



Januar 2019

Vertec-Software Release 6.3

Inhaltsverzeichnis

1 Einführung	5
1.1 Neuinstallation von Vertec	5
1.2 Vor dem Update auf 6.3	5
1.3 Update von Vertec	5
1.4 Das erste Aufstarten nach dem Konvert	6
1.5 Firebird Version 3.0.4	6
2 Versionen	7
2.1 Highlights der Version 6.3	7
2.2 Eingestellte Features in Vertec 6.3	7
3 Dokumentenverwaltung und Reports	10
3.1 SharePoint Online DMS Schnittstelle	10
3.2 Erweiterte Office-Berichte als Excel-Ausgabe	11
3.3 Berücksichtigung von Projekt-Sprache einstellbar	18
3.4 Context-Variablen in OCL verwenden	19
3.5 Zugriff auf andere Felder in Feldberechnung	20
3.6 Context-Expression als Conditional-Band-Expression	20
3.7 Translations auch in Kopf- und Fusszeilen	21
4 Phone App	22
4.1 Belegbilder für Spesen	22
4.2 Leistungserfassung mit von, bis und Pause	23
4.3 Drittprodukte / Datenschutzerklärung in Phone App	23
4.4 Mobile Device Management Unterstützung	24
5 Rechnungsstellung	25
5.1 Unterstützung von Firmenadresse auf Zahlungstyp	25
6 Bearbeiter	26
6.1 Stichwort-Ordner Recht	26
6.2 Erfassung von Abwesenheiten neu mit DatePicker	27
7 Customizing	28
7.1 Auswahl-Suchfeld in SQL-Ordner Suchdialogen	28
7.2 Inaktive Einträge anzeigen bei SQL-Ordnern konfigurierbar	29
7.3 OCL Null Operator	30
7.4 Initiale Anzeige bei ReferenceBox konfigurierbar	31
7.5 Stichworte mittels Eintrag Id einblenden	31

7.6 Syntax Highlighting in XML Editor erweitert	32
7.7 TextArea Info Appearance	33
8 Scripting	34
8.1 Neuer Script Editor	34
8.2 Python Konsole	35
8.3 Vertec-Listen in Python	37
8.4 Neue Python Module im Restricted Scripting Modus	39
9 Config Sets	40
9.1 Ordner exportieren Subordner nicht mehr automatisch	40
9.2 Configset-reference entfernt	40
9.3 Configset-requirements entfernt	40
9.4 Listeneinstellungen im Config Set aufnehmen	40
9.5 Dateigrösse für Import von Config Sets erhöht	41
9.6 Bessere Fehlermeldungen bei Config-Sets	42
9.7 Speichern unter Dialog beim Erstellen von Config Sets	42
10 Schnittstellen	43
10.1 Extensions als Vertec Objekte	43
10.2 Schnittstelle kundenspezifisch anpassen	45
10.3 Listenbasiertes Buchen	46
10.4 BMD Debitoren Listen Schnittstelle	47
10.5 Unterstützung von Geschäftsbereichen in AbaConnect Webservice Schnittstelle	49
10.6 Abacus: Buchen ohne "Customer"-Lizenz	49
10.7 Sage50 Storno implementiert	49
10.8 Selectline: COM Freischaltcode nur verwenden falls angegeben	50
10.9 DATEV Schnittstelle überarbeitet	51
11 Performance	52
11.1 Firebird Page- und Buffergröße erhöht	52
11.2 Berücksichtigung von Aktivitäten bei Outlook Addin Projektauswahl optimiert	52
11.3 Projekt persistent auf allen Phasen speichern	53
12 Technik	54
12.1 Notif als Teil des CloudServer Dienstes	54
12.2 Verbessertes Upgrade Verhalten von Vertec.Desktop	55
12.3 XML Server mit fixem Zahlenformat	55
12.4 Key-Values und Tags über den XML Server setzen	55

1 Einführung

Dies sind die Release-Notes zur Vertec-Version 6.3. Bitte beachten Sie folgende Punkte:

1.1 Neuinstallation von Vertec

Für eine Neuinstallation verwenden Sie das zum Download verfügbare Installationsprogramm [VertecSetup-6.3.0.exe](#). Nähere Informationen zur Installation finden Sie in unserer Online Knowledge Base unter www.vertec.com/kb/neuinstallation.

1.2 Vor dem Update auf 6.3

Reports überprüfen

Mit Vertec Version 6.3 werden die alten Reporting-Systeme **Interne Berichte** und **Bericht-Designer Berichte** (Rb-Reports) abgeschaltet. Da sie dann nicht mehr lauffähig sind, werden sämtliche Berichte dieser Art aus den Berichtsvorlagen entfernt.

➔ **Die Reports sind nach dem Update auf Vertec 6.3 nicht mehr vorhanden!**

Kontrollieren Sie **VOR** dem Update auf Vertec 6.3 Ihre Reports und überführen Sie diejenigen, die Sie weiterverwenden möchten, auf das neue System.

Detaillierte Informationen finden Sie im Kapitel 2.2.

Manuell installierte Vertec Service Dienste deinstallieren

Der Notif, der Dienst für die Datenaktualisierung, ist nun Teil des Cloud Server Dienstes (siehe 12.1). Der bisherige Notif Server als Teil des Vertec Service Dienstes wird nicht mehr benötigt.

Somit entfällt der bisherige Vertec Service (**VertecServer.exe**) und wird vom Update entfernt.

Im Normalfall muss nichts gemacht werden. Wurde aber der Vertec Service unter eigenem Namen registriert, muss er auch manuell wieder deinstalliert werden. Dies muss **VOR** dem Update auf Vertec 6.3 erfolgen, da danach das VertecServer.exe, welches für die Deinstallation benötigt wird, nicht mehr vorhanden ist.

Weitere Informationen finden Sie im Kapitel 12.1.

1.3 Update von Vertec

Für ein Update einer Vertec Installation verwenden sie ebenfalls das Installationsprogramm. Bei grösseren Kundeninstallationen mit vielen kundenspezifischen Berichten und Listeneinstellungen empfehlen wir, vorgängig eine Testinstallation einzurichten und den Release-Wechsel durchzuspielen, bevor das Update auf dem aktiven System durchgeführt wird. Weitere Informationen finden Sie im Artikel **Testinstallationen** in unserer Online Knowledge Base unter www.vertec.com/kb/testinstallation.

Rückwärtskompatibilität

Bei den Features, bei welchen die Rückwärtskompatibilität eine Rolle spielt, wird das im entsprechenden Artikel im Abschnitt **Rückwärtskompatibilität** genau beschrieben:

Artikel	Seite
2.2 Eingestellte Features in Vertec 6.3	7
6.1 Stichwort-Ordner Recht	26
8.3 Vertec-Listen in Python	37

Artikel	Seite
9 Config Sets	40
12.1 Notif als Teil des CloudServer Dienstes	54
12.3 XML Server mit fixem Zahlenformat	55

1.4 Das erste Aufstarten nach dem Konvert

Der erste Start von Vertec nach dem Konvert ist wichtig und gehört ebenfalls zur Konvertierung dazu, da diverse Anpassungen an Daten und Strukturen etc. erst dann durchgeführt werden. Es ist wichtig, dass dieser erste Start reibungslos durchläuft. Keinesfalls sollte der erste Startvorgang nach einem Konvert abgebrochen werden.

1.5 Firebird Version 3.0.4

Vertec 6.3 liefert die neuste Firebird Version 3.0.4 mit. Das Update wird vom Vertec-Setup nach einer entsprechenden Abfrage automatisch vorgenommen.

Weitere Informationen Weitere Informationen zu Firebird finden Sie in der Online Knowledge Base unter www.vertec.com/kb/firebird.

2 Versionen

2.1 Highlights der Version 6.3

Die Version 6.3 wartet wieder mit vielen Highlights auf. Die Spannendsten hier in einem kurzen Überblick:

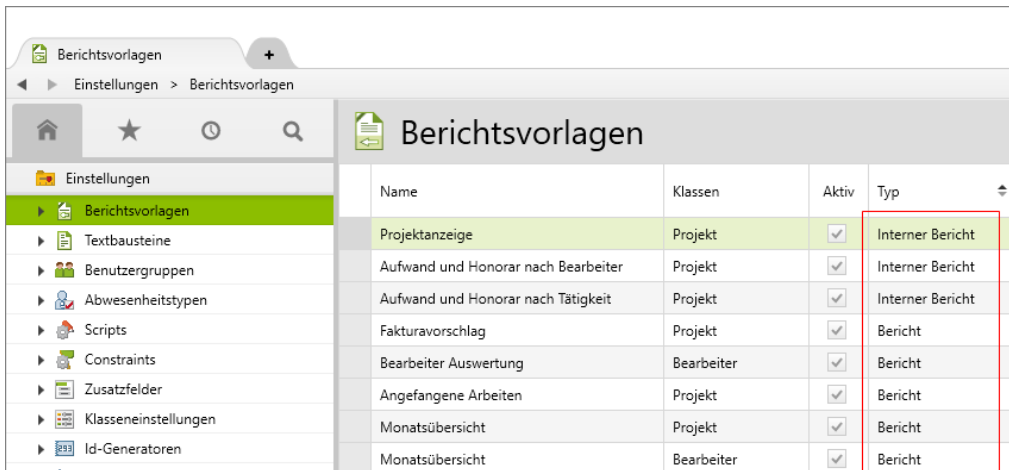
Artikel	Seite
3.1 SharePoint Online DMS Schnittstelle	10
3.2 Erweiterte Office-Berichte als Excel-Ausgabe	11
7.1 Auswahl-Suchfeld in SQL-Ordner Suchdialogen	28
8.2 Python Konsole	35
12.1 Notif als Teil des CloudServer Dienstes	54

2.2 Eingestellte Features in Vertec 6.3

Interne Berichte und Bericht-Designer Berichte werden eingestellt

Mit Vertec 6.3 werden die alten Reporting-Systeme **Interne Berichte** und **Bericht-Designer Berichte** (Rb-Reports) abgeschaltet. Da sie dann nicht mehr lauffähig sind, werden sämtliche Berichte dieser Art aus den Berichtsvorlagen entfernt. Sie sind nach einem Update auf Vertec 6.3 nicht mehr verfügbar.

Es geht um alle Berichte unter **Einstellungen > Berichtsvorlagen**, welche vom Typ **Bericht** oder **Interner Bericht** sind:



Name	Klassen	Aktiv	Typ
Projektanzeige	Projekt	☒	Interner Bericht
Aufwand und Honorar nach Bearbeiter	Projekt	☒	Interner Bericht
Aufwand und Honorar nach Tätigkeit	Projekt	☒	Interner Bericht
Fakturavorschlag	Projekt	☒	Bericht
Bearbeiter Auswertung	Bearbeiter	☒	Bericht
Angefangene Arbeiten	Projekt	☒	Bericht
Monatsübersicht	Projekt	☒	Bericht
Monatsübersicht	Bearbeiter	☒	Bericht

Abbildung 1: Legacy Berichte in den Berichtsvorlagen

Vertec liefert folgende Berichte neu als erweiterte Office-Berichte aus:

Bericht	Registriert auf	Vorlage im Vertec Reports-Verzeichnis
Angefangene Arbeiten	Projekt/Mandat	CommencedWork.docx CommencedWork.xlsx
Bearbeiter Auswertung	Bearbeiter	UserAnalysis.docx
Budget Übersicht	Projekt/Mandat	BudgetOverview.docx
Fakturavorschlag	Projekt/Mandat	InvoiceProposal.docx

Bericht	Registriert auf	Vorlage im Vertec Reports-Verzeichnis
Ferienabrechnung	Bearbeiter	VacationReport.docx
Monatskontrolle Leistungen	Bearbeiter	MonthlyControl.docx
Monatsübersicht mit Sollzeiten	Bearbeiter	MonthlyOverview.docx
Projekt Auswertung	Projekt/Mandat	ProjectEvaluation.docx
Rechnung mit Leistungsliste	Rechnung	InvoiceWithList.docx
Rekapitulation Buchungen	Projekt/Mandat	RecapitulationOfBookings.docx
Rückzuerstattende Spesen nach Typ	Bearbeiter	ExpensesToReimburse.docx
Überstundenkontrolle	Bearbeiter	OvertimeReport.docx
Vorschüsse	Projekt/Mandat	AdvancePayments.docx
Zeitabrechnung	Bearbeiter	TimeSheet.docx

Wenn Sie alte und neue Berichte vergleichen möchten, machen Sie bitte vor dem Update auf Vertec 6.3 eine Testinstallation (siehe auch Kapitel 1.3). Bitte beachten Sie, dass die alten Reports nach einem Update auf 6.3 nicht mehr verfügbar sein werden.

Die von Vertec mitgelieferten erweiterten Office-Berichte werden in Version 6.3 automatisch aktiviert und alle entsprechenden alten Berichte gelöscht.

Weitere Informationen zu den erweiterten Office-Berichten finden Sie in der Online Knowledge Base unter www.vertec.com/kb/erweiterte-office-berichte.

Wenn Sie diese Berichte ohne eigene Konfiguration "as is" verwenden, können Sie danach mit der gleichen Funktionalität weiterarbeiten. Ansonsten passen Sie bitte vorgängig auch diese Berichte gemäss Ihren Anforderungen an.

Bei Fragen wenden Sie sich bitte an Ihren Vertec-Betreuer.

Bearbeiter-Jahre definitiv entfernt

Mit dem im Dezember 2008 mit Vertec 5.4 umgestellten Sollzeitsystem wurden die vorher verwendeten BearbeiterJahre durch BearbeiterVorgaben abgelöst.

Dabei wurde für die Rückwärtskompatibilität ein OCL Operator getJahre eingeführt, mit welchem die bisherigen BearbeiterJahre instantiiert werden konnten.

Mit der Version 6.3 werden nun sowohl die BearbeiterJahre als auch der OCL Operator komplett entfernt.

Kunden, welche bereits vor 5.4 mit Vertec gearbeitet haben und kundenspezifische Reports, Listeneinstellungen, ExpressionOrdner etc. einsetzen, die auf dem früheren System mit BearbeiterJahren oder dem OCL Operator getJahre basieren, müssen diese nun definitiv umstellen.

Eine Übersicht über die notwendigen Umstellungen mit vielen Vorher/Nachher-Vergleichen finden Sie im Artikel [Umstellung des Sollzeitsystems ab Vertec Version 5.4](#) in der Online Knowledge Base unter www.vertec.com/kb/umstellungssollzeitsystem.

Weitere Informationen Alle OCL Operatoren für Sollzeit und Ferien finden Sie in der Online Knowledge Base unter www.vertec.com/kb/ocl#bearbeiteroperatoren.

VB Script Editor entfernt

Ab Version 6.3 gibt es keinen VB Script Editor mehr. Der entsprechende Menüpunkt im Menü Einstellungen wurde entfernt.

Nach wie vor können VB Scripts als Scripts registriert und dort via **Ausführen** Button zum Test ausgeführt werden.

Als Script Editor gibt es einen neuen Python Script Editor. Dieser ist in Kapitel 8.1 beschrieben.

3 Dokumentenverwaltung und Reports

3.1 SharePoint Online DMS Schnittstelle

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

Die SharePoint Online DMS Extension erlaubt die Integration mit SharePoint Online.

- Vertec-generierte und erweiterte Office-Berichte können ihre Ausgabe direkt in Sharepoint speichern. Das Ergebnis wird in Office Online angezeigt und kann direkt weiterverarbeitet werden.
- Dokumente und Mails können aus den Office-, Explorer- und Outlook Addins direkt in SharePoint gespeichert werden.
- Dokumente können aus Vertec heraus in Office Online geöffnet und bearbeitet werden.

Folgende Vorgänge sind mit der vorliegenden Version nicht möglich:

- Speichern von Office-generierten Berichten in SharePoint
- Verwenden von Berichts-Vorlagen aus SharePoint

Zur Inbetriebnahme der Schnittstelle sind folgende Schritte nötig (anhand der Anwendung mit Office 365 erklärt):

Vertec Applikation in Office 365 registrieren

Dieser Vorgang muss einmal pro Firma durchgeführt werden. Das genaue Vorgehen finden Sie in der Beschreibung der **SharePoint Online DMS Schnittstelle** auf unserer Online Knowledge Base unter www.vertec.com/kb/dmssharepoint.

Dabei erhalten Sie einen Schlüssel, welcher in den Vertec Systemeinstellungen unter **DMS SharePoint Application Secret** eingegeben werden muss, sowie die Anwendungs-ID, welche in den Vertec Systemeinstellungen unter **DMS SharePoint Application ID** eingetragen werden muss (siehe dazu den Abschnitt **Schnittstelle in Vertec installieren** weiter unten).

Sharepoint Site anlegen oder bestimmen

Dieser Vorgang muss einmal pro Firma durchgeführt werden. Das genaue Vorgehen finden Sie in der Beschreibung der **SharePoint Online DMS Schnittstelle** auf unserer Online Knowledge Base unter www.vertec.com/kb/dmssharepoint.

Dabei erhalten Sie einen Pfad (URL), welcher in den Vertec Systemeinstellungen unter **DMS SharePoint Document Library URL** eingetragen werden muss (siehe dazu den Abschnitt **Schnittstelle in Vertec installieren** weiter unten).

Schnittstelle in Vertec installieren

Um die Schnittstelle zu installieren, öffnen Sie im Ordner **Einstellungen > Extensions** die **SharePoint Online DMS Schnittstelle** und klicken auf **Installieren**.

Die Schnittstelle legt folgende **Systemeinstellungen CRM/Aktivitäten** an:

DMS SharePoint Application ID: Vertec Registrierungs-Schlüssel in SharePoint Online, siehe Abschnitt **Vertec Applikation in Office 365 registrieren** weiter oben.

PropertyName: DmsSharePointApplicationId. StringProperty.

DMS SharePoint Application Secret: Anwendungs-ID in SharePoint Online, siehe Abschnitt [Vertec Applikation in Office 365 registrieren](#) weiter oben.

PropertyName: DmsSharePointApplicationSecret. StringProperty.

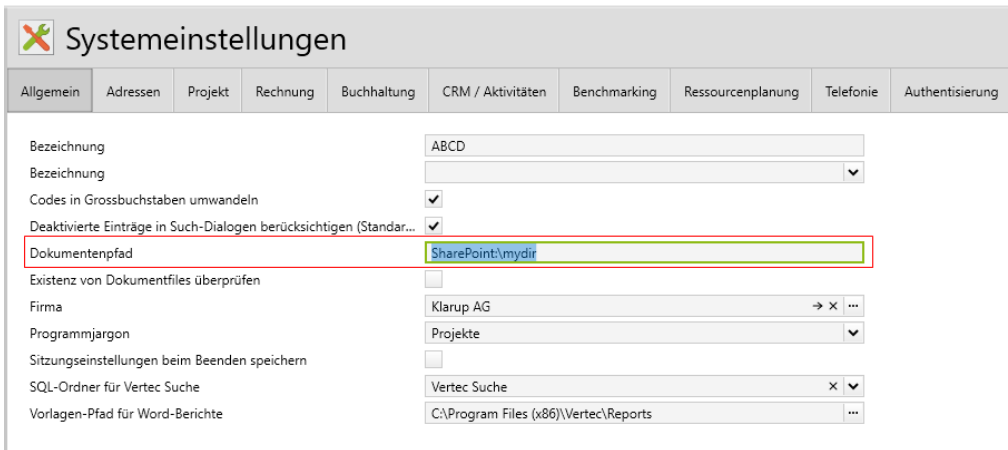
DMS SharePoint Library URL: Pfad zur freigegebenen SharePoint Site, siehe Abschnitt [Sharepoint Site anlegen oder bestimmen](#) weiter oben.

PropertyName: DmsSharePointDocumentLibraryUrl. StringProperty.

Nach Änderung der Werte in den Systemeinstellungen muss Vertec neu gestartet oder bei der SharePoint Online DMS Extension auf den Button [Reload](#) geklickt werden.

