den durch den Benutzer getroffenen Einstellungen und aus den Parametern, mit denen der Installer selbst aufgerufen wurde.

15 Übertragen von Datensätzen

Beim Einspielen von AddOns haben Sie die Möglichkeit Beispiel und/oder Konfigurationsdatensätze zu übertragen. Die Übertragung von Datensätze aus der Quell- in die Zieltabelle findet nur dann statt, wenn die Zieltabelle keine Datensätze beinhaltet, also:

- entweder beim ersten Einspielen des AddOns oder
- wenn in einer neuen Version des AddOns eine neue Tabelle hinzukommt oder
- wenn der Benutzer noch keine Datensätze in einer bestehenden Tabelle angelegt hat.

Der Benutzer hat immer im Installer über die „Experteneinstellungen“ immer die Möglichkeit die Option „Beispiel-Datensätze kopieren“ zu deaktivieren.

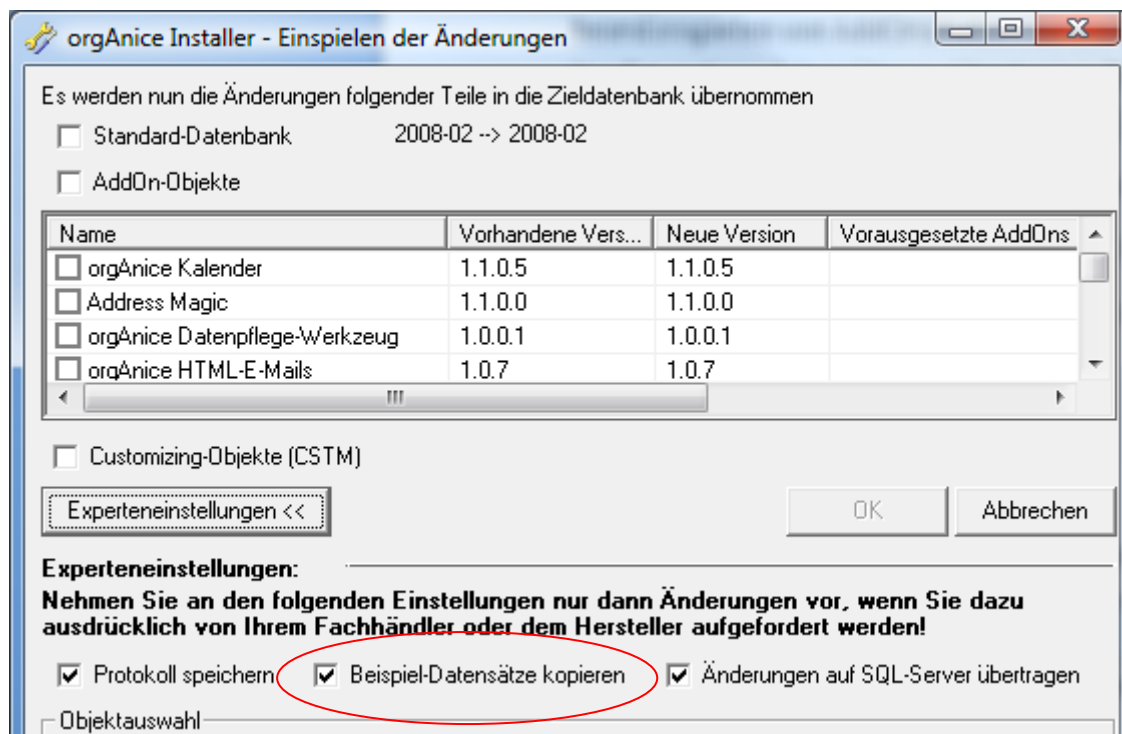


Abbildung 22: Beispiel-Datensätze kopieren

16 Anhang A – Schichtenmodell der Zugriffe

In jeder Datenbank, die die Regeln für AddOns und Customizing einhält, kann es die drei Bereiche Standard-Datenbank, AddOn und Customizing geben, die die in der unteren Abbildung angezeigten Verflechtungen haben können.