Dokumentenpfad

Der Basispfad – Systemeinstellungen Allgemein > Dokumentenpfad – wird mit [SharePoint:](#) gefolgt vom Verzeichnis, in welches im SharePoint die Vertec Dokumente gespeichert werden sollen, angegeben, z.B. [SharePoint:\mydir](#).



The screenshot shows the 'Systemeinstellungen' window with the 'Allgemein' tab active. The 'Dokumentenpfad' field is highlighted with a red box and contains the text 'SharePoint:\mydir'. Other fields include 'Bezeichnung' (ABC), 'Codes in Grossbuchstaben umwandeln' (checked), 'Deaktivierte Einträge in Such-Dialogen berücksichtigen (Standard...)' (checked), 'Existenz von Dokumentfiles überprüfen' (unchecked), 'Firma' (Klarup AG), 'Programmjargon' (Projekte), 'Sitzungseinstellungen beim Beenden speichern' (unchecked), 'SQL-Ordner für Vertec Suche' (Vertec Suche), and 'Vorlagen-Pfad für Word-Berichte' (C:\Program Files (x86)\Vertec\Reports).

Abbildung 2: Der Dokumentenpfad zeigt auf SharePoint

Die Dokumente, die aus Vertec heraus generiert oder via Office-Addins ins Vertec gespeichert werden, werden so direkt in SharePoint Online gespeichert.

Das Vertec Pfad-System wurde so erweitert, dass mit dem [SharePoint:](#) Präfix (anstelle von z.B. [C:.](#)) ein Sharepoint Pfad angegeben werden kann. Es ist also auch möglich, nur bestimmte Dokumente in Sharepoint zu speichern, ohne den generellen Dokumentpfad auf Sharepoint umzustellen.

3.2 Erweiterte Office-Berichte als Excel-Ausgabe

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Erweiterte Office-Berichte können neu auch im Excel-Format ausgegeben werden.

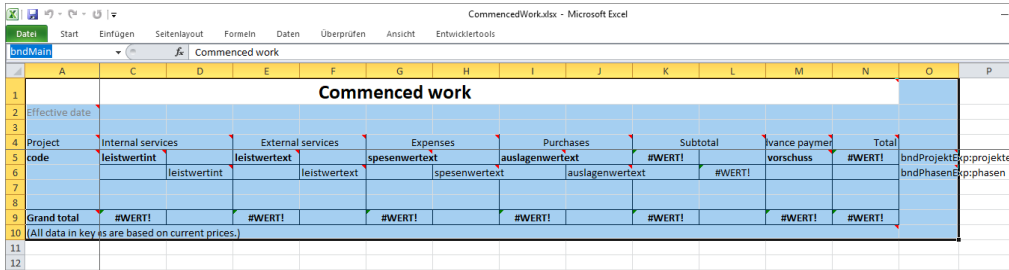
Der zugrunde liegende Python-Code ist dabei derselbe wie bei Word-Berichten; die beiden Formate unterscheiden sich nur durch die Vorlage.

Excel-Vorlagen sind analog zu den Word-Vorlagen aufgebaut. In den folgenden Beispielen ist jeweils die Vorlage [CommencedWork.xlsx](#) dargestellt, welche mit dieser Vertec Version standardmässig ausgeliefert wird (Report Angefangene Arbeiten).

Bänder

Um ein Band zu setzen, markieren Sie die entsprechenden Zellen und fügen im Feld oben links den Bandnamen ein.

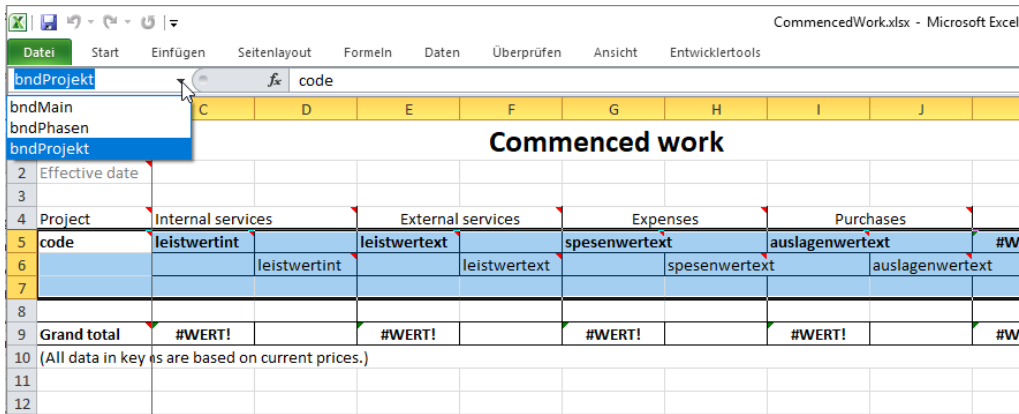
Wie bei den Word-Vorlagen gibt es ein Main-Band, welches alles umfasst und sich auf das Hauptobjekt bezieht, also das Main-Frame enthält.



	A	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Commenced work														
2	Effective date														
3															
4	Project	Internal services	External services	Expenses	Purchases	Subtotal	Advance payment	Total							
5	code	leistwertint	leistwertext	spesenwertext	auslagenwertext	#WERT!	vorschuss	#WERT!	bndProjekt	op:projekte					
6		leistwertint	leistwertext	spesenwertext	auslagenwertext	#WERT!			bndPhasen	op:phasen					
7															
8															
9	Grand total	#WERT!	#WERT!	#WERT!	#WERT!	#WERT!	#WERT!	#WERT!							
10	(All data in key es are based on current prices.)														
11															
12															

Abbildung 3: Das Haupt-Band wird gesetzt

Die anderen Bänder und Sub-Bänder werden gleich gesetzt: Zellen markieren und oben links den Namen einfügen:



	C	D	E	F	G	H	I	J
1	Commenced work							
2	Effective date							
3								
4	Project	Internal services	External services	Expenses	Purchases			
5	code	leistwertint	leistwertext	spesenwertext	auslagenwertext	#W		
6		leistwertint	leistwertext	spesenwertext	auslagenwertext			
7								
8								
9	Grand total	#WERT!	#WERT!	#WERT!	#WERT!	#W		
10	(All data in key es are based on current prices.)							
11								
12								

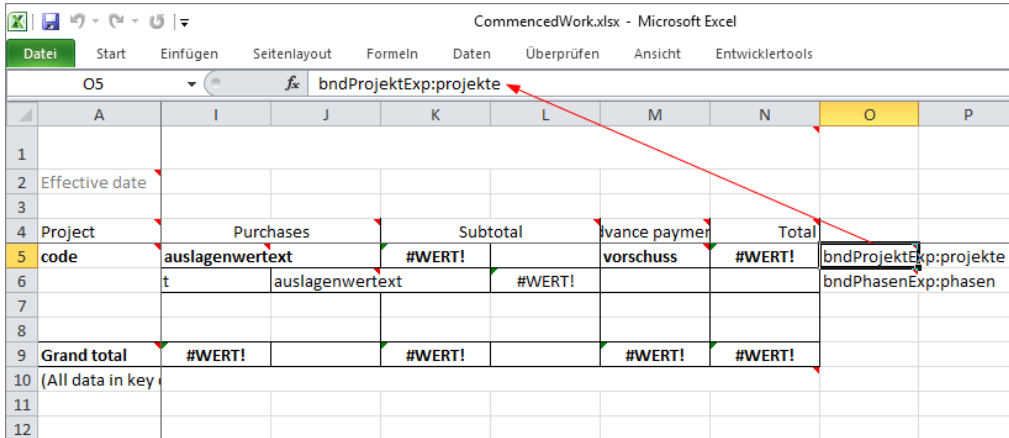
Abbildung 4: Verschiedene Bänder eines erweiterten Excel-Reports

Wichtig Achten Sie beim Setzen der Bänder darauf, dass nicht die gesamten Zeilen (hinten rechts bis unendlich) markiert sind, sondern nur die Zellen, die tatsächlich benötigt werden. Sonst dauert der Aufbau des Berichts länger, da Vertec durch alle Zellen iterieren muss, um herauszufinden, ob irgendwo ein Wert gesetzt ist.

Band-Expressions

Es gibt zwei Arten von Band-Expressions: Cond- und Exp-Bands.

- Der Text beginnt mit dem Namen des Bands, auf welches sich die Expression bezieht, gefolgt von Exp oder Cond, je nachdem, um welche Art von Expression es sich handelt (siehe unten).
- Die Expression (Referenz auf die Daten) wird mit `:name` angegeben.

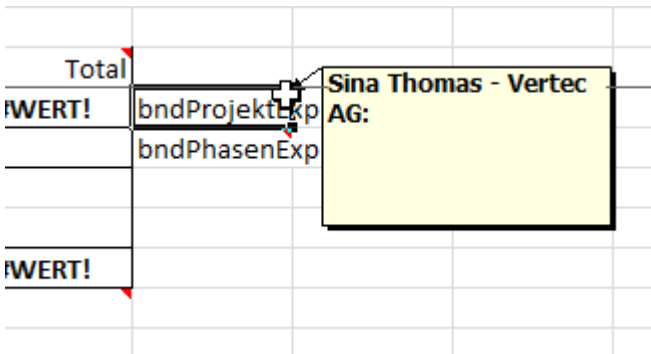


	A	I	J	K	L	M	N	O	P
1									
2	Effective date								
3									
4	Project	Purchases		Subtotal		Advance payment		Total	
5	code	auslagenwerttext		#WERT!		vorschuss		#WERT!	bndProjektExp:projekte
6		t	auslagenwerttext	#WERT!					bndPhasenExp:phasen
7									
8									
9	Grand total	#WERT!		#WERT!		#WERT!		#WERT!	
10	(All data in key)								
11									
12									

Abbildung 5: Eine Band-Expression

Als Band-Expression wird also zum Beispiel `bndProjekteExp:projekte` geschrieben, dieser Text markiert und mit einem leeren Kommentar versehen.

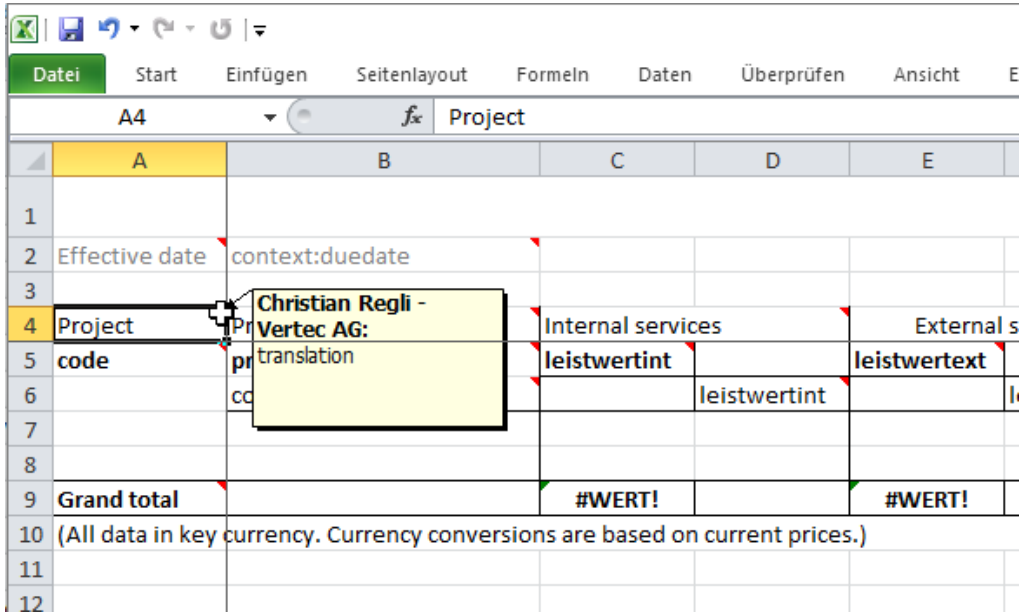
Hinweis Excel fügt beim Anlegen eines Kommentars den Autor automatisch ein. Es ist wichtig, dass Sie diesen lassen und nicht rauslöschen. Mit "leerem" Kommentar ist also kein gänzlich leerer Kommentar gemeint, sondern einer, bei dem nur der Autor drin steht:



	A	I	J	K	L	M	N	O	P
1									
2	Effective date								
3									
4	Project	Purchases		Subtotal		Advance payment		Total	
5	code	auslagenwerttext		#WERT!		vorschuss		#WERT!	bndProjektExp:projekte
6		t	auslagenwerttext	#WERT!					bndPhasenExp:phasen
7									
8									
9	Grand total	#WERT!		#WERT!		#WERT!		#WERT!	
10	(All data in key)								
11									
12									

Felder

Ein Feld ist ein durch einen Kommentar markierter Text innerhalb eines Bandes, der beim Ausführen des Reports durch Daten aus Vertec ersetzt wird. Der markierte Text enthält dabei die Referenz auf die im Python-Code definierten Felder. Der Kommentar selbst ist leer (siehe oben) oder enthält das Schlüsselwort `translation`, um die Übersetzung für dieses Feld zu aktivieren.



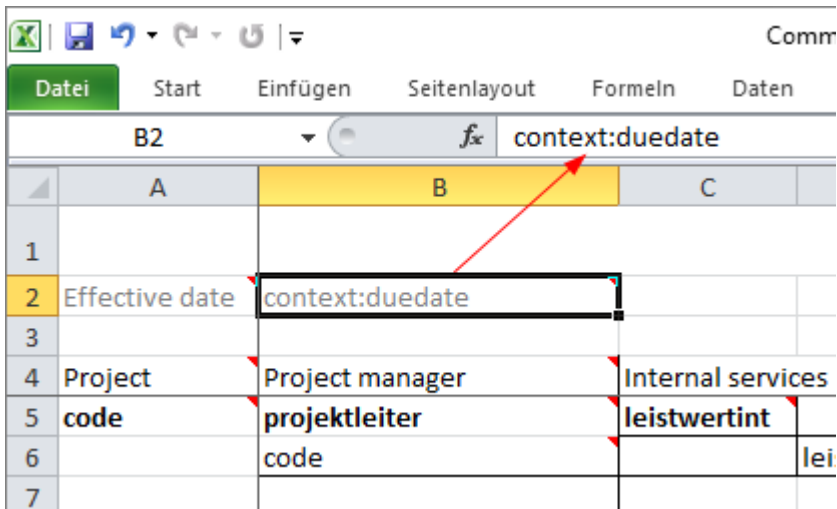
	A	B	C	D	E
1					
2	Effective date	context:duedate			
3					
4	Project	Internal services			External s
5	code	leistwertint			leistwerttext
6			leistwertint		
7					
8					
9	Grand total		#WERT!		#WERT!
10	(All data in key currency. Currency conversions are based on current prices.)				
11					
12					

Abbildung 6: Ein Begriff wird übersetzt

Hinweis Pro Zelle kann jeweils nur ein Begriff gesetzt werden. Es können also nicht in einer Zelle gleichzeitig ein Feld und eine Band-Expression gesetzt werden. Im Beispielreport in Abbildung 3 wurde das so gelöst, dass in der Zelle am Ende der Zeile die Band-Expression eingefügt und das Band um eine Spalte breiter gemacht wurde.

Context-Variablen

Auch die Context-Variablen werden unterstützt. Statt einem Python-Feld enthält die Zelle den Text `context:`, gefolgt von der entsprechenden Variable:



	A	B	C
1			
2	Effective date	context:duedate	
3			
4	Project	Project manager	Internal services
5	code	projektleiter	leistwertint
6		code	lei
7			

Abbildung 7: Eine Context-Variable wird referenziert

Datentypen in Excel

Im Gegensatz zu Word gibt es in Excel verschiedenen Datentypen. Die automatische Erkennung von Excel ist dabei nicht immer sinnvoll. Deshalb kann in der Excel-Vorlage die entsprechende Zelle mit der gewünschten Formatierung versehen werden. Beim Einfügen der Daten bzw. Kopieren der Zellen nach unten wird die Formatierung mitkopiert.

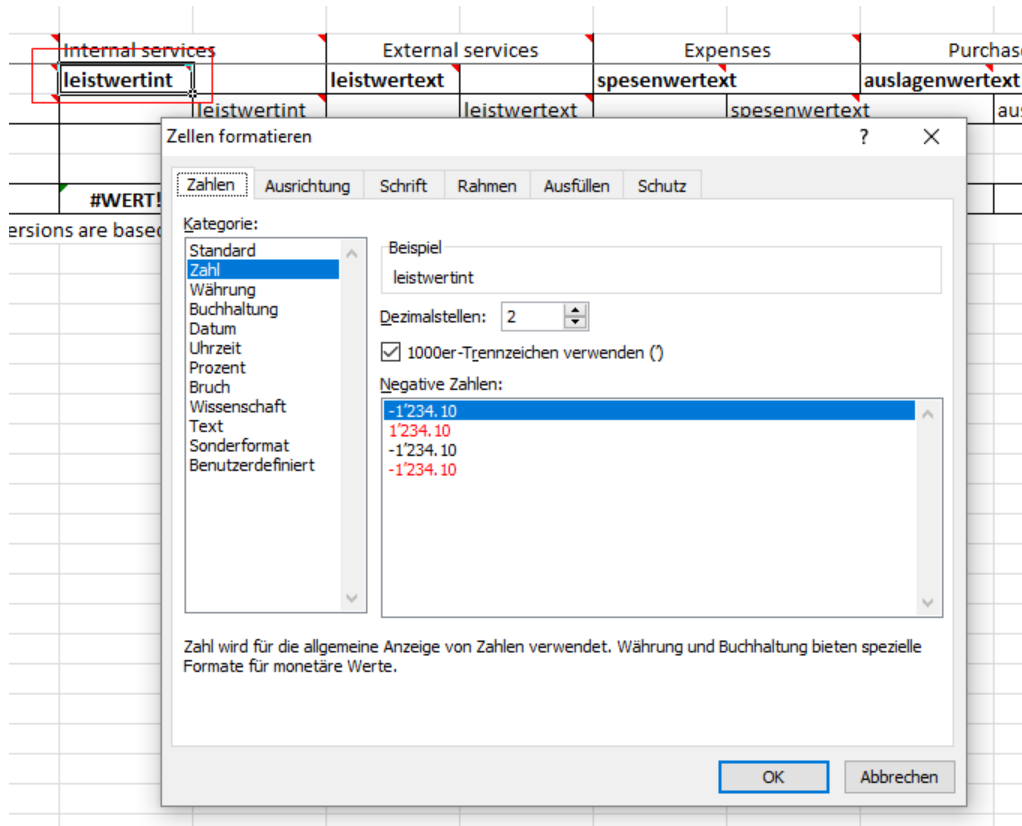


Abbildung 8: Eine Zelle wird als Zahl formatiert.

Summierung

Summierung in Excel funktioniert mit der gewohnten Excel Formel `=Summe (<Band>)`.

Zeilensummen werden wie üblich mit den Zell-Bezügen gemacht:

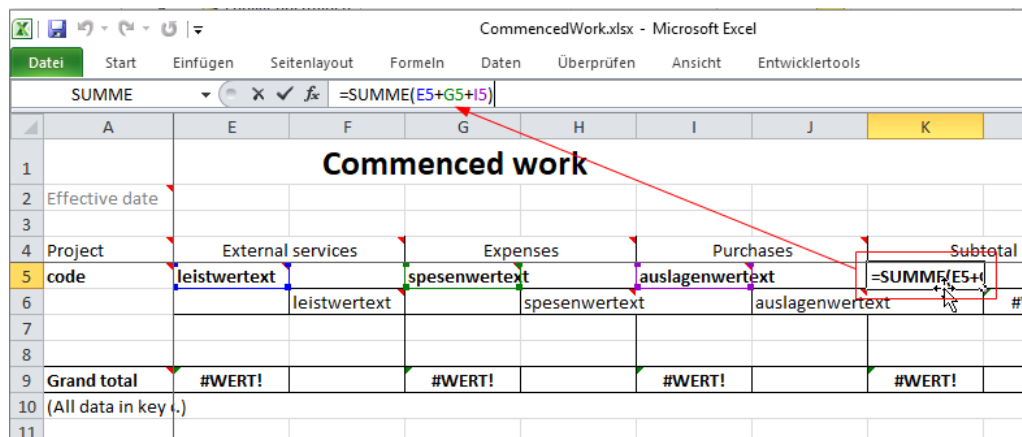


Abbildung 9: Zwischensumme, die sich auf die Werte in der Zeile bezieht

Bei den Summen in den Spalten gelten folgende Richtlinien:

- Es werden sämtliche Excel-Operatoren akzeptiert. Das Format muss aber `=Operator (<Band>)` sein.
- Die Formel muss sich ausserhalb des durch die Summe referenzierten Bands befinden, aber innerhalb des dieses einschliessenden Bands, also eine Ebene tiefer.
- Summiert wird die Spalte, in der sich die Formel befindet.

- Zwischensummen sind möglich, aber nur innerhalb desselben Datensatzes.
- Summierung aller Teilsummen in derselben Spalte wird nicht unterstützt.

	A	B	C	D	E	F	G	H
46								
47	Rechnung	09090022						
48								
49	Leistungen							
50								
51	Datum	Text	WertExt					
52	14.08.2009	div. Papierarbeiten	264.00					
53	14.08.2009	e und Versand an Kunde	720.00					
54								
55	Subtotal		=SUMME(bndLeistung)					
56								
57	Rechnung	08120014						
58								
59	Leistungen							
60								
61	Datum	Text	WertExt					
62	25.10.2008	Projektarbeiten	320.00					
63								
64	Subtotal		=SUMME(bndLeistung)					
65								
66	Rechnung	09080015						
67	Datum	11.08.2009						
68	Total							
69								
70	Gesamttotal		=SUMME(bndLeistung)					
71								

Abbildung 10: Summierungen sind nur innerhalb desselben Datensatzes möglich

Aufgrund der Formatierung der Datentypen (siehe den Abschnitt **Datentypen in Excel**) kann es bei den Zellbezügen zum Hinweis kommen "Ein in der Formel verwendeter Wert ist vom falschen Datentyp."

	Subtotal	Advance paymer	Total
	#WERT!	vorschuss	#WERT!
agenw			

Ein in der Formel verwendeter Wert ist vom falschen Datentyp.

Abbildung 11: Hinweis auf "falschen" Datentyp

Dies ist deshalb so, weil in den Feldern z.B. `leistwertint`, `leistwerttext` etc. steht, was ja nicht summiert werden kann. Nach der Generierung des Reports funktioniert die Summierung, da dann die Zahlenwerte drin stehen.

Registrierung

Erweiterte Office-Berichte als Excel-Bericht werden genau gleich registriert wie die entsprechenden Word-Berichte. Da sie sich die Code-Basis teilen, kann in derselben Registrierung auch beides gleichzeitig registriert werden:

Erweiterter Office-Bericht

Name	Angefangene Arbeiten (erweitert)	☒ Aktiv
Gruppierung	0 Einzelindex 0	
Klassen	Projekt	▼
Bedingung		...
Ausgabeformat	PDF	▼
☒ Einzelreport ☒ Listenreport		

Automatisches Speichern

Pfad	
Betreff	
☐ Dialog zeigen ☐ Aktivität erzeugen	
☐ Verknüpfung auf Hauptobjekt ☐ Verknüpfung auf Adresse	
Aktivitätstyp	x ▼
Bemerkung	

Dokumentvorlagen

Word Vorlage	CommencedWork.docx	...
Excel Vorlage	CommencedWork.xlsx	...

☐ Übersetzung in Projektsprache aktivieren

Abbildung 12: Eine Excel-Vorlage wird registriert

Soll die Ausgabe über Excel erfolgen, muss oben beim **Ausgabeformat** **Excel** eingestellt werden:

Erweiterter Office-Bericht

Name	Angefangene Arbeiten (erweitert)	☒ Aktiv
Gruppierung	0 Einzelindex 0	
Klassen	Projekt	▼
Bedingung		...
Ausgabeformat	Excel Word PDF Excel	

Abbildung 13: Bei Ausgabeformat wird Excel eingestellt

Möchten Sie über das Ausgabeformat ad hoc entscheiden, wenn Sie den Bericht ausführen, klicken Sie das Häkchen bei **Dialog zeigen** an. In diesem Fall erscheint vor dem Ausführen ein Dialog, wo das Ausgabeformat gewählt werden kann:

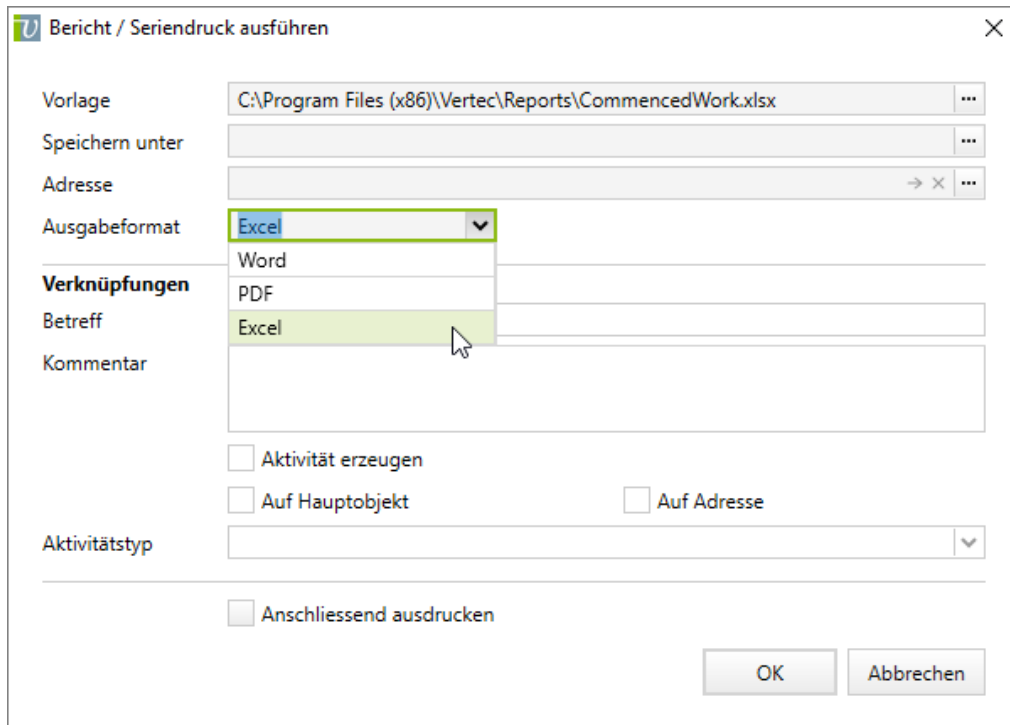


Abbildung 14: Das Ausgabeformat wird beim Ausführen gewählt

- **Word** und **PDF** verwenden immer die Word-Vorlage
- **Excel** verwendet immer die Excel-Vorlage

Passt das eingestellte Ausgabeformat nicht zu den registrierten Vorlagen – wird z.B. Excel gewählt, wenn nur eine Word-Vorlage hinterlegt ist – erscheint eine Fehlermeldung.

Weitere Informationen Weitere Informationen zu den erweiterten Office-Berichten finden Sie in der Online Knowledge Base unter www.vertec.com/kb/erweiterte-office-berichte.

3.3 Berücksichtigung von Projekt-Sprache einstellbar

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3.

Bei der Berichtsregistrierung von erweiterten Office-Berichten gibt es neu ein Häkchen **Übersetzung in Projektsprache aktivieren**:

Erweiterter Office-Bericht

Name	Rechnung mit Leistungsliste	☒ Aktiv
Gruppierung	0 Einzelindex 0	
Klassen	Rechnung	▼
Bedingung		...
Ausgabeformat	PDF	▼
☒ Einzelreport ☐ Listenreport		

Automatisches Speichern

Pfad	
Betreff	
☒ Dialog zeigen ☒ Aktivität erzeugen	
☒ Verknüpfung auf Hauptobjekt ☐ Verknüpfung auf Adresse	
Aktivitätstyp	x ▼
Bemerkung	

Dokumentvorlagen

Word Vorlage	InvoiceWithList.docx	...
Excel Vorlage		...

☒ Übersetzung in Projektsprache aktivieren

Abbildung 15: Übersetzung in Projektsprache aktivieren

Ist es aktiviert, hat das folgende Auswirkungen:

- Berichte, die auf einzelnen Objekten ausgeführt werden, welche eine definierte Beziehung zu einem Projekt haben (z.B. Rechnungen oder Projektauswertungen, werden in diese Projektsprache übersetzt (bzw. alle mit **Translation** versehenen Labels).
- Berichte, die auf mehreren Objekten ausgeführt werden (Listenberichte), werden weiterhin in die aktuelle Vertec Oberflächensprache übersetzt.

Standardmässig aktiviert ist diese Option bei den von Vertec mitgelieferten erweiterten Office-Berichten (siehe 2.2) nur bei der **Rechnung mit Leistungsliste**. Bei allen anderen muss die Option bei Bedarf manuell aktiviert werden.

3.4 Context-Variablen in OCL verwenden

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.7

Ab sofort können die Context-Variablen von erweiterten Office-Berichten direkt in OCL verwendet werden. Berechnungen der folgenden Art können dadurch stark vereinfacht werden:

```
class Bearbeiter(Frame):
    businessclass = "Projektbearbeiter"
    fields = [
        MinuteField("balance", "calc_balance"),
    ]
    def calc_balance(context):
        return context.currentobject.evalocl("self-
        >getFerienSaldo(%s)" % vtcapp.ocldate(context.todate))
```

Durch die Möglichkeit der Verwendung der Context-Variable in OCL reduziert sich dieser Code auf:

```
class Bearbeiter(Frame):
    businessclass = "Projektbearbeiter"
    fields = [
        OclMinuteField("balance", "self->getFerienSaldo(todate)"),
    ]
```

Es gelten folgende Regeln:

- Namensgebung: Die Namen der automatisch gesetzten Variablen sind gleich wie auf dem Contextobjekt, nämlich `varXY` und `varXYList`. Die manuell auf dem Context gesetzten Variablen können mit Keywords in OCL kollidieren (self, date, now). In diesem Fall erscheint eine Fehlermeldung.
- Die Reporting OCL Variablen werden auf dem privaten OCL Evaluator des ausgeführten Reports angelegt. Das bedeutet, dass die Variablen nur in OCL Expressions von OCL-Feldern oder in der `context.evalocl()` Methode verwendet werden können. Alle anderen OCL Aufrufe, insbesondere die `evalocl()` Methoden von Businessobjekten, verwenden den globalen OCL Evaluator und der kennt die lokalen Report-Variablen nicht. Statt:

```
templist = spesenList.evalocl("self->select(getbearbeiter=
                                varBearbeiter)->ordermulti('datum:boldid')")
```

verwendet man deshalb:

```
templist = context.evalocl("spesenList->select(getbearbeiter=
                                varBearbeiter)->ordermulti('datum:boldid')")
```

- Es werden nur OCL kompatible Werte als Variablen unterstützt (Vertec-Objekte und -Listen, einfache Werte). Bei inkompatiblen Werten (z.B. Python Dictionaries, Python-Listen, etc.) erscheint ein Fehler.

3.5 Zugriff auf andere Felder in Feldberechnung

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.8

Mit `context.get_fieldvalue(fieldname)` kann in erweiterten Office-Berichten auf andere Felder desselben Frames zugegriffen werden. Dadurch müssen Berechnungen nur noch einmal durchgeführt und danach kann direkt mit dem Wert weitergearbeitet werden. Das ist nicht nur übersichtlicher, sondern auch ein Performance-Vorteil.

3.6 Context-Expression als Conditional-Band-Expression

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.4

Als Conditional-Expression eines Bands in erweiterten Office-Berichten kann ab sofort auch eine Context-Expression angegeben werden. Die Syntax lautet:

```
bndXXXCond:context:name.
```

Total	sum:projekte.b	sum:projekte.bdg	sum:projekte.e
bndTotalCond:context:showtotal ^[sth23]	dgaufwand ^[sth24]	honorar ^[sth25]	faufwand ^[sth26]


```

131 def calculate_main_frame(context):
132
133     ---Projektliste sortieren.
134     context.projekte = vtcapp.createlist("Projekt", context.rootlist.evalocl("self->orderby
(asstring)"))
135
136     ---Gesamttotal nur anzeigen, wenn auf einer Liste ausgeführt
137     context.showtotal = len(context.projekte) > 1
138

```

Abbildung 16: Eine Context-Expression wird als Conditional-Expression angegeben

Weitere Informationen Weitere Informationen zu Conditional Bands in erweiterten Office-Berichten finden Sie in der Online Knowledge Base unter www.vertec.com/kb/erweiterte-office-berichte#band-expressions.

Beispiel

```

class Projekt(Frame):
    businessclass = "Projekt"
    fields = [
        OclMinuteField("bdgaufwand", "if (phasen->size>0) then
            phasen.bdgvalue('planminutenint', duedate, -1)->sum else
            planminutenint endif"),
        OclMinuteField("effaufwand", "leistsums->select(projekt=
            varProjekt)->collect(minutenintoffen+
            minutenintverrechnet)->sum"),
        MinuteField("restaufwand", "calculate_restaufwand"),

        def calculate_restaufwand(context):
            return context.get_fieldvalue("bdgaufwand") -
                context.get_fieldvalue("effaufwand")

```

3.7 Translations auch in Kopf- und Fusszeilen

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.4

In Kopf- und Fusszeilen von erweiterten Office-Berichten können Begriffe in doppelten Anführungszeichen mit **Translation** übersetzt werden. Beispielsweise möchten wir "Page" X "of" XX übersetzen lassen:

```
{{Translation:Page}} X {{Translation:of}} XX
```

4 Phone App

4.1 Belegbilder für Spesen

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.7

Bei der Spesenerfassung in der Phone App kann im unteren Teil ein Belegbild erfasst werden.

Durch Tippen in das Feld Belegbild kann ein Bild aufgenommen oder aus der Bildergalerie ausgewählt werden:



Abbildung 17: Ein Belegbild wird hinzugefügt

Das Belegbild wird dann direkt auf der Spese angezeigt:

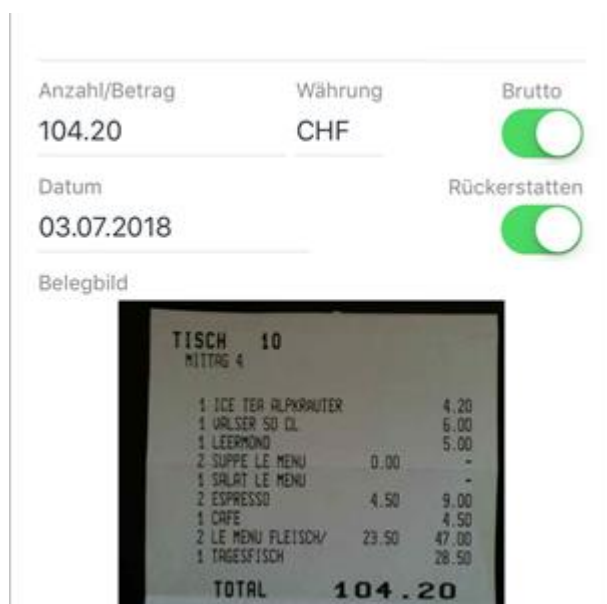


Abbildung 18: Das Belegbild wird auf der Spese angezeigt

4.2 Leistungserfassung mit von, bis und Pause

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.8

Auch in der Phone App können Leistungen nun via Angabe der Uhrzeit **von** - **bis** und mit eventueller **Pausenzeit** erfasst werden:

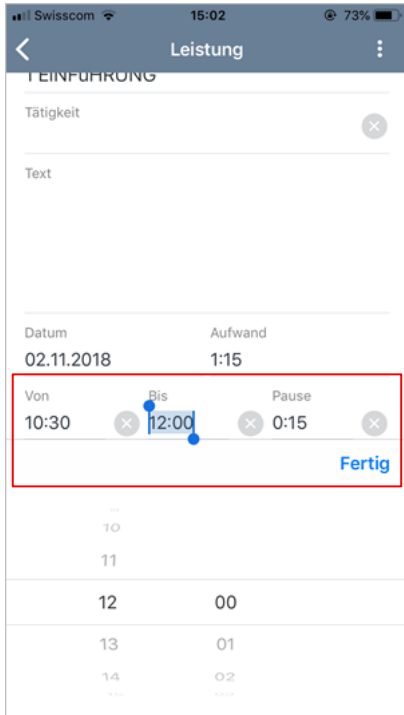


Abbildung 19: Eingabe der Uhrzeit für die Leistungserfassung

Vertec berechnet daraus automatisch den entsprechenden Aufwand.

Wird unter **Aufwand** ein Wert direkt gesetzt, werden die Werte **von**, **bis** und **Pause** wieder auf null gesetzt.

4.3 Drittprodukte / Datenschutzerklärung in Phone App

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Zur Einsicht in die Vertec Datenschutz-Erklärung sowie die verwendeten Drittprodukte via Phone App gibt es neu Links unter **Einstellungen**:



Abbildung 20: Links auf Drittprodukte und Datenschutzerklärung in der Vertec Phone App

4.4 Mobile Device Management Unterstützung

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.11

Neu unterstützt die Phone App die zentralisierte Mobilgeräteverwaltung (Mobile Device Management). So kann der Pfad zum Server (ServerUrl) per AppConfig hinterlegt werden, so dass der einzelne User diese nicht mehr manuell eingeben muss.

Die Umsetzung ist wie folgt:

- Beim Neustart der Phone App wird geprüft, ob eine AppConfig Einstellung für die ServerUrl vorhanden ist.
- Falls ja, wird die ServerUrl auf diesen Wert gesetzt.

Das hat folgende Auswirkungen:

- Ist eine ServerUrl via AppConfig gesetzt, wird beim Neustarten der App immer diese URL eingesetzt. Es erscheint dann kein Einstellungen Dialog, sondern direkt das Login.
- Wenn via AppConfig eine Änderung der ServerUrl gesetzt wird, dann hat das erst eine Auswirkung, wenn die Phone App neu gestartet wird.
- Wenn die ServerUrl manuell in der Phone App geändert wird, bleibt dieser Wert während der Laufzeit der App bestehen. Sobald die App neu gestartet wird, wird wieder automatisch der Wert aus AppConfig verwendet.

Weitere Informationen Weitere Informationen dazu finden Sie in der Online Knowledge Base unter www.vertec.com/kb/vertecphoneapp#mobile-device-management.

5 Rechnungsstellung

5.1 Unterstützung von Firmenadresse auf Zahlungstyp

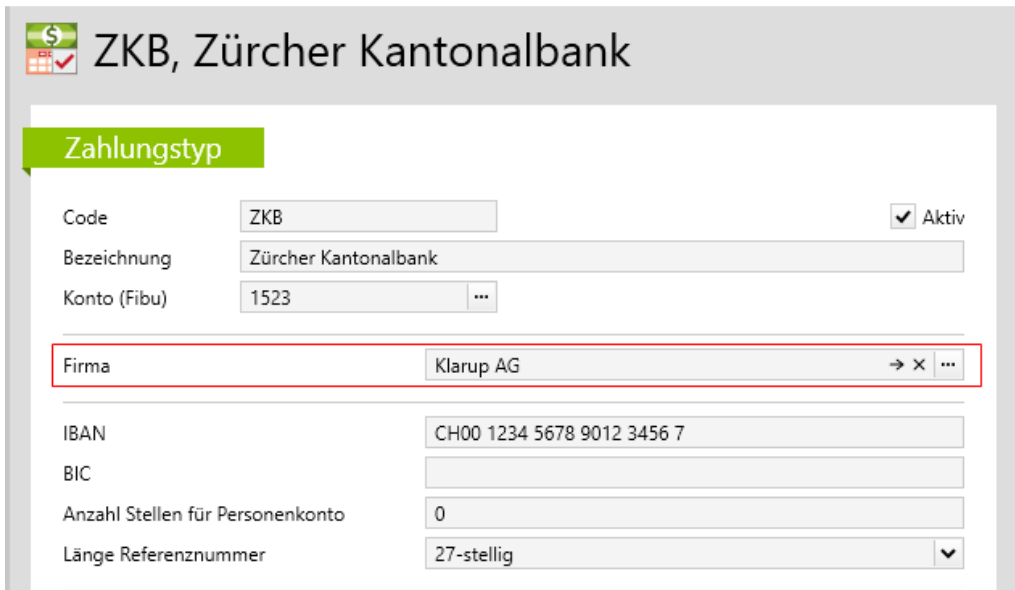
Linie: Expert | Module: Leistung & CRM | Version: 6.3

Auf dem Zahlungstyp wurden zwei neue Felder eingefügt: **CompanyAddress** und **BIC**.