Ein AddOn darf auf eigene Objekte, auf Objekte der Standarddatenbank oder auf Objekte der vorausgesetzten AddOns zugreifen (diese verwenden). Es darf die eigene Objekte und die Objekte der Standarddatenbank (m.E.) verändern. Es darf Custom-Objekte weder verwenden, noch verändern.

Der Bereich Customizing darf sowohl auf eigene, auf Objekte von AddOns als auch auf Objekte der Standarddatenbank zugreifen (diese verwenden) und diese ggf. verändern (m.E.)

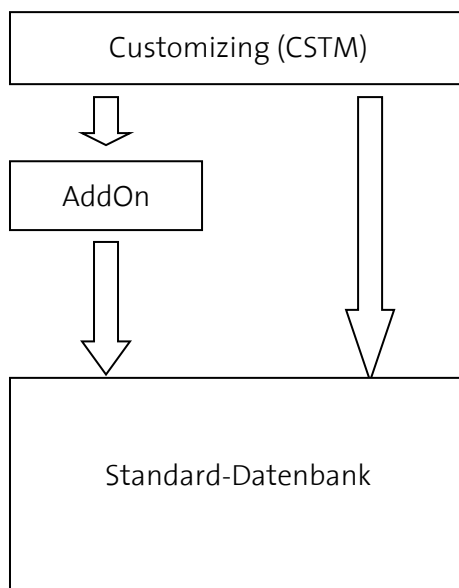


Abbildung 23: Bereiche einer orgAnice-Datenbank

17 Anhang B – Objektbezogene Funktion des Customizers/Installers

Die unten stehenden Tabellen geben an, ob ein neues Objekt oder eine Änderung vom orgAnice Customizer übertragen wird oder nicht (X – wird übertragen, O – wird nicht übertragen. Ein Sternchen * deutet darauf hin, dass die Übertragung nicht immer, sondern nur unter bestimmten Voraussetzungen stattfindet.)

17.1 Objekte des OrgDbServers

1. Ebene	2. Ebene	3. Ebene	Neu	Änderung / Update	Kommentar / Ausnahmen / Einschränkungen
Parser					
	Angepasste Funktionen (UDFs = user defined functions)		X	X	Werden immer übertragen
	persistente Variablen	einzelner Wert	X	O	Nur neue, keine Änderung
	persistente Variablen, Lookup-Tabellen	Lookup-Werte	X	X*	Bei geänderten Lookup-Tabellen werden neue Werte hinzugefügt, bestehende Werte bleiben erhalten.
Benutzer			O	O	Werden gar nicht eingespielt, da Benutzer nicht Standard sein können
	Benutzer-Eigenschaften		O	O	Benutzereigenschaften werden gar nicht eingespielt, da sie nicht Standard sein können und gleiche Eigenschaften für alle Benutzer keinen Sinn machen.
Berechtigungen			X		
	berechtigte Benutzer		X*	O	Sofern Benutzer bereits in der Datenbank vorhanden, anhand des Benutzernamens
Tabellen			X		
	Tabellen-Titel			X	Wird immer übertragen
	Felder		X	X*	Bei Änderung: Nur Feldverlängerungen oder setzen auf StringLength o (=variable Länge)

1. Ebene	2. Ebene	3. Ebene	Neu	Änderung / Update	Kommentar / Ausnahmen / Einschränkungen
		NoInput NoUpdate		X	Werden immer übernommen
	Permissions			O	Nur bei Neuanlage der Tabelle
	Tabellen- Eigenschaften		X	O	Nur neue, keine Änderung, da diese für Konfigurationseinstellungen verwendet werden sollten und damit nicht überschrieben werden dürfen.
	Indizes		X	O (!)	Keine Änderung von Indizes!
	Markierungen		O	O	Macht keinen Sinn, da höchstwahrscheinlich andere Datenbankinhalte vorliegen
	Markierungs- Ausdrücke	Ausdruck	X	X	Vorgegebene Markierungsausdrücke werden IMMER geändert. Der Benutzer kann ja leicht eigene erstellen.
		markierende Tabelle		X	Die markierende Tabelle wird ebenfalls immer überschrieben, falls sie geändert wird.
	Summierungs- Ausdrücke		X		Wie Markierungsausdrücke
		Summierungs- Gruppierungs- Ausdruck Bedingung		X	Änderungen werden IMMER überschrieben, da der Benutzer leicht eigene Abfragen erstellen kann und dies bei Strukturänderungen wichtig sein kann.
Relationen			X	O	Nur, wenn nicht oberhalb einer Standard-Tabelle. Keine Änderung durch Installer
Binäre Objekte			X	O	Nur neue Binäre Objekte, deren Name nicht mit „Org“ beginnt, werden eingespielt.
DB- Eigenschaften			X	X	DB-Eigenschaften werden immer eingespielt (ausser Standard-DB, da nur Office_Version).