Dies erlaubt die Verwaltung mehrerer Firmen innerhalb einer Vertec Installation.

CompanyAddress

Auf dem Zahlungstyp gibt es ein Feld **Firma**.



ZKB, Zürcher Kantonalbank

Zahlungstyp

Code: ZKB ☒ Aktiv

Bezeichnung: Zürcher Kantonalbank

Konto (Fibu): 1523 ...

Firma: Klarup AG → × ...

IBAN: CH00 1234 5678 9012 3456 7

BIC:

Anzahl Stellen für Personenkonto: 0

Länge Referenznummer: 27-stellig

Abbildung 21: Firma auf dem Zahlungstyp

Dieses übernimmt standardmässig die Firma aus der **Systemeinstellung Allgemein > Firma**. In diesem Fall wird Firma in schwarzer Schrift dargestellt.

Durch Klick auf den Button mit den drei Punkten kann auch eine alternative Firma auf diesem Zahlungstyp gesetzt werden. In diesem Fall erscheint die Schrift grün.

BIC

Das Feld für den BIC (Bank Identifier Code) ist ein Textfeld.

Rückwärtskompatibilität

Die Rechnungen sowie der QR-Code verwenden die CompanyAddress neu als Empfänger-Adresse. Da die CompanyAddress standardmässig die Firmenadresse aus den Systemeinstellungen nimmt, ergibt sich dadurch kein Unterschied zu vorher, bevor nicht explizit eine Firma auf dem Zahlungstyp gesetzt wird.

Das BIC Feld hat informativen Charakter und wird standardmässig in keinem Report verwendet.

6 Bearbeiter

6.1 Stichwort-Ordner Recht

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3.

Das bestehende **Ordner-Links Administrator** Recht wurde in ein neues **Stichwort Ordner Administrator** Recht umgewandelt und wie folgt verändert:

- Benutzer mit dem Stichwort Ordner Recht können Unterordner von Stichwort-Ordnern anlegen, löschen und bearbeiten.
- Ansonsten können Benutzer Ordner nur noch modifizieren, wenn sie sich in ihrem persönlichen Ordner befinden.
- Diese beiden Punkte beziehen sich auch auf das Modifizieren der Listeneinstellungen.
- Einträge zu Ordnern hinzufügen dürfen neu alle Benutzer.

Die bestehende Benutzergruppe **Ordner/Links Administratoren** wurde in **Stichwort Ordner Administratoren** umbenannt.

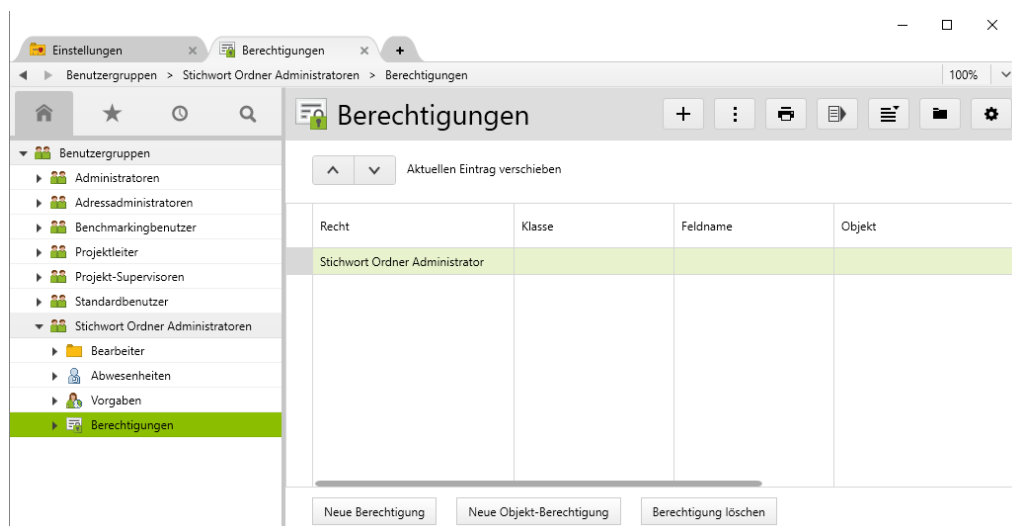


Abbildung 22: Ordner-Links Administratoren heißen neu Stichwort Ordner Administratoren

Das **Ordner-Links User** Recht wurde ersatzlos gelöscht, zusammen mit den entsprechenden Rechte-Zuordnungen.

Rückwärtskompatibilität

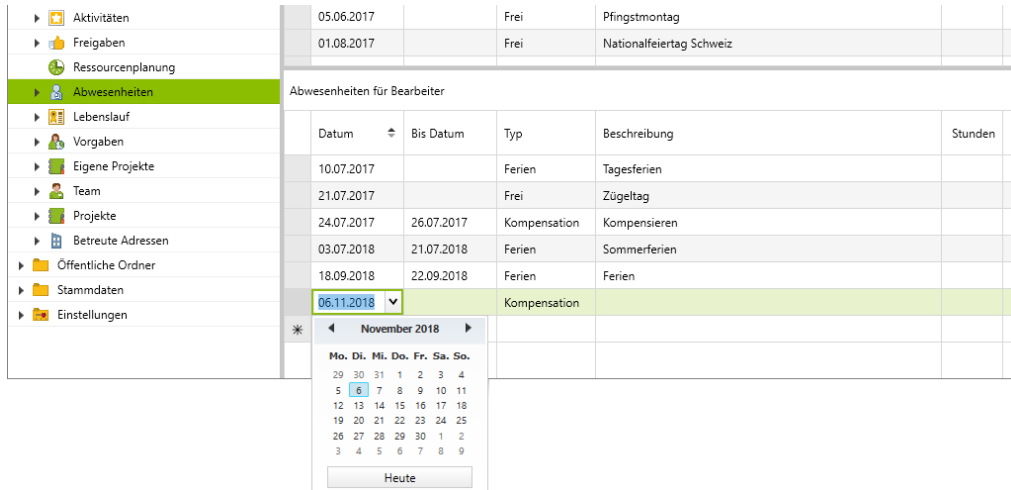
Das Anlegen, Bearbeiten und Löschen von Unterordnern sowie das Modifizieren der Listeneinstellungen kann neu auch von ehemaligen Ordner-Links Administratoren – jetzt Stichwort Ordner Administratoren – nur noch bei Stichwort Ordnern sowie in den persönlichen Ordnern vorgenommen werden.

Das Anlegen von Ordner-Strukturen und Anpassen der Listeneinstellungen sind Einstellungsdaten, welche nur noch von Benutzern mit Administratoren-Rechten gesetzt werden können. Je nachdem müssen Sie dafür in Ihrem Vertec die Rechte neu vergeben.

6.2 Erfassung von Abwesenheiten neu mit DatePicker

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Bei der Erfassung von Abwesenheiten kann das Datum neu standardmässig mittels DatePicker ausgewählt werden.



Datum	Bis Datum	Typ	Beschreibung	Stunden
05.06.2017		Frei	Pfingstmontag	
01.08.2017		Frei	Nationalfeiertag Schweiz	
Abwesenheiten für Bearbeiter				
10.07.2017		Ferien	Tagesferien	
21.07.2017		Frei	Zugeltag	
24.07.2017	26.07.2017	Kompensation	Kompensieren	
03.07.2018	21.07.2018	Ferien	Sommerferien	
18.09.2018	22.09.2018	Ferien	Ferien	
06.11.2018		Kompensation		

Abbildung 23: Das Datum der Abwesenheit wird mittels DatePicker gesetzt

Nach wie vor ist auch die manuelle Eingabe im Feld möglich.

7 Customizing

7.1 Auswahl-Suchfeld in SQL-Ordner Suchdialogen

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

In SQL-Abfragen gibt es einen neuen Feld-Typ **Auswahl**:

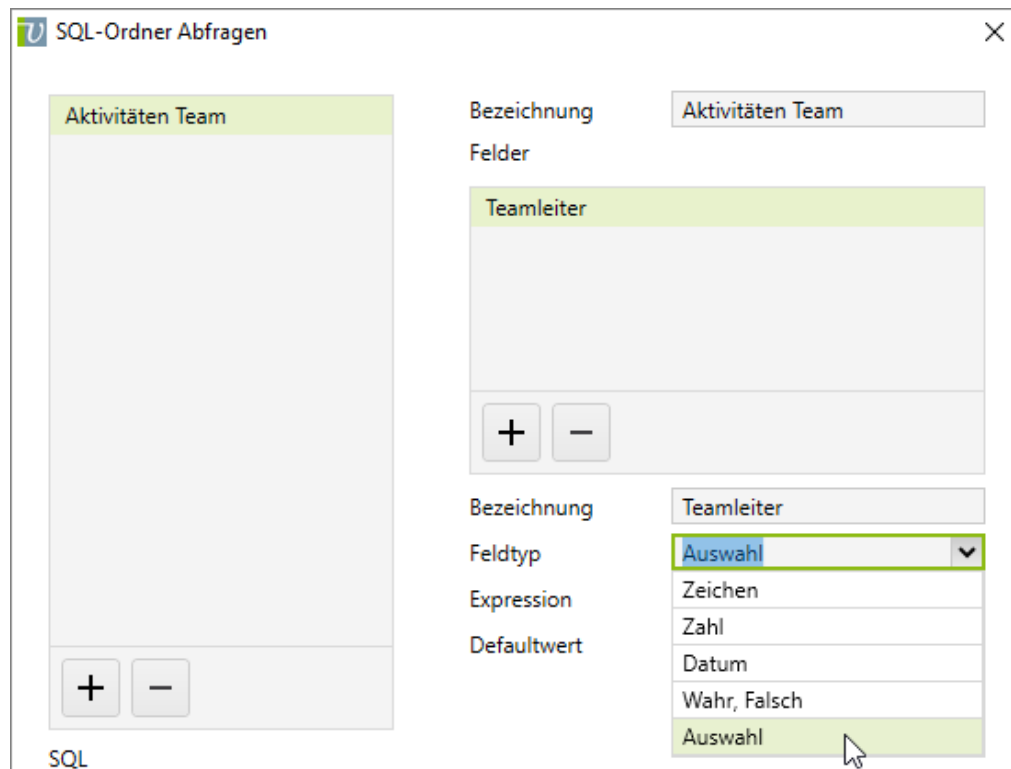


Abbildung 24: Feldtyp Auswahl

Sobald dieser Typ gesetzt ist, wird gleich darunter ein Feld für die Expression eingeblendet.

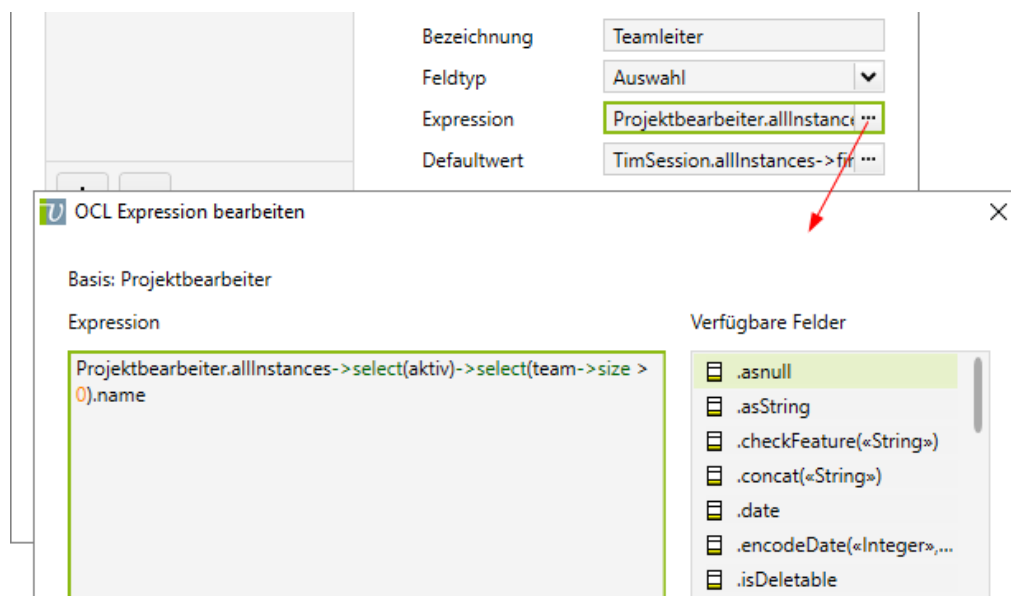
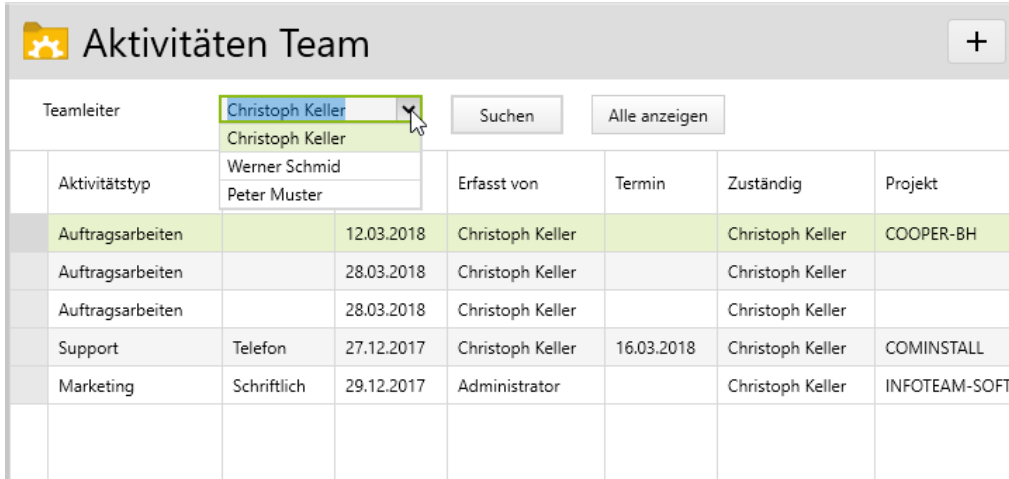


Abbildung 25: Die Expression für die Auswahl

Die Expression muss als Resultat eine Liste von Strings zurückgeben.

Auf der Oberfläche wird dann ein Auswahlfeld angezeigt, aus welchem der gewünschte Wert ausgewählt werden kann.

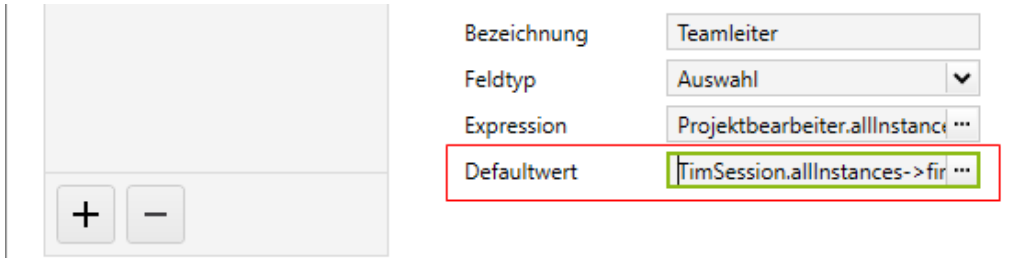


Aktivitätstyp	Erfasst von	Termin	Zuständig	Projekt
Auftragsarbeiten	Christoph Keller	12.03.2018	Christoph Keller	COOPER-BH
Auftragsarbeiten	Christoph Keller	28.03.2018	Christoph Keller	
Auftragsarbeiten	Christoph Keller	28.03.2018	Christoph Keller	
Support	Christoph Keller	27.12.2017	16.03.2018	COMINSTALL
Marketing	Administrator	29.12.2017	Christoph Keller	INFOTEAM-SOFT

Abbildung 26: Der gewünschte Wert kann aus dem Auswahlfeld ausgewählt werden

Als Standard wird der unter Defaultwert angegebene Eintrag angezeigt, falls einer definiert ist.

Falls ein Defaultwert definiert ist, wird standardmässig dieser angezeigt.



Bezeichnung: Teamleiter

Feldtyp: Auswahl

Expression: Projektbearbeiter.allInstances

Defaultwert: TimSession.allInstances->fir

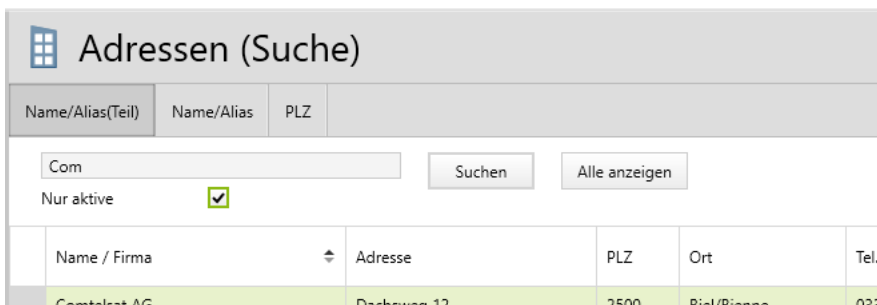
Abbildung 27: Ein Standardwert wird definiert

Falls der angegebene Defaultwert nicht in der Auswahlliste vorhanden ist oder es keinen Defaultwert hat, wird der erste Eintrag der Auswahlliste genommen.

7.2 Inaktive Einträge anzeigen bei SQL-Ordern konfigurierbar

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

Bei SQL-Ordern für Adresseinträge und Projekte wird, falls ein Suchdialog angezeigt wird, eine Option **Nur aktive** angezeigt.



Name / Firma	Adresse	PLZ	Ort	Tel.
Comalast AG	Dachau 12	8500	Biel/Bienne	033

Abbildung 28: Nur aktive Adressen werden durchsucht

Ob das Häkchen standardmässig gesetzt ist oder nicht, kann mit der Systemeinstellung Allgemein > **Deaktivierte Einträge in Such-Dialogen berücksichtigen** gesteuert werden.

Neu kann diese Option auch ganz ausgeblendet werden. Im Abfrage-Definitions-Dialog gibt es dafür eine Option **'Nur aktive' ausblenden**:

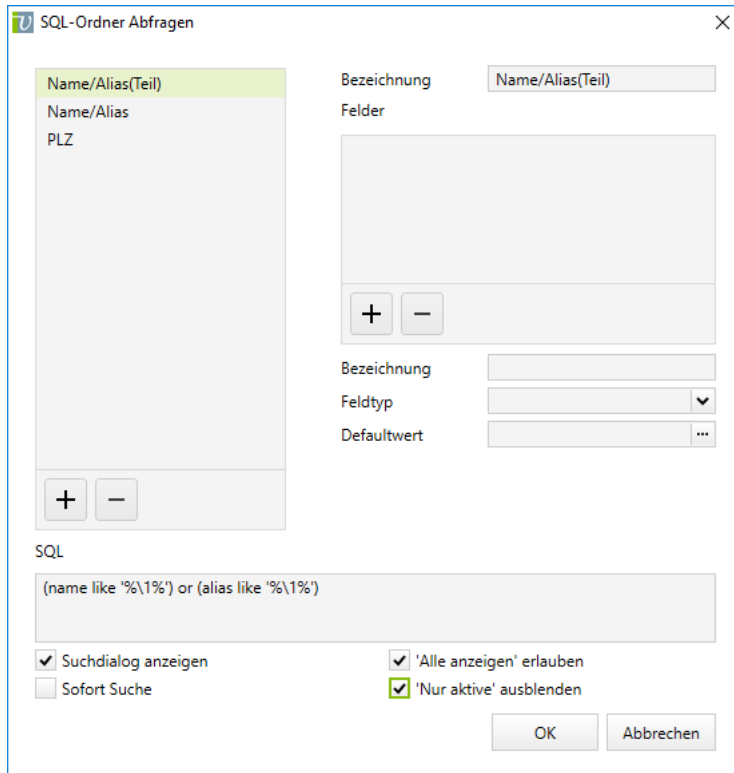


Abbildung 29: Die Option 'Nur aktive' wird ausgeblendet

Dies hat folgende Auswirkungen:

- Die Checkbox **Nur aktive** wird auf der Oberfläche nicht mehr angezeigt.
- Die Suche erfolgt bei ausgeblendeter Checkbox immer in allen Einträgen, also in aktiven und in inaktiven. Ist die Option ausgeblendet, wird die oben genannte Systemeinstellung **Deaktivierte Einträge in Such-Dialogen berücksichtigen** nicht beachtet.

Rückwärtskompatibilität

Die neue Option ist standardmässig nicht gesetzt. Das bedeutet, dass es bei allen bestehenden Suchkonfigurationen keinen Unterschied im Verhalten gibt.

7.3 OCL Null Operator

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.4

Der OCL Null Operator ermöglicht es, für beliebige Attribute-Typen Null-Werte zu erhalten.

Dies ist nützlich, um bei Zahlenwerten statt 0 nichts darzustellen, ohne dafür die ganze Expression in einen String umwandeln zu müssen. So können Expressions der Art:

```
if usespesen then spesenint.asstring else '' endif
```

durch folgende Expression ersetzt werden:

```
if usespesen then spesenint else spesenint.asnull endif
```

Da OCL eine statisch und streng typisierte Sprache ist, müssen auch ein Null-Wert einen Typ haben, damit die Gültigkeit der Expression festgestellt werden kann. Deshalb wurde ein `.asnull` Operator auf Instanzen und `.null` auf Typen eingeführt:

- `<typ>.null` erstellt aus einem OCL-Typ einen Null-Value. Unter Verwendung des null Operators auf Type würde obige Expression wie folgt lauten:

```
if usespesen then spesenint else VtcCurrency.null endif.
```

- `<value>.asnull` erstellt aufgrund eines bestehenden Wertes (Member-Referenz) einen null Value desselben Typs, ohne dass man den korrekten Namen des Typen wissen muss:

```
if usespesen then spesenint else spesenint.asnull endif.
```

7.4 Initiale Anzeige bei ReferenceBox konfigurierbar

Linie: Expert | Module: Leistung & CRM | Version 6.2.0.3

Bei der **ProjectReferenceBox**, dem Steuerelement zum Auswählen eines Projektes via SQL Suche, werden im Suchdialog standardmässig die Projekte angezeigt, auf denen im letzten Monat Leistungen erfasst wurden.

Dies kann bei Installationen mit sehr vielen Mitarbeitern und Projekten zu einem Performanceproblem werden.

Deshalb wurde das Property **ShowInitialContent** eingeführt, welches die Abfrage einer initialen Ergebnisliste ein- oder ausschaltbar macht.

Dieses ist standardmässig **True**, damit sich keine Änderung des bisherigen Verhaltens ergibt.

Wird es auf **False** gesetzt, dann wird im Suchdialog keine Favoritenliste mehr angezeigt.

Beispiel

Möchten Sie zum Beispiel in den Listeneinstellungen eine Projekt-Suche ohne Favoriten-Anzeige einfügen (anstelle des Steuerelements **edtProjekt**), lautet der Code unter Steuerelement XML wie folgt:

```
<ProjectReferenceBox ShowInitialContent="False" />
```

7.5 Stichworte mittels Eintrag Id einblenden

Linie: Expert | Module: Leistung & CRM | Version 6.2.0.9

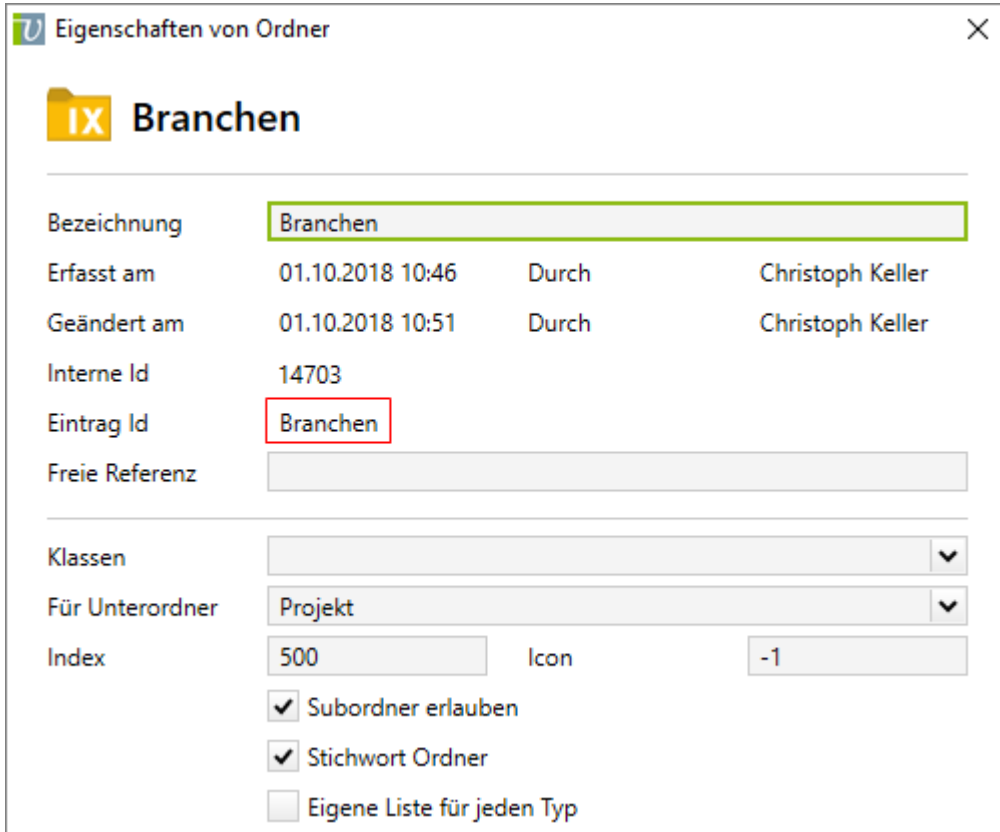
Einzelne Stichwörter können in den Seiten platziert werden mit dem Element **KeywordGroup**. Dieses stellt eine Gruppe mit der Bezeichnung des Stichwort-Ordnern und einer Auswahl je nach Art mit Checkboxes oder einer Combobox dar.

Zur Identifikation des Stichwortordners musste bisher als Eigenschaft **FolderId** die Interne Id des Ordners angegeben werden.

Neu kann alternativ dazu auch die Eigenschaft **FolderEntryId** verwendet werden, welche die Eintrag Id des Ordners enthält.

```
<KeywordGroup FolderEntryId="Branchen" PlaceAfter="InvoicePeriod" />
```

Die Eintrag Id des Ordners finden Sie in den Ordneigenschaften:



Eigenschaften von Ordner

IX Branchen

Bezeichnung: Branchen

Erfasst am: 01.10.2018 10:46 Durch: Christoph Keller

Geändert am: 01.10.2018 10:51 Durch: Christoph Keller

Interne Id: 14703

Eintrag Id: Branchen

Freie Referenz:

Klassen:

Für Unterordner: Projekt

Index: 500 Icon: -1

☒ Subordner erlauben

☒ Stichwort Ordner

☐ Eigene Liste für jeden Typ

Abbildung 30: Als FolderEntryId wird die Eintrag Id des Stichwortordners angegeben

Weitere Informationen Weitere Informationen zum Page Customizing finden Sie in der Online Knowledge Base unter www.vertec.com/kb/seiten.

7.6 Syntax Highlighting in XML Editor erweitert

Linie: Expert | Module: Leistung & CRM | Version 6.3

Das Syntax Highlighting von XML Code wurde um Tag-Matching erweitert.

Das bedeutet, dass Anfang und Ende von Tags angezeigt werden können, indem der Cursor darauf positioniert wird:



```

5
6 <Group Name="GeneralGroup" Orientation="Vertical">
7   <Group Name="ProjectCodeActiveGroup">
8     <TextBox Name="ProjectCode" Label="{Translate 'Project-code'}"
9       ValueExpression="code" />
10    <CheckBox Name="Active" Text="{Translate 'Active'}" ValueExpression="aktiv"
11      HorizontalAlignment="Right" ShowLabel="False" />
12  </Group>
13  <AddressReferenceBox Name="Customer" Label="{Translate 'Client'}"
14    ValueExpression="kunde" ShowRemoveButton="True" />
15  <TextArea Name="Description" Label="{Translate 'Description'}"
16    ValueExpression="beschrieb" Lines="2" />
17  <TextArea Name="Regarding" Label="{Translate 'Regarding'}"
18    ValueExpression="betreffend" Lines="2" />
19 </Group>

```

Abbildung 31: Vom Tag <Group> werden Anfang und Ende angezeigt

Dadurch wird der Code lesbarer und übersichtlicher.

7.7 TextArea Info Appearance

Linie: Expert | Module: Leistung & CRM | Version 6.3

Die TextArea wurde so erweitert, dass mehrzeiliger Text dargestellt werden kann, ohne dass dieser bearbeitbar ist oder so aussieht.

Dafür kann das Property **Appearance** neu `= "Info"` sein.

Beispiel

```
<TextArea Label="Text Info" Lines="3" Appearance="Info" Text="Dies  
ist ein Text mit 3 Zeilen. Dieser wird so angezeigt, dass es wie ein  
Info-Text aussieht. Er kann nicht bearbeitet werden und dient nur zur  
Darstellung von Information." />
```

Das sieht auf der Oberfläche so aus:

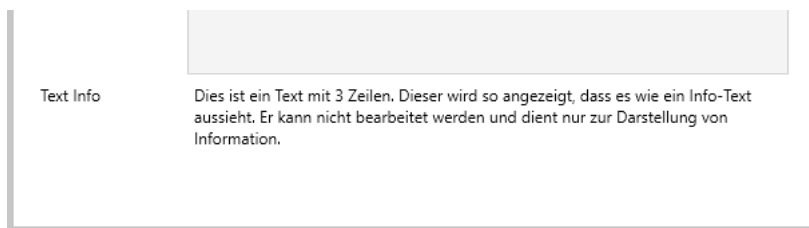


Abbildung 32: Ein Text wird mit Appearance="Info" angezeigt

Weitere Informationen Weitere Informationen zum Customizing der Vertec Benutzeroberfläche finden Sie in unserer Online Knowledge Base unter www.vertec.com/kb/grundlagen.

8 Scripting

8.1 Neuer Script Editor

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Für die Version 6.3 wurde ein neuer Python Script Editor implementiert, der auch in der Cloud App und Web App verwendet werden kann.

Der neue Script-Editor wird als eigene Seite in einem neuen Tab angezeigt.

Über Menü **Einstellungen > Script Editor** aufgerufen, erscheint er wie folgt:

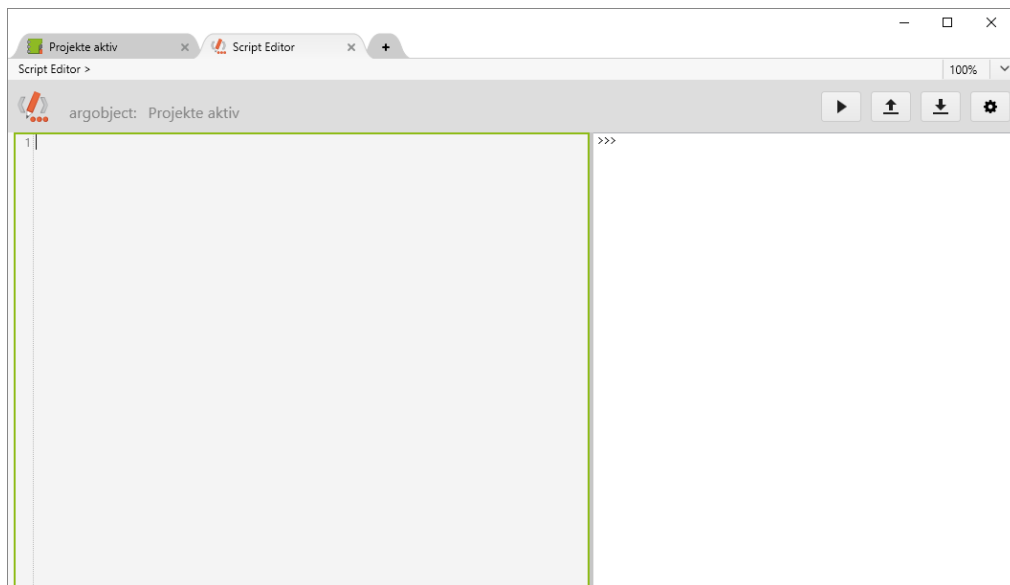


Abbildung 33: Der neue Script Editor öffnet sich in einem neuen Tab

Im linken Teil des Fensters wird der Code angezeigt bzw. bearbeitet. Es muss sich dabei um gültigen Python Code handeln.

Der rechte Teil des Fensters ist das Ausgabefenster.

Als separate Anwendung gibt es neu zusätzlich die Python Konsole, welche nur das Ausgabefenster anzeigt. Die Python Konsole ist im Kapitel 8.2 genau beschrieben.

Es gibt folgende 4 Buttons:

	Ausführen: Führt den im linken Teil des Fensters angezeigten Code aus.
	Datei öffnen: Damit kann ein Script vom Filesystem im Script Editor geöffnet werden
	Datei speichern: Speichert den im linken Teil des Fensters angezeigten Code in die entsprechende Datei. Handelt es sich um neuen Code, kommt ein Speicherdialog.
	Das Vertec Einstellungen Menü

Der Script Editor verwendet als **argobject** (aktuelles Objekt) das zuletzt ausgewählte Vertec Objekt in einem der anderen Tabs. Das jeweilige argobject wird im oberen Teil des Script Editor Fensters angezeigt.

Der Script Editor kann auch über den Button **Script Editor** in einem Scripteintrag geöffnet werden. In diesem Fall enthält er den Scripttext des entsprechenden Scripteintrags.

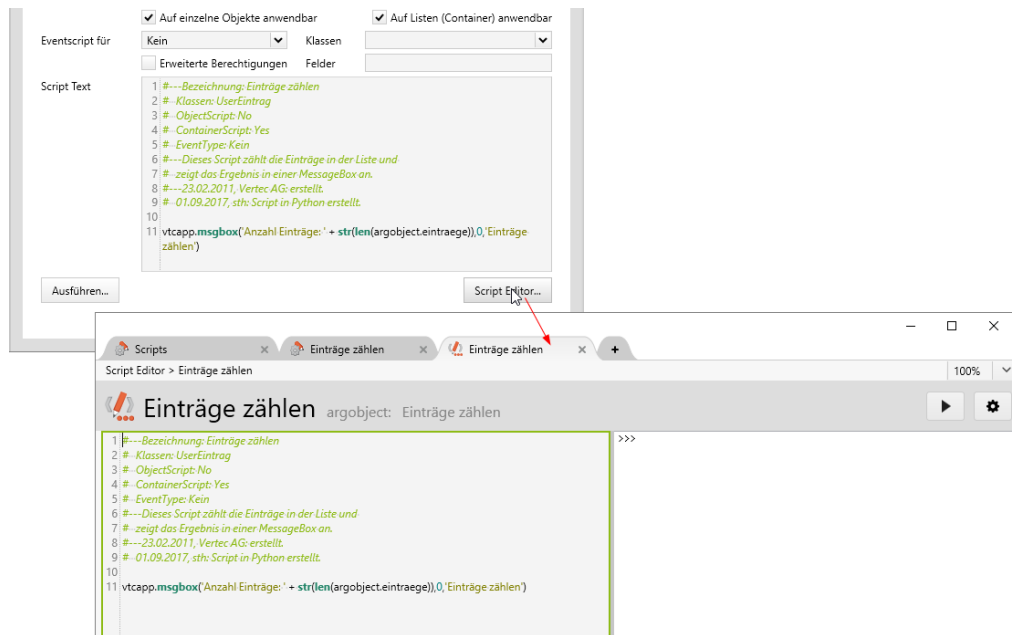


Abbildung 34: Ein registriertes Script wird im Script Editor geöffnet

Der Script Editor ist dann mit dem Scripteintrag direkt verknüpft. Das heisst: wird hier Code angepasst, ändert sich der Scripttext im Scripteintrag ebenfalls.

Mehrere Script Editoren Es können beliebig viele Script Editor Fenster parallel geöffnet werden. Diese funktionieren unabhängig voneinander, zeigen jedoch alle jeweils als argobject das zuletzt ausgewählte Vertec Objekt.

Der neue Script Editor ersetzt den bisherigen Python Script Editor.

Der bisher ebenfalls verfügbare VB Script Editor wurde auf diese Version hin entfernt. Siehe dazu den Hinweis in Kapitel 2.2.

8.2 Python Konsole

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Ein wichtiger Teil des Script Editors (siehe 8.1) ist die Möglichkeit, auf dem aktuellen Objekt eine Python Anweisung auszuführen und das Ergebnis zu sehen, also die Konsolen-Funktionalität.