Tabelle 3: Objekte des OrgDbServers

17.2 Objekte von OrgData

1. Ebene	2. Ebene	3. Ebene	Neu	Änderung / Update	Kommentar / Ausnahmen / Einschränkungen
Arbeitsbereichsgruppen			X		Neue Arbeitsbereichsgruppen werden als vorletzte Gruppe (vor Konfiguration) eingespielt.
	Arbeitsbereiche		X		In dieser Gruppe neu hinzugekommene Arbeitsbereiche werden ebenfalls übertragen. Die Reihenfolge der Arbeitsbereiche kann nicht garantiert werden, wird jedoch bei neuen Gruppen eingehalten.
Arbeitsbereiche			X	X *	Änderungen eines Arbeitsbereichs werden nur dann eingespielt, wenn Sie a) <i>Rahmen- oder Spaltenzahl verändert haben</i> oder b) <i>Mindestens ein Tabellenlayout durch ein anderes ausgetauscht haben, das einer anderen Tabelle zugeordnet ist.</i>
Tabellenlayouts			X	X	Bei Änderung wird nach Sicherung des bestehenden das neue eingespielt.
ImageLists			X	O	Nur neu, keine Änderungen
Standard-Schriften			O	O	Werden gar nicht eingespielt
VBA-Module	<i>(außer UserMacros und Macros)</i>		X	X	Werden immer überschrieben, außer den Modulen <i>UserMacros</i> und <i>Macros</i> . Fehlende AddOn-Module werden aus der Ziel-Datenbank gelöscht! Der AddOn-Entwickler ist für die Einhaltung der Schnittstelle verantwortlich.
	UserMacros	Prozeduren	X	X	Prozeduren (nach Namenskonvention) werden neu eingespielt und auch bei Änderungen aktualisiert.
	Macros	Prozeduren	X	X	Prozeduren (nach Namenskonvention) werden neu eingespielt und auch bei Änderungen aktualisiert. Änderungen im Modul Macros gehören immer nur zu Standard-Datenbank und sind der orgAnice Software GmbH vorbehalten!

Tabelle 4: Objekte von orgData

17.3 Allgemeine Hinweise zu Änderungen von Tabellenlayouts und Arbeitsbereichen

Zu überschreibende Arbeitsbereiche oder Tabellenlayouts werden vom Installer / Customizer zuvor innerhalb der Datenbank gesichert. Ihrem bisherigen Namen wird dabei ein Unterstrich und eine laufende 3-stellige Ziffernfolge vorangestellt, die automatisch bei mehreren vorhandenen Sicherungen hochgezählt wird.

Beispiel:

Soll das Tabellenlayout *AdressenMini* überschrieben werden, so kopiert der Installer dies zu einem Layout mit dem Namen *_001_AdressenMini* an.

Beachten Sie, dass im Rahmen einer AddOn-Entwicklung die Tabellenlayouts der Standard-Datenbank nicht geändert werden dürfen – diesen Änderungen werden nicht eingespielt.

Möchten Sie ein Standard-Tabellenlayout um Steuerelemente erweitern, so müssen Sie eine Kopie des Tabellenlayouts (am besten mit dem orgAnice Designer) erstellen und diese Kopie mit dem Add-On-Präfix versehen.