Um dies zu erleichtern, wurde eine rasch und unkompliziert aufrufbare Python Konsole geschaffen, die für Administrator-Benutzer verfügbar ist. Die Konsole kann über das Menü **Einstellungen** sowie über das Tastaturkürzel **F3** ein- und ausgeblendet werden:

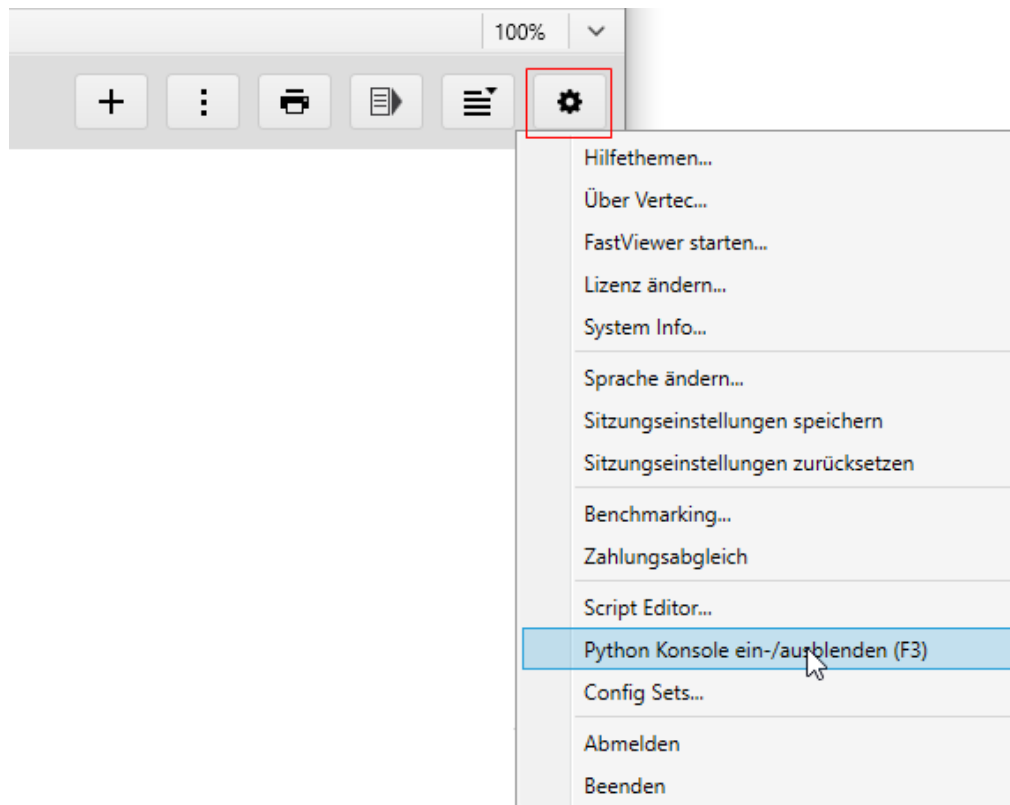


Abbildung 35: Einstellungen > Python Konsole ein-/ausblenden

Am unteren Rand des Vertec-Fensters wird die Konsole angezeigt:

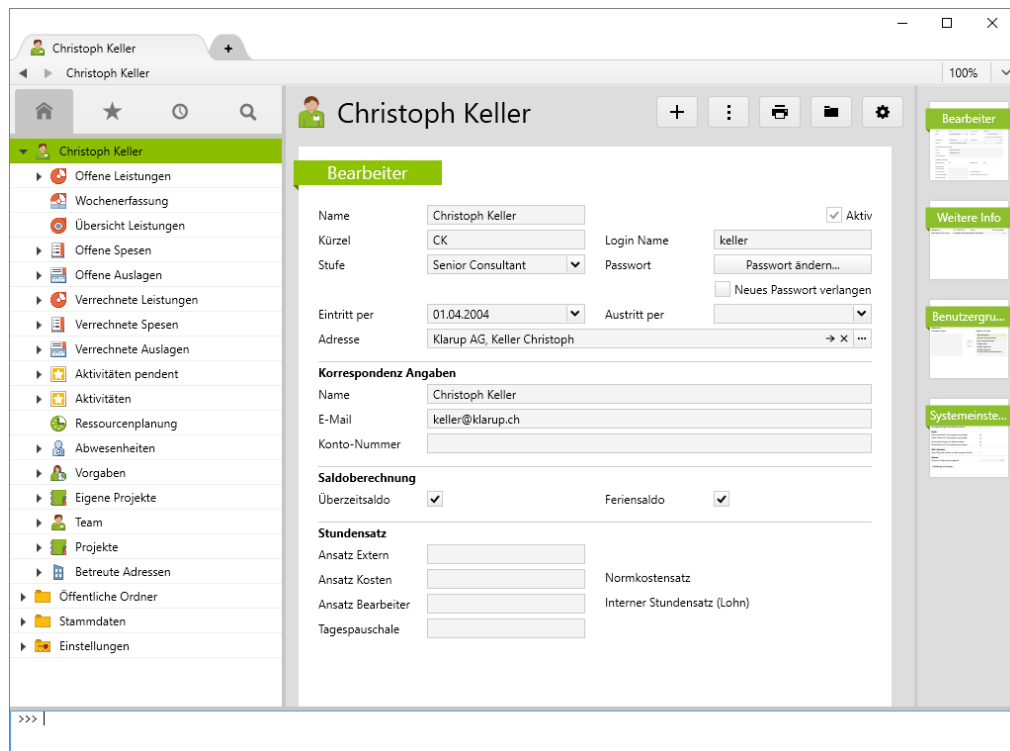


Abbildung 36: Die Python Konsole wird angezeigt

Hier können Python Anweisungen ausgeführt werden.

Die Höhe der Konsole kann mit der Maus verändert werden:

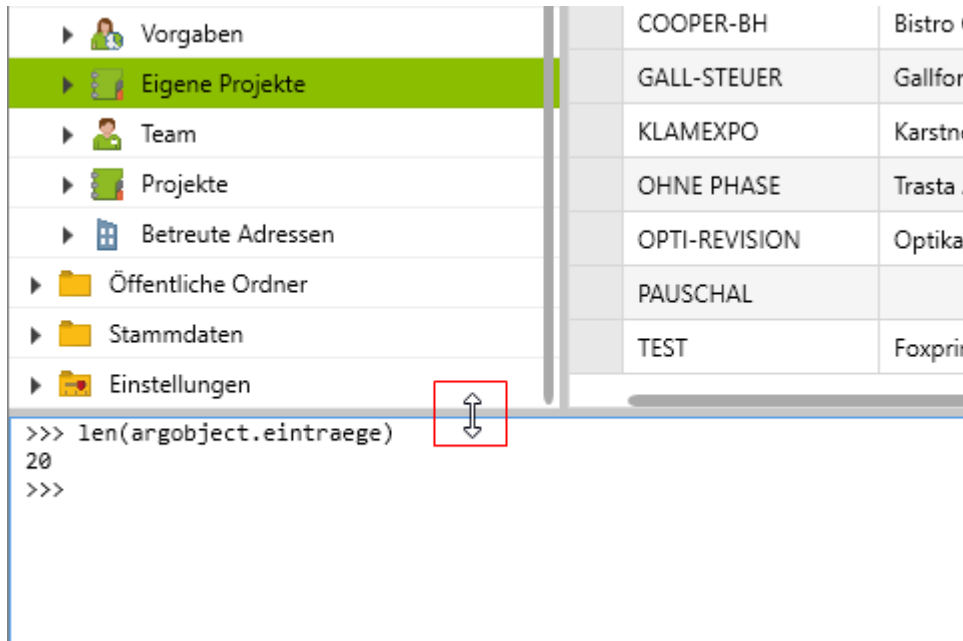


Abbildung 37: Die Höhe der Konsole wird verändert

Mit Rechtsklick in die Python-Konsole erscheint ein Context-Menü mit Option zum Löschen des Konsolen-Inhalts.

8.3 Vertec-Listen in Python

Linie: Expert | Module: Leistung & CRM | Version 6.2.0.7

Neu geben alle Python Methoden, welche Listen zurückgeben, direkt eine Vertec-Liste zurück und keine Python-Liste mehr. Die aus den Methoden

- evalocl()
- getwithsql()
- getmemberwithsql()

resultierenden Listen sind also direkt Vertec-Listen und müssen nicht mehr mit `vtcapp.createlist()` erzeugt werden.

Vorher:

```
liste = vtcapp.createlist("Projekt", vtcapp.evalocl("projekt->select (code.sqllike ('A%'))"))
```

Neu:

```
liste = vtcapp.evalocl("projekt->select (code.sqllike ('A%'))")
```

Methoden auf Vertec-Listen Eine Auflistung der Methoden auf Vertec-Listen finden Sie im Artikel [Vertec Python Interfaces](https://www.vertec.com/kb/pythoninterfaces#methoden-auf-vertec-listen) in der Online Knowledge Base unter www.vertec.com/kb/pythoninterfaces#methoden-auf-vertec-listen.

Rückwärtskompatibilität

Vertec- und Python-Listen unterscheiden sich in folgenden Punkten:

Der + Operator ist nicht auf Vertec-Listen anwendbar. Code der Art

```
projektIds = argobject.eval("eintraege").idstring()
leistungen = vtcapp.getwithsql("OffeneLeistung", "projekt IN (%s)" % projektIds, "") \
    + vtcapp.getwithsql("VerrechneteLeistung", "projekt IN (%s)" % projektIds, "")
```

funktioniert nicht. Es erscheint eine Fehlermeldung: `TypeError: can only concatenate list (not "BoldListAdapter") to list`.

Stattdessen muss `extend()` oder `+=` verwendet werden:

```
leistungen = vtcapp.getwithsql("OffeneLeistung", "projekt IN (%s)" % projektIds, "")
leistungen.extend(vtcapp.getwithsql("VerrechneteLeistung", "projekt IN (%s)" % projektIds, ""))
```

bzw.

```
leistungen = vtcapp.getwithsql("OffeneLeistung", "projekt IN (%s)" % projektIds, "")
leistungen += vtcapp.getwithsql("VerrechneteLeistung", "projekt IN (%s)" % projektIds, "")
```

Aus Vertec-Listen können zur Herstellung der Kompatibilität Python-Listen erstellt werden durch `list(vertecliste)`.

Man muss beachten, dass Vertec-Listen "live" sind. Das heisst, bei Aktionen, die sich auf die Logik der Listenberechnung beziehen, kann sich diese verändern, wie in folgenden Beispielen gezeigt:

Bei `remove()` und `delete()` ist die Liste nach jeder Iteration um eines kürzer. Am Schluss ist nur die Hälfte der Liste gelöscht. Um alle Objekte einer Liste zu löschen, sollte man sie deshalb am besten in eine Python-Liste umwandeln:

```
for l in list(argobject.offeneleistungen):
    l.delete()
```

Im folgenden Fall handelt es sich um eine Liste von offenen Leistungen:

```
leistungen = phase.evalocl("offeneleistungen")
for l in leistungen:
    l.makeverrechnet()
```

Werden sie der Reihe nach verrechnet, verschwinden sie automatisch aus der Liste. Der Effekt ist derselbe wie oben: Es wird nur die Hälfte der Leistungen verrechnet. Stattdessen in eine Python-Liste umwandeln:

```
for l in list(phase.evalocl("offeneleistungen")):
    l.makeverrechnet()
```

'count', 'pop', 'reverse' und 'sort' gibt es nur auf Python-Listen. Wenn Sie diese Methoden verwenden, müssten Sie diese umschreiben bzw. mit `list(vertecliste)` aus der Vertec-Liste wieder eine Python-Liste machen.

8.4 Neue Python Module im Restricted Scripting Modus

Linie: Expert | Module: Leistung & CRM | Version 6.2.0.8/6.2.0.11

Die folgenden Python Module sind neu für die Verwendung im Restricted Scripting Modus verfügbar:

Ab Version 6.2.0.8

- requests

Ab Version 6.2.0.11

- string
- re
- json
- urlparse
- email
- smtplib
- base64

Weitere Informationen Weitere Informationen zum Thema **Restrict Scripting** finden Sie in unserer Online Knowledge Base unter www.vertec.com/kb/restrict-scripting.

9 Config Sets

Bei den Config Sets, welche mit Vertec 6.2 eingeführt wurden, wurden noch einige Veränderungen vorgenommen. Diese sind hier beschrieben.

Rückwärtskompatibilität

Es wurde nicht auf die Rückwärtskompatibilität geachtet. Das bedeutet, dass der aktuelle Stand massgebend ist. Bisher erstellte Config Sets können noch laufen; müssen jedoch mit dem aktuellen Stand getestet und allenfalls neu erstellt werden.

9.1 Ordner exportieren Subordner nicht mehr automatisch

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.9

Bisher wurden Subordner eines Ordners immer implizit in das Configset übernommen. Dadurch war es unmöglich, Ordner-Hierarchien in verschiedene Config-Sets aufzuteilen.

Neu müssen Subordner explizit ins Config-Set aufgenommen werden, sonst werden sie nicht übernommen.

9.2 Configset-reference entfernt

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.9

Das `configset-reference` Element wurde entfernt. Somit können andere Config Sets nicht mehr referenziert werden.

Bestehen bleiben die Elemente `entryid-reference` sowie `ocl-reference`.

Weitere Informationen Weitere Informationen zu den Referenzen finden Sie in der Online Knowledge Base unter www.vertec.com/kb/config-set-xml#references.

9.3 Configset-requirements entfernt

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.9

Die `configset-requirements` wurden ebenfalls entfernt.

Um sicherzustellen, dass bestimmte Objekte in Vertec vorhanden sind, können weiterhin die `ocl-requirements` verwendet werden.

Weitere Informationen Weitere Informationen zu den Requirements finden Sie in der Online Knowledge Base unter www.vertec.com/kb/config-set-xml#requirements.

9.4 Listeneinstellungen im Config Set aufnehmen

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.9

Wird ein Ordner in ein Config Set aufgenommen, werden der Ordner sowie seine Listeneinstellungen übernommen und im Zielsystem neu angelegt.

Neu können auch nur die Listeneinstellungen im Config Set aufgenommen werden.

Wird ein Ordner im Config Set Builder hinzugefügt, kann mit der rechten Maustaste drauf geklickt werden. Im Kontextmenü erscheint der Menüpunkt `Listeneinstellungen anstelle von Ordner verwenden`.

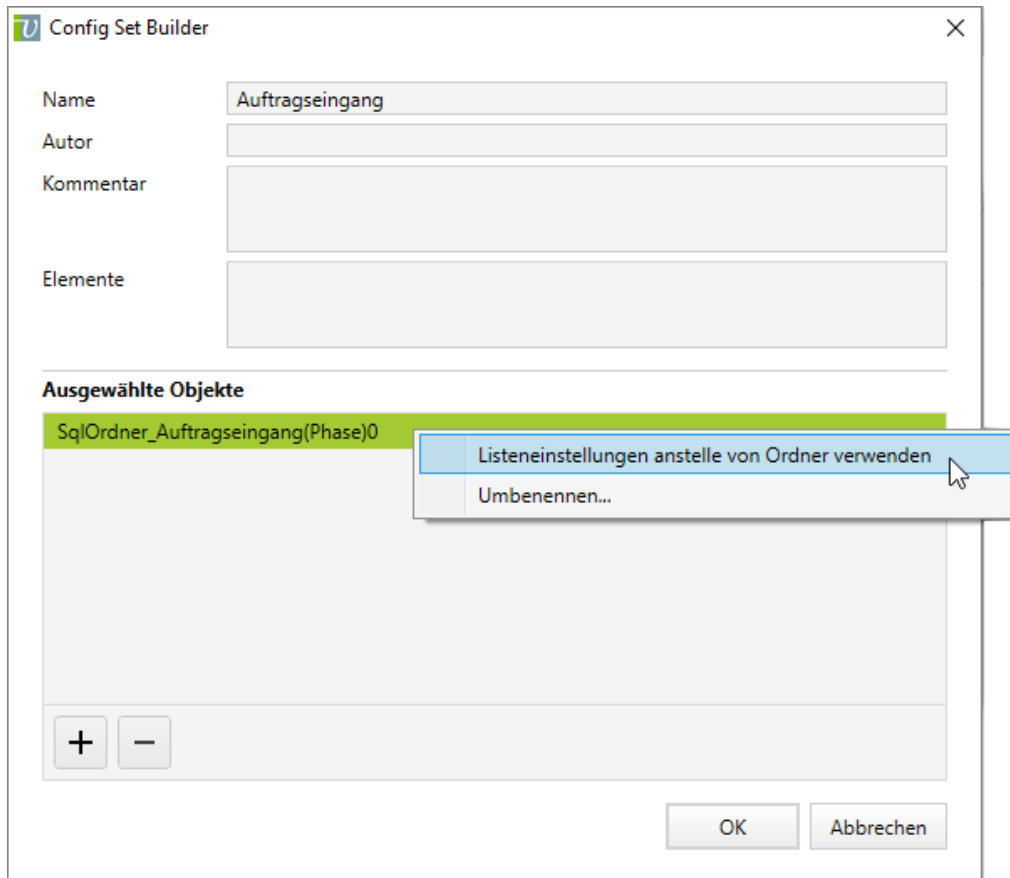


Abbildung 38: Listeneinstellungen des Ordners ins Config Set übernehmen

Klickt man darauf, werden nur die Listeneinstellungen übernommen. Die Zeile sieht dann wie folgt aus:

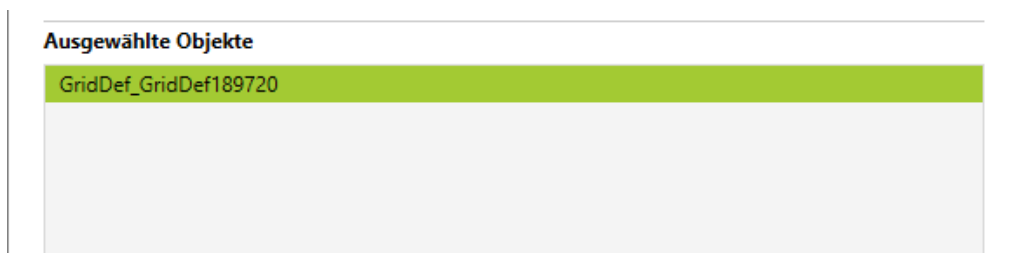


Abbildung 39: Die Listeneinstellungen wurden hinzugefügt

Das bedeutet, dass NUR die Listeneinstellungen, nicht aber der Ordner exportiert werden. In diesem Fall muss der entsprechende Ordner via Eintrag Id bereits vorhanden sein im Zielsystem.

9.5 Dateigrösse für Import von Config Sets erhöht

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.7

Die maximale Grösse für Config Set Files zum Import in Vertec wurde von 1 MB auf 50 MB erhöht.

9.6 Bessere Fehlermeldungen bei Config-Sets

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.9

Bei der Verwendung von Config Sets wird nun detaillierter gemeldet, was fehlt, falls eine Referenz auf ein nicht vorhandenes Objekt zeigt. Dies sowohl beim Import als auch bei der Erstellung von Config Sets. Beispiel:

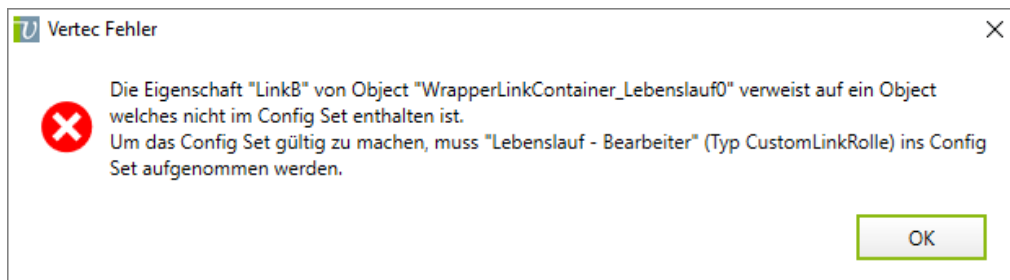


Abbildung 40: Detaillierte Fehlermeldung beim Erstellen eines Config Sets

9.7 Speichern unter Dialog beim Erstellen von Config Sets

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.7

Beim Exportieren von Config Sets kommt neu ein Speicher-Dialog, wo angegeben werden kann, wo das Config Set gespeichert werden soll (bisher wurde das File automatisch im Temp-Verzeichnis gespeichert und direkt geöffnet).

- **Web App**: Bei der Web App fragt der Dialog, ob man öffnen oder speichern (unter Downloads) möchte.
- **Cloud App** und **Desktop App** schlagen den Speicherpfad der letzten Speicherung vor.

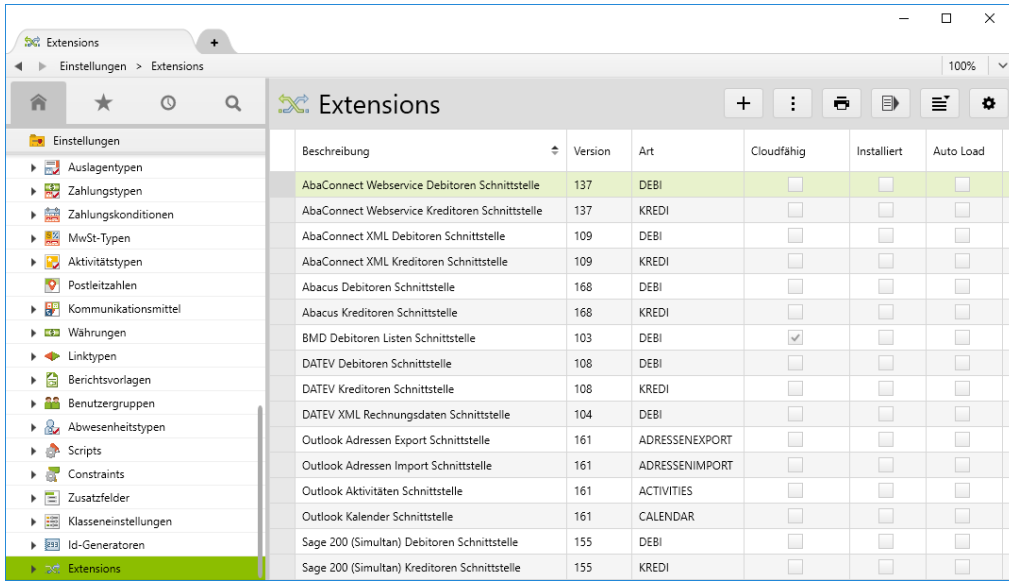
10 Schnittstellen

10.1 Extensions als Vertec Objekte

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

Das Extensions-Handling wurde auf die Version 6.3 komplett umgestellt. Neu sind die Extensions eigenständige Vertec Objekte (vom Typ **UserEintrag**). Das ermöglicht neue Einsatzmöglichkeiten für Extensions, zum Beispiel im Vertec Cloud-Abo.

Im Ordner **Einstellungen** gibt es dafür einen Ordner **Extensions**:



Beschreibung	Version	Art	Cloudfähig	Installiert	Auto Load
AbaConnect Webservice Debitoren Schnittstelle	137	DEBI	☐	☐	☐
AbaConnect Webservice Kreditoren Schnittstelle	137	KREDI	☐	☐	☐
AbaConnect XML Debitoren Schnittstelle	109	DEBI	☐	☐	☐
AbaConnect XML Kreditoren Schnittstelle	109	KREDI	☐	☐	☐
Abacus Debitoren Schnittstelle	168	DEBI	☐	☐	☐
Abacus Kreditoren Schnittstelle	168	KREDI	☐	☐	☐
BMD Debitoren Listen Schnittstelle	103	DEBI	☒	☐	☐
DATEV Debitoren Schnittstelle	108	DEBI	☐	☐	☐
DATEV Kreditoren Schnittstelle	108	KREDI	☐	☐	☐
DATEV XML Rechnungsdaten Schnittstelle	104	DEBI	☐	☐	☐
Outlook Adressen Export Schnittstelle	161	ADRESSEEXPORT	☐	☐	☐
Outlook Adressen Import Schnittstelle	161	ADRESSEIMPORT	☐	☐	☐
Outlook Aktivitäten Schnittstelle	161	ACTIVITIES	☐	☐	☐
Outlook Kalender Schnittstelle	161	CALENDAR	☐	☐	☐
Sage 200 (Simultan) Debitoren Schnittstelle	155	DEBI	☐	☐	☐
Sage 200 (Simultan) Kreditoren Schnittstelle	155	KREDI	☐	☐	☐

Abbildung 41: Liste der Extensions in den Einstellungen

Neu ist für alle verfügbaren Extensions ein Extension-Objekt vorhanden, ob die Extension installiert ist oder nicht, wird durch die **Installiert** Eigenschaft bestimmt.

Durch Doppelklick öffnet sich die Einzelmaske:

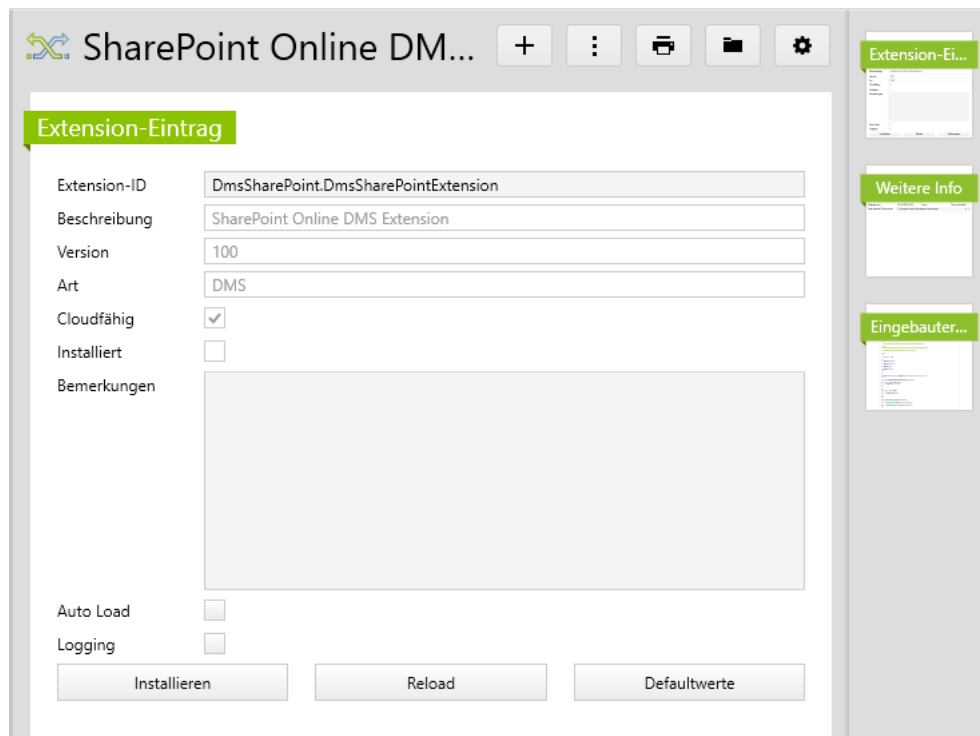


Abbildung 42: Einzelfenster einer Extension

Schnittstelle installieren

Um die Schnittstelle zu installieren, klicken Sie auf den Button **Installieren**.

Der Button wechselt daraufhin seine Bezeichnung zu **Deinstallieren**. Klicken Sie darauf, wenn Sie die Schnittstelle wieder deinstallieren möchten.

Auto Load

Auto Load bedeutet, dass die Extension beim Start von Vertec geladen wird.

Normalerweise wird eine Extension erst geladen und initialisiert, wenn zum ersten Mal darauf zugegriffen wird. Das ist für Buchhaltungs-Schnittstellen ausreichend. Telefonie-Schnittstellen, die eingehende Anrufe behandeln sollen, müssen unabhängig vom ersten Zugriff beim Start der Applikation geladen werden. Dafür braucht es die Auto Load Einstellung.

Logging

Die Logging-Checkbox schaltet den Log-Output für die entsprechende Extension ein. Die Einstellung gilt nur, solange die Applikation läuft. Der Log-Output erscheint in der Python Konsole (siehe 8.2) oder im Output-Fenster des Script-Editors (siehe 8.1), falls diese geöffnet sind, sowie im Vertec Logfile.

Defaultwerte

Setzt alle Einstellungen der Extension auf die Werte zurück, mit denen sie beim Installieren initialisiert werden.

Reload

Startet die Extension neu bzw. lädt nach einer Code-Änderung den Code neu, ohne dass die Applikation dafür extra neu gestartet werden muss.

Eingebauter Code

Mit Vertec Version 6.3 werden auch zwei neue, cloudfähige Extensions mitgeliefert:

- SharePoint Online DMS Schnittstelle (siehe 10.3)
- BMD Debitoren Listenschnittstelle (siehe 10.3)

Von diesen beiden Schnittstellen befindet sich der Code in der Vertec Applikation und nicht mehr im Extensions Ordner. Er kann auf dem Extension-Objekt auf der Seite **Eingebauter Code** eingesehen werden:

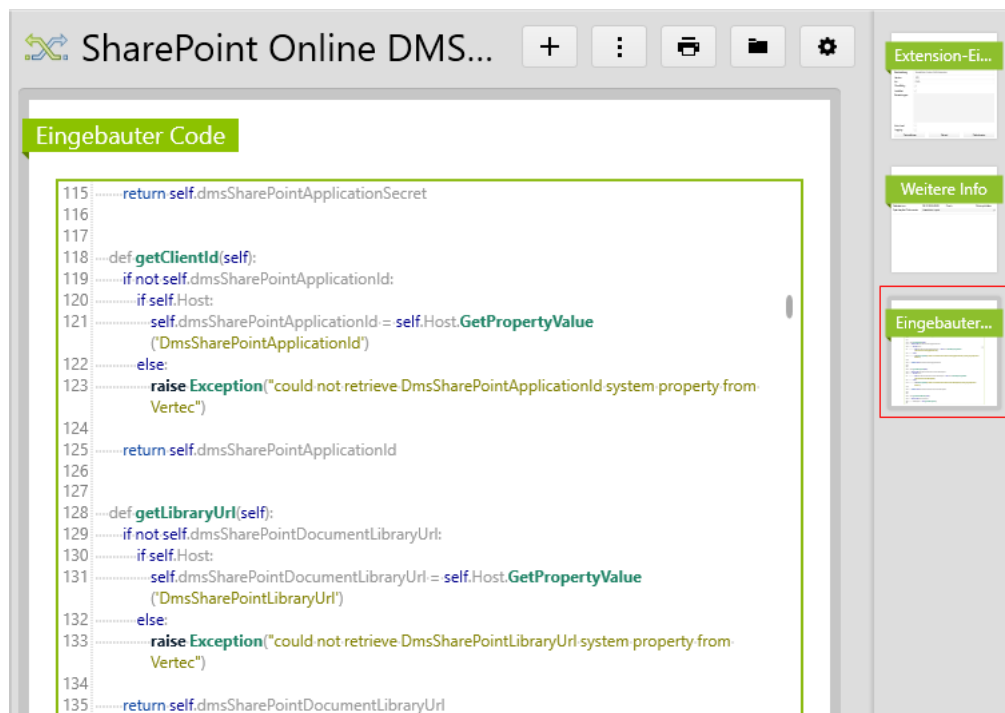


Abbildung 43: Eingebauter Code einer Extension

10.2 Schnittstelle kundenspezifisch anpassen

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

Schnittstellen-Script erzeugen

Um eine neue Schnittstelle zu erzeugen oder eine bestehende anzupassen, erzeugen Sie im Ordner Einstellungen > Scripts einen neuen Scripteintrag und verfassen den Schnittstellen-Code als Python Script.

Die **Bezeichnung** des Scripteintrags ist der Modulname. Zusammen mit dem Klassen-Namen der Extension bildet er die Extension-ID. Diese geben Sie bei der Registrierung der Schnittstelle auf dem Extension-Objekt an (siehe 10.1). Modulnamen müssen eindeutig sein. Sie "überschreiben" also nicht eine bestehende Schnittstelle, indem Sie ein Modul gleichen Namens verfassen, sondern indem Sie ein neues Modul mit neuem Namen verfassen und die Extension-Klasse aus diesem statt dem bereits bestehenden installieren.

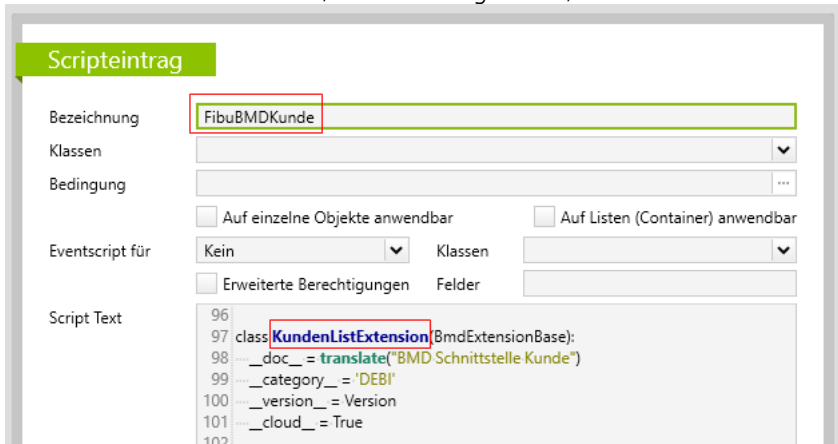
Extension registrieren

Daraufhin erzeugen Sie im Ordner Einstellungen > Extensions eine neue Extension und füllen Sie wie folgt aus:



Abbildung 44: Eine Extension wird registriert

- **Extension-ID:** Setzt sich zusammen aus dem Modulnamen (Name des Scripteintrags) und der Extension-Klasse im Code, durch Punkt getrennt, z.B.



Folgende Attribute werden durch Eigenschaften der Klasse im Code gesteuert:

- **Beschreibung:** `__doc__`
- **Version:** `__version__`
- **Art:** `__category__` (die verschiedenen Arten sehen Sie in der Liste unter Einstellungen > Extensions)
- **Cloudfähig:** `__cloud__` (True oder False)

10.3 Listenbasiertes Buchen

Linie: Standard, Expert | Module: Leistung & CRM | Version: 6.3

Zur Unterstützung von file-basierten Buchhaltungs-Schnittstellen in Cloud-Szenarien wird mit Vertec 6.3 die Möglichkeit eingeführt, Buchungen auf Listen auszuführen.

Dabei wird beim Buchen und Stornieren eine Liste von Rechnungen oder Debitoren statt einzelner Objekte an die Schnittstelle übergeben.

- Beim Buchen/Stornieren über rechte Maustaste in Listen wird die ganze gewählte Liste an die Schnittstelle übergeben.
- Beim Klicken auf den Buchen-/Stornieren-Button in einem einzelnen Fenster wird eine Liste mit einem einzelnen Objekt an die Schnittstelle übergeben.

Auf der Debitor-Extension (zum Thema Extensions siehe auch 10.1) gibt es dafür zwei neue Interface Methoden:

- `PostInvoiceBookings (Rechnung-List)` bucht eine Liste von Rechnungen
- `CancelInvoiceBookings (Rechnung-List)` storniert eine Liste von Rechnungen

Die Buchbarkeit wird geprüft. Wenn eine Rechnung auf der Liste die Voraussetzungen nicht erfüllt, also nicht buchbar ist, wird sie nicht gebucht – die zu buchende Liste wird eins kürzer. Wenn keine der zu buchenden Rechnungen gebucht werden kann, wird eine Meldung ausgegeben und keine Datei geliefert.

Am Schluss des Buchens wird ein Buchungs-File als Download an den Client zurückgeliefert.

Falls es sich beim Client um die Desktop App handelt, wird zudem noch die Systemeinstellung Buchhaltung > **Buchhaltungsdateien Speicherpfad** berücksichtigt: Ist dort ein lokaler Pfad eingetragen, wird die Datei dort gespeichert und nach dem Speichern eine Meldung angezeigt.

Als erste Schnittstelle hat die BMD Debitoren Listen Schnittstelle (siehe 10.4) das listenbasierte Buchen implementiert.

10.4 BMD Debitoren Listen Schnittstelle

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Bei der BMD Debitoren Listen Schnittstelle handelt es sich um eine Dateischnittstelle, welche CSV Dateien erzeugt.

Aktuell können damit Debitor Rechnungen in BMD gebucht und storniert werden.

Die Schnittstelle ist cloudfähig und implementiert das listenbasierte Buchen (siehe 10.3). Da bei jedem Buchungsvorgang direkt eine Datei erzeugt wird, können damit Buchungen direkt auf ganzen Listen durchgeführt werden statt für alle Objekte einzeln.

Schnittstelle installieren

Um die Schnittstelle zu installieren, öffnen Sie im Ordner **Einstellungen > Extensions** die **BMD Debitoren Listen Schnittstelle** und klicken auf **Installieren**.

Die Schnittstelle legt folgende **Systemeinstellungen Buchhaltung** an:

Buchhaltungsdateien Speicherpfad: Optional. Hier kann ein Dateipfad hinterlegt werden, wo die erzeugten Buchhaltungsdateien gespeichert werden sollen. Dieser wird nur berücksichtigt, wenn mit der Desktop App gearbeitet wird (siehe Abschnitt Rechnungen buchen / stornieren weiter unten).

PropertyName: AccountingFilePath. StringProperty.

Nächste Belegnummer: Standardwert: 0.

Steht hier eine Zahl > 0 drin, dann ist dies der Zähler für Belegnummern. Jede Buchung (Buchung und Storno) wird dann mit diesem Zähler als Belegnummer gebucht. Beim Buchen wird die verwendete Belegnummer in Vertec als Belegnummer in die Rechnung geschrieben, beim Stornieren nicht (Standardverhalten von Vertec Schnittstellen).

Steht hier etwas anderes (0, eine negative Zahl, leer), wird als Belegnummer immer die Rechnungsnummer verwendet. Somit hat eine Buchung und ihr Storno dieselbe Belegnummer.

PropertyName: AccountingNextDocumentnumber. IntegerProperty.

Rechnungen buchen / stornieren

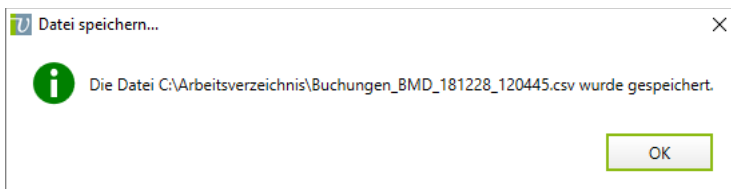
Das Buchen / Stornieren von Rechnungen erfolgt entweder auf einer einzelnen Rechnung auf der Seite Buchhaltung, oder auf einer Liste von Rechnungen durch Rechtsklick > **Buchen** bzw. **Stornieren**.

Alle Datensätze einer Datei müssen zum selben Mandanten gehören. Bei Buchungen wird das von der Schnittstelle geprüft und gegebenenfalls eine Fehlermeldung ausgegeben.

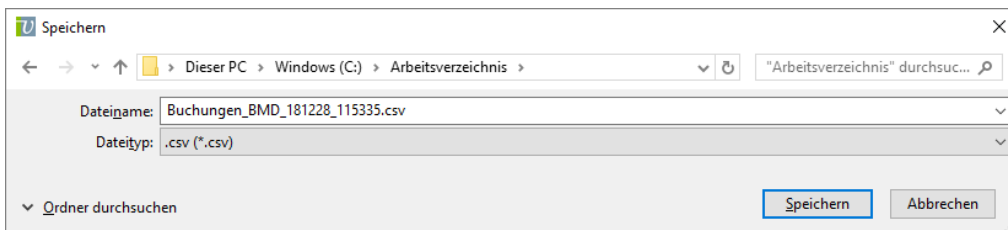
Für jeden Buchung-/Stornierungsvorgang wird eine Datei erzeugt.

Die erzeugten Buchungsdateien heissen **Buchungen_Mandantcode_YYMMDD_HHMMSS.csv**.

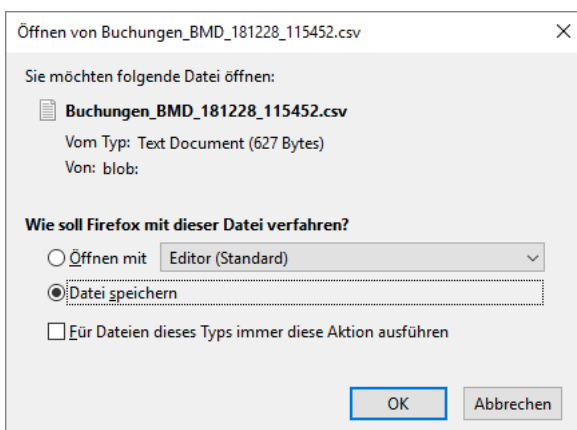
Handelt es sich beim Client um die **Desktop App** und ist in den Systemeinstellungen Buchhaltung ein Buchhaltungsdateien Speicherpfad angegeben, wird die Datei direkt unter diesem Pfad gespeichert:



Handelt es sich um die **Cloud App**, erscheint beim Buchen ein Dialog, wo die Datei gespeichert werden soll:



Handelt es sich um die Web App, wird die Datei direkt zum Download angeboten:



10.5 Unterstützung von Geschäftsbereichen in AbaConnect Webservice Schnittstelle

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.3

Neu unterstützt die Schnittstelle Geschäftsbereiche. Geschäftsbereiche betreffen in Abacus nur Bewegungsdaten (Buchungen), nicht Stammdaten (Personenkonto).