17.4 Allgemeine Hinweise zu Änderungen von Makro-Modulen

Zu überschreibende Makromodule (Formulare, Module oder Klassenmodule) oder – falls geändert – das Modul UserMacros werden zuvor in den Ordner

`\Modules\<yyyyMMddHHmm>`

unterhalb des Zieldatenbank-Ordners exportiert, wobei `<yyyyMMddHHmm>` für die aktuelle Zeit steht.

17.5 Objekte von OrgPrint

1. Ebene	2. Ebene	3. Ebene	Neu	Änderung / Update	Kommentar / Ausnahmen / Einschränkungen
Reports			X	O	Aus technischen Gründen können nur neue Reports oder Etiketten übernommen werden
Labels			X	O	dto.

Tabelle 5: Objekte von OrgPrint

17.6 Word-Makro-Vorlagen

1. Ebene	2. Ebene	3. Ebene	Neu	Änderung / Update	Kommentar / Ausnahmen / Einschränkungen
VBA-Module			X	X	Werden immer überschrieben. Fehlende AddOn-Module werden aus der Ziel-Datenbank gelöscht! Der AddOn-Entwickler ist für die Einhaltung der Schnittstelle verantwortlich.

Tabelle 6: Word-Makro-Vorlagen

18 Anhang C – VBA Standardprozeduren

Ihr Addon sollte **immer** ein Modul mit Namen <AddOn-Präfix>_Macros enthalten. In der aktuellen Standard-Datenbank finden Sie das Modul CSTM_Makros, welches Sie ein Vorlage für Ihr eigenen AddOn-Macros-Modul verwenden können. Es beinhaltet viele der unten beschriebenen Standard-Prozeduren.

In dem Modul <AddOn-Präfix>_Macros sollten sich folgende Prozeduren befinden, deren Parameter nicht geändert werden dürfen:

18.1 Auto-Makros

Diese Makros werden automatisch beim Auftreten von bestimmten Zuständen in orgAnice Data aufgerufen.

18.1.1 <AddOn-Präfix>_AutoOpen()

Wird aus den Standardmakros am Ende der Prozedur AutoOpen, nach dem Aufruf von AutoOpenUser aufgerufen.

18.1.2 <AddOn-Präfix>_AutoClose()

Wird aus den Standardmakros am Ende der Prozedur AutoClose, nach dem Aufruf von AutoCloseUser aufgerufen.

18.1.3 <AddOn-Präfix>_CloseRequest ()

Wird aus den Standardmakros am Ende der Prozedur CloseRequest, nach dem Aufruf von CloseRequestUser aufgerufen.

18.1.4 <AddOn-Präfix>_InitGlobalObjects ()

Wird aus den Standardmakros am Ende der Prozedur InitGlobalObjects, nach dem Aufruf von InitGlobalObjectsUser aufgerufen.

18.1.5 <AddOn-Präfix>_ClearGlobalObjects ()

Wird aus den Standardmakros am Ende der Prozedur ClearGlobalObjects, nach dem Aufruf von ClearGlobalObjectsUser aufgerufen.

18.2 Installer-Makros

Diese Makros werden vom orgAnice-Installer aufgerufen und können dazu verwendet werden, seine Funktionalität zu erweitern.

18.2.1 <AddOn-Präfix>_BeforeUpdate()

Wird bei der Installation Ihres AddOns vom Customizer / Installer **nach Übertragen der Makros vor dem Übertragen der Datenbankstruktur** aufgerufen.

Sie können hier also während der Installation Vorbereitungen Treffen, so z.B. einzelne Objekte löschen, damit sie vollständig eingespielt werden.

18.2.2 <AddOn-Präfix>_Update()

Wird bei der Installation Ihres AddOns vom Customizer / Installer **nach dem Einspielen aller Objekte** aufgerufen.

Sie können hier z.B. Datenimporte vornehmen o.Ä. vornehmen oder eine Haupt-Konfiguration Ihres AddOns anstoßen.

18.3 Konfigurations-Makros

Diese Makros werden für die Konfiguration von AddOns verwendet.