In Vertec sind die Geschäftsbereiche so implementiert, dass sie eine Eigenschaft des Personenkontos sind. Das heisst, ein Personenkonto weiss, in welchen Geschäftsbereich es buchen soll, und bucht immer in denselben Geschäftsbereich. Bereichsübergreifende Buchungen werden derzeit nicht unterstützt.

Der Geschäftsbereich muss dem Code des Mandanten mit einem Bindestrich folgend eingetragen werden. Soll also ein Personenkonto in den Geschäftsbereich 10 des Mandanten 7777 gebucht werden, so muss der Eintrag 7777-10 lauten.

10.6 Abacus: Buchen ohne "Customer"-Lizenz

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.4

Das Buchen von Debitorrechnungen funktioniert nun ohne die Abacus-Lizenz für das Interface "Customer". Das bedeutet: Ab sofort können auch Kunden, die nur "Document" lizenziert haben, die Schnittstelle verwenden.

Dies betrifft sowohl die AbaConnect Webservice sowie die XML Schnittstelle.

10.7 Sage50 Storno implementiert

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

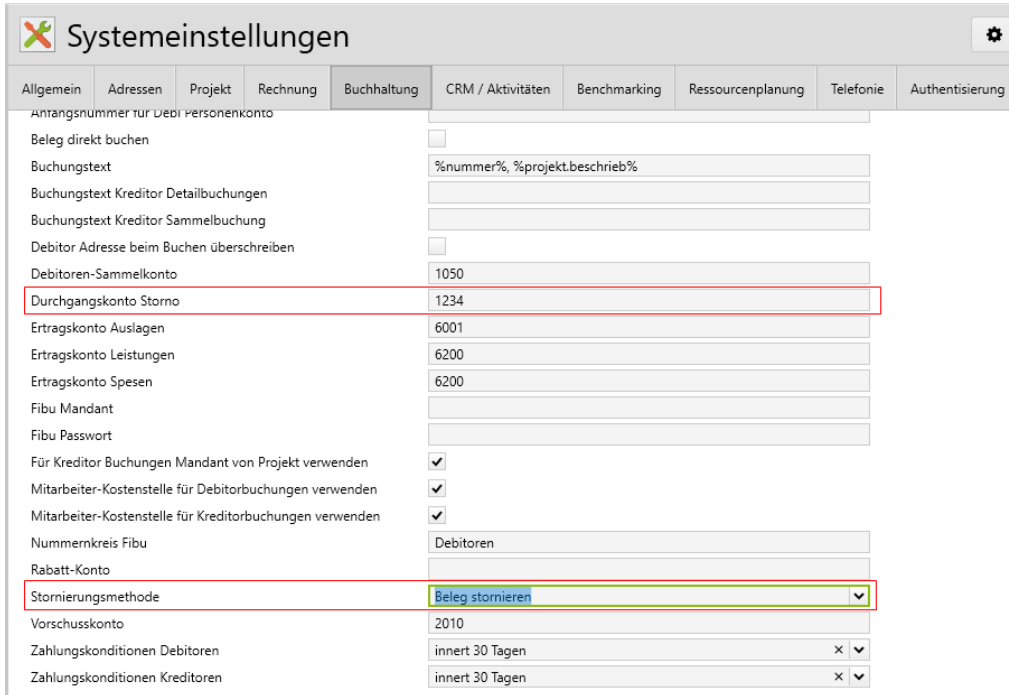
Neu können Rechnungen bzw. Kreditoren mit der Sage50 Schnittstelle auch storniert werden. Dafür gibt es neu in Vertec zwei **Systemeinstellungen Buchhaltung**:

Stornierungsmethode: Hier wird die Stornierungsmethode hinterlegt. Der Standardwert ist **Beleg löschen**. Möchten Sie stattdessen die neue Stornierungsmethode verwenden, stellen Sie diese Einstellung auf **Beleg stornieren**.

PropertyName: Stornierungsmethode. AuswahlProperty.

Durchgangskonto Storno: Wenn Sie unter Stornierungsmethode **Beleg stornieren** ausgewählt haben, müssen Sie ein Sage50 Durchgangskonto für den Storno hier eintragen.

PropertyName: Durchgangskonto. StringProperty.



Systemeinstellungen	
Alleinstellung	Adressen
Projekt	Rechnung
Buchhaltung	CRM / Aktivitäten
Benchmarking	Ressourcenplanung
Telefonie	Authentisierung
Anfangsnummer für Ueoi Personenkonto	
Beleg direkt buchen	
Buchungstext	%nummer%, %projekt.beschrieb%
Buchungstext Kreditor Detailbuchungen	
Buchungstext Kreditor Sammelbuchung	
Debitor Adresse beim Buchen überschreiben	
Debitoren-Sammelkonto	1050
Durchgangskonto Storno	1234
Ertragskonto Auslagen	6001
Ertragskonto Leistungen	6200
Ertragskonto Spesen	6200
Fibu Mandant	
Fibu Passwort	
Für Kreditor Buchungen Mandant von Projekt verwenden	☒
Mitarbeiter-Kostenstelle für Debitorbuchungen verwenden	☒
Mitarbeiter-Kostenstelle für Kreditorbuchungen verwenden	☒
Nummernkreis Fibu	Debitoren
Rabatt-Konto	
Stornierungsmethode	Beleg stornieren
Vorschusskonto	2010
Zahlungskonditionen Debitoren	innert 30 Tagen
Zahlungskonditionen Kreditoren	innert 30 Tagen

Abbildung 45: Diese beiden Felder müssen ausgefüllt sein, damit mit der Sage50 Schnittstelle storniert werden kann

Wenn Sie in Vertec auf der Rechnung oder dem Kreditor unter **Buchhaltung** auf **Stornieren** klicken, wird in Sage50 folgender Vorgang durchgeführt:

- Der Storno macht auf dem originalen OP einen neuen Beleg auf das Durchgangskonto und gibt diesem eine neue Belegnummer. Der OP hat nach dem Storno somit 2 Belege. Die Belegnummer des Stornobeleges kann von Vertec nicht verarbeitet werden. Die Belegnummer auf Rechnung oder Kreditor bleibt deshalb die des ersten Beleges.
- Anschliessend erstellt der Storno einen weiteren OP (den Storno-OP). Darauf werden 2 Belege erstellt: Der eine Beleg gleicht den original Beleg aus (Soll/Haben vertauscht), der andere Beleg gleicht den Übergangskonto-Beleg des originalen OPs aus.
- Der Storno-OP erhält die Nummer des originalen OPs mit einem grossen 'S' angehängt.

Das bedeutet mit anderen Worten: Der Storno wird durch das Erstellen eines weiteren OP (Storno-OP) vorgenommen, mithilfe dessen die Buchungen über ein Durchgangskonto ausgeglichen werden.

Die Rechnung bzw. der Kreditor erscheint in Vertec daraufhin wieder als "nicht gebucht".

Hinweis Mit derselben OP Nummer kann nicht mehr gebucht werden. Möchten Sie die Rechnung bzw. den Kreditor erneut buchen, müssen Sie die Rechnungsnummer bzw. die Kreditorennummer in Vertec ändern.

Dieser Vorgang kann auch über Geschäftsjahresgrenzen hinweg durchgeführt werden.

10.8 Selectline: COM Freischaltcode nur verwenden falls angegeben

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.4

Der COM Freischaltcode, welcher von neueren Selectline Versionen via API verlangt wird, wird nur an das Selectline API übermittelt, wenn er eingetragen ist.

Ist er nicht eingetragen, wird auch die entsprechende Methode im API nicht aufgerufen. Dies ermöglicht sowohl das Betreiben von neuen als auch von alten Selectline Versionen, welche diese API Methode noch nicht kennen.

Dies gilt sowohl für die Selectline Fibu- als auch Auftrags-Schnittstellen.

Das Verhalten ist wie folgt:

- Ab Selectline Version 16: Wenn in Vertec ein korrekter COM Freischaltcode eingetragen ist, werden die Aktionen ausgeführt. Ist kein COM Freischaltcode eingetragen, wird eine Fehlermeldung geworfen.
- Bis Selectline 16: Wenn in Vertec kein COM Freischaltcode eingetragen ist, werden die Aktionen ausgeführt. Ist ein COM Freischaltcode vorhanden, wird eine Fehlermeldung geworfen, da Selectline das Feld noch nicht kennt.

10.9 DATEV Schnittstelle überarbeitet

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

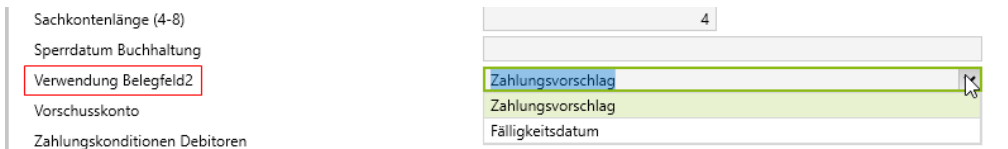
Die DATEV Schnittstelle wurde in folgenden Punkten überarbeitet:

Unterstützte Buchhaltungsdateien

Zusätzlich zum bisherigen Format Version 3.0 unterstützt die Schnittstelle nun auch alle nachfolgenden kompatiblen Versionen.

Verwendung Belegfeld2 einstellbar

Neu gibt es eine Systemeinstellung Buchhaltung namens **Verwendung Belegfeld2**.



The screenshot shows a settings window with a list of options on the left and a corresponding selection area on the right. The option 'Verwendung Belegfeld2' is highlighted in the list. The selection area on the right shows a dropdown menu with 'Zahlungsvorschlag' selected, and 'Fälligkeitsdatum' is also visible as an option.

Hier kann eingestellt werden, ob das Belegfeld2 mit einem Zahlungsvorschlag oder dem Fälligkeitsdatum beschrieben werden soll.

11 Performance

11.1 Firebird Page- und Buffersize erhöht

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Die Performance der Firebird Datenbank lässt sich verbessern, wenn man dem Datenbankserver mehr Memory für seine Datenbank zur Verfügung stellt.

Dafür werden beim Datenbank-Konvert auf Version 6.3 folgende Werte geändert:

- Page-Size: von bisher 8192 Bytes auf den Maximalwert von 16384 Bytes erhöht.
- Buffer-Size: von bisher 2048 auf 10'000 Pages erhöht.

Dadurch steht dem Firebird Datenbank-Server mehr Memory zur Verfügung, was sich positiv auf die Performance auswirken kann.

Arbeitsspeicher Vertec benötigt ab Version 6.3 dadurch mehr Memory auf dem Datenbankserver. Dieser muss über ausreichend Arbeitsspeicher verfügen.

11.2 Berücksichtigung von Aktivitäten bei Outlook Addin Projektauswahl optimiert

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.2.0.8

Die Logik zur Bestimmung der vorgeschlagenen Projekte im Outlook Addin berücksichtigte bis anhin auch Aktivitäten, die mit der gefundenen Adresse über den allgemeinen Aktivitäten-Link (**eintraege**) verbunden (also der Aktivität über "Weitere Adressen" oder "Objekte aus Ordner" zugeordnet) sind.

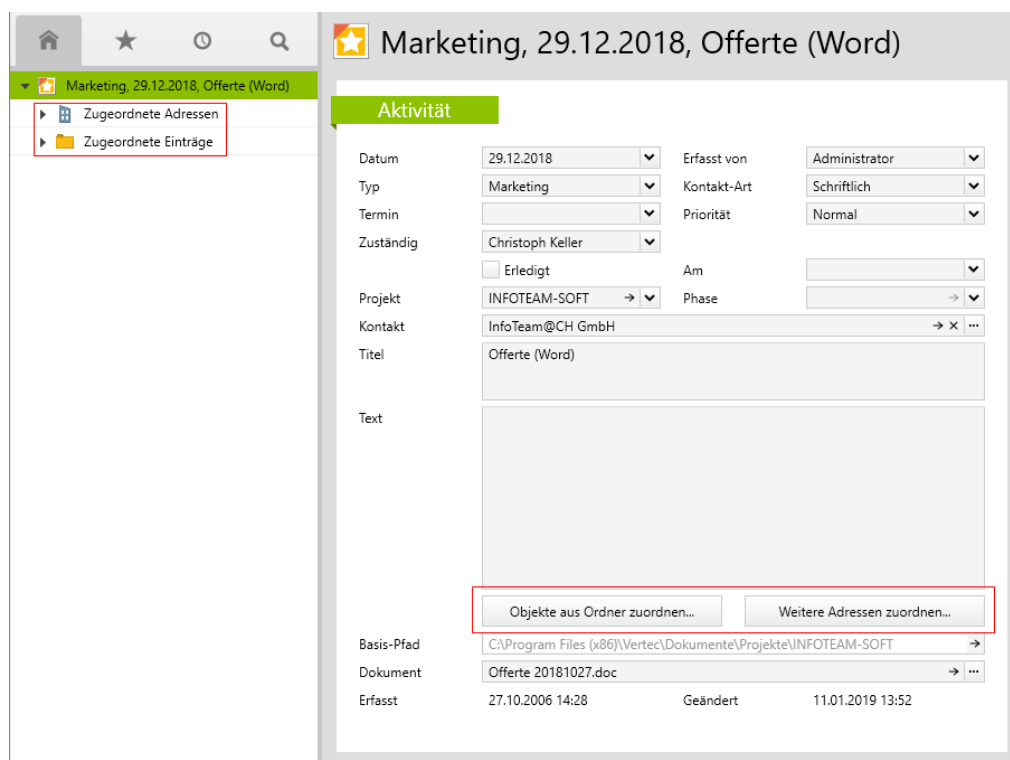


Abbildung 46: Zugeordnete Adressen und Einträge werden im Outlook Addin nicht mehr berücksichtigt

Diese Teilabfrage hat viel Zeit benötigt und wenig Nutzen gestiftet und wurde darum entfernt.

11.3 Projekt persistent auf allen Phasen speichern

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Das Datenmodell von hierarchischen Phasen (Subphasen) ist so gestaltet, dass die Subphase jeweils auf ihre Parent-Phase verweist.

Das bedeutet, dass eine Subphase keinen direkten Bezug zum Projekt hat, zu welchem die Phasen-Hierarchie gehört.

Dadurch ist es schwierig, z.B. eine SQL Abfrage zu formulieren, welche alle Phasen eines oder mehrerer Projekt abfragt.

Auf Projekten gab es bisher ein derived (zur Laufzeit berechnetes) Attribut namens **allePhasen**. Dieses ist neu persistent und enthält die Liste aller Phasen und Subphasen aller Ebenen.

Auf den Phasen heisst das Attribut **owningProjekt** und enthält das entsprechende Projekt.

12 Technik

12.1 Notif als Teil des CloudServer Dienstes

Linie: Standard, Expert | Module: Leistung & CRM | Version 6.3

Der Notif Server ist neu in den CloudServer Dienst integriert.

Für den normalen Betrieb sind keine Änderungen notwendig.

Die Angabe von `Port=XXXX` in der `[Notif]` Section in der Konfigurationsdatei `Vertec.ini` aktiviert oder deaktiviert wie bisher den Notif Server.

In der `[CloudServer]` Section kann der Notif Server explizit deaktiviert werden. `Notif Server = false` in der `[CloudServer]` Section übersteuert oben genannte Aktivierung.

```
[CloudServer]
Server Address=SERVER1
Server Port=8081
Management Port=8082
Notif Server = false

[Notif]
Host= SERVER1
Port=8091
PollInterval=30000
```

Die Zeile `Notif Server = xxx` ist optional.

Wenn sie nicht vorhanden ist, ist das Verhalten äquivalent zu `Notif Server = true`.

Wird `Vertec.Desktop.exe` mit `/DB` gestartet, dann ist Notif explizit ausgeschaltet, da es sich um eine andere DB handelt.

Vertec Service (VertecServer.exe) wird entfernt

Damit ist der Vertec Service Dienst nicht mehr in Verwendung und wird vom Update entfernt.

Rückwärtskompatibilität

Im Normalfall muss nichts gemacht werden. Wurde aber der `VertecService` unter eigenem Namen registriert, muss er auch manuell wieder deinstalliert werden. Das Vertec Update entfernt nur die Standard-Registrierung.

Die Deinstallation muss vor dem Update erfolgen, da danach kein `VertecServer.exe` mehr zur Verfügung steht, welches für ein `/uninstall` gebraucht wird.

Zumindest muss sichergestellt sein, dass ein solcher Dienst nicht parallel zum CloudServer mit integriertem Notif läuft. Das würde zu Port-Problemen führen, resp. der CloudServer könnte nicht starten, wenn der andere Service noch läuft.

Weitere Informationen zur manuellen Installation/Deinstallation von Vertec Service Diensten finden Sie in der Online Knowledge Base unter www.vertec.com/kb/serviceinstanzen.

12.2 Verbessertes Upgrade Verhalten von Vertec.Desktop

Linie: Expert | Module: Leistung & CRM | Version 6.3

Bisher gab es das Problem, dass wenn ein einzelner Benutzer unvorhergesehen Vertec.Desktop mit einer neueren Version startete, alle anderen Clients danach einen Fehler meldeten, dass die Datenbank-Version neuer sei als die Applikations-Version.

Neu werden Versions-Upgrades durch die Desktop App nur noch zugelassen, wenn die App auf dem Datenbankserver gestartet wurde. Auf den Clients erscheint hingegen eine Fehlermeldung.

12.3 XML Server mit fixem Zahlenformat

Linie: Expert | Module: Leistung & CRM | Version 6.2.0.8

Bisher wurden Zahlenwerte vom XML Server regional dargestellt und interpretiert.

Neu verwendet der XML Server kein regionsspezifisches Zahlenformat mehr, sondern akzeptiert nur noch fix den Punkt als Dezimaltrennzeichen und keine Tausendertrennzeichen mehr.

Rückwärtskompatibilität

Regionale Zahlenformatierungen werden neu nicht mehr akzeptiert. Das bedeutet, dass bestehende Scripts kontrolliert und allenfalls umgeschrieben werden müssen.

12.4 Key-Values und Tags über den XML Server setzen

Linie: Expert | Module: Leistung & CRM | Version 6.3

Neu können Key-Values und Tags über den XML Server gesetzt bzw. entfernt werden.

Dafür gibt es im XML Update Element folgende neuen Elemente:

- `<keyvalue>` mit Sub-Elementen `<key>` und `<value>`
- `<tag>` mit Sub-Elementen `<add>` und `<remove>`

```
<Body>
  <Update>
    <Projektbearbeiter>
      <objref>4019</objref>
      <kuerzel>ABC</kuerzel>
      <keyvalue>
        <key>ExternalId</key>
        <value>"123465"</value>
      </keyvalue>
      <tag>
        <add>selected</add>
      </tag>
    </Projektbearbeiter>
    ...
  </Update>
</Body>
```

Die Abfrage von bestehenden Key-Values bzw. Tags im Query Element kann wie gewohnt via OCL erfolgen, hier gibt es keine neuen Elemente:

```
<Body>
  <Query>
    <Selection>
      <ocl>projektbearbeiter->select (hasTag ('selected')) </ocl>
    </Selection>
    <ResultDef>
      <member>kuerzel</member>
    </ResultDef>
  </Query>
</Body>
```

12.5 Key-Values und Tags via COM setzen

Linie: Expert | Module: Leistung & CRM | Version 6.3

Neu können Key-Values und Tags via COM gesetzt bzw. entfernt werden.

Dafür wurden auf **IVtcObject** analog zu Python die folgenden Methoden eingeführt:

SetKeyValue(Key as String, Value as Variant)

```
Set Vertec = CreateObject("Vertec.App")

Set argobject = Vertec.argobject
argobject.SetKeyValue "Datum", Date
argobject.SetKeyValue "Boolean", True
```

AddTag(Tagname as String)

```
argobject.addTag ("mytag")
```

RemoveTag(Tagname as String)

```
argobject.removeTag ("mytag")
```

HasTag(Tagname as String): boolean

```
If argobject.hasTag("mytag") Then
```