18.3.1 <AddOn-Präfix>_DisplaySettings()

Wird aufgerufen, wenn der Administrator im Dialog Datenbankeinstellungen Ihr AddOn auswählt und die Schaltfläche „Konfigurieren“ betätigt.

Sie sollten hier eine eigene Verwaltung Ihrer Datenbankeinstellungen, so z.B. Verwaltung von Tabelleneigenschaften anbieten.

18.3.2 <AddOn-Präfix>_DisplayPrivateUserSettings()

Wird aufgerufen, wenn der Benutzer im Dialog Benutzereinstellungen Ihr AddOn auswählt und die Schaltfläche „Konfigurieren“ betätigt.

Hier sollten Sie auch ggf. die Einstellungsoption zur Verfügung stellen, ob eine bestimmte Funktionalität des AddOns beim Start der Datenbank ausgeführt werden soll (analog zur Alarmliste oder Geburtstagsliste).

18.3.3 <AddOn-Präfix>_DisplayPublicUserSettings(ByRef psSelectedUserName as String)

Wird aufgerufen, wenn der Administrator im Dialog Erweiterte Benutzereinstellungen Ihr AddOn auswählt und die Schaltfläche „Konfigurieren“ betätigt.

Sollte innerhalb der AddOn-Benutzerverwaltung der Benutzer selektierbar sein und geändert werden, so muss dieser über den Parameter psSelectedUserName zurückgegeben werden.

18.4 Dokumentenbehandlung

Diese Makros können die Funktionalität der Dokumentenbehandlung erweitern.

18.4.1 <AddOn-Präfix>_CreateDocument(pType As CDocumentTypeEnum, ByRef pbProceed As Boolean)

Wird beim Erstellen eines Dokumenten aufgerufen. Damit wird ermöglicht, die Standarddokumentbehandlung um eigene Dokumente zu erweitern. Über den Parameter pbProceed kann gesteuert werden, ob mit der Standardbehandlung fortgefahren werden soll.

18.4.2 <AddOn-Präfix>_CompleteDocument(ByVal pDocument As OrgDbServer31.Document, ByRef pbProceed As Boolean)

Wird beim Ausfüllen eines Dokumenten aufgerufen. Damit wird ermöglicht, die Standarddokumentbehandlung um eigene Behandlung zu erweitern. Das auszufüllende Dokument wird über den Parameter pDocument übergeben. Über den Parameter pbProceed kann gesteuert werden, ob mit der Standardbehandlung fortgefahren werden soll.

18.5 Fakturierung

Diese Makros können die Funktionalität der Fakturierung erweitern. Es wird immer die aktuelle Instanz von CInvoicing übergeben, über den Parameter pbProceed kann gesteuert werden, ob mit der Standardbehandlung fortgefahren werden soll.

18.5.1 <AddOn-Präfix>_AddCurrentArticleToPositions(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Hinzufügen des aktuellen Artikels zu den Positionen einer Rechnung oder einer Verkaufschance aufgerufen.

18.5.2 <AddOn-Präfix>_CollectUrgingLetters(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Erstellen von Mahnschreiben aufgerufen.

18.5.3 <AddOn-Präfix>_CopyInvoice(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Kopieren einer Rechnung aufgerufen.

18.5.4 <AddOn-Präfix>_CreateInvoiceFromOpportunity(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Umwandeln eines Angebots in eine Rechnung aufgerufen.

18.5.5 <AddOn-Präfix>_FindArticle(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Suchen des einer Position zugrundeliegenden Artikels aufgerufen.

18.5.6 <AddOn-Präfix>_UpdateSum(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Aktualisieren der Rechnungs- oder Verkaufschancensumme aufgerufen.

18.5.7 <AddOn-Präfix>_UpdateTurnover(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Aktualisieren des Kundenumsatzes aufgerufen.

18.5.8 <AddOn-Präfix>_UpdateVAT(pobjInvoicing As CInvoicing, ByRef pbProceed As Boolean)

Wird beim Aktualisieren des MwSt-Satzes in den Positionen einer Rechnung oder einer Verkaufschance aufgerufen.