

Sagittarius-IT



kivitendo

kivitendo Benutzerhandbuch

Veröffentlicht: 15. Januar 2016

Sagittarius-IT

Pfungstädter Str. 22

64297 Darmstadt

<http://www.sagittarius-it.de/>

Copyright © 2016 Sagittarius-IT und Mitwirkende. Alle Rechte vorbehalten.

Es wird die Genehmigung erteilt, dieses Dokument unter den Bedingungen der GNU Free Documentation License, Version 1.2 oder (optional) Version 1.3 zu vervielfältigen, zu verbreiten und/oder zu verändern; die unveränderlichen Abschnitte hierbei sind der Urheberrechtshinweis und die Lizenzbedingungen. Eine Kopie dieser Lizenz (Version 1.2) finden Sie im Abschnitt „GNU Free Documentation License“.

SUSE und openSUSE sind Marken von SUSE LLC

kivitendo ist eine Marke von LINET Services GmbH

Alle anderen Drittanbieter-Marken sind das Eigentum der jeweiligen Inhaber.

Dieses Benutzerhandbuch stellt einen Auszug aus der original kivitendo Dokumentation dar, der mit openSUSE entsprechenden Erweiterungen versehen wurde. Die Original Dokumentation ist von der Weboberfläche aus erreichbar und liegt der Installation bei.

Alle Informationen in diesem Buch wurden mit größter Sorgfalt zusammengestellt. Doch auch dadurch kann hundertprozentige Richtigkeit nicht gewährleistet werden. Weder Sagittarius-IT noch ihre Tochtergesellschaften noch die Autoren noch die Übersetzer können für mögliche Fehler und deren Folgen haftbar gemacht werden.

Sagittarius-IT, Pfungstädter Str. 22, 64297 Darmstadt

Inhaltsverzeichnis

1 Grundkonfiguration.....	5
1.0. Übersicht.....	5
1.1. Benutzerauthentifizierung und Administratorpasswort.....	5
1.1.1. Grundlagen zur Benutzerauthentifizierung.....	5
1.1.2. Administratorpasswort.....	5
1.1.3. Authentifizierungsdatenbank.....	5
1.1.4. Passwortüberprüfung.....	6
1.1.5. Name des Session-Cookies.....	6
1.1.6. Anlegen der Authentifizierungsdatenbank.....	7
2 Mandanten-, Benutzer- und Gruppenverwaltung.....	7
2.1.1. Zusammenhänge.....	7
2.1.2. Mandanten, Benutzer und Gruppen.....	7
2.1.3. Datenbanken anlegen.....	8
2.1.4. Gruppen anlegen.....	9
2.1.5. Benutzer anlegen.....	9
2.1.6. Mandanten anlegen.....	10
2.2. Drucker- und Systemverwaltung.....	10
2.2.1. Druckeradministration.....	10
2.2.1.1 Drucker anlegen in YAST.....	10
2.2.2. System sperren / entsperren.....	11
2.3. E-Mail-Versand aus kivitendo heraus.....	12
2.3.1. Versand über lokalen E-Mail-Server.....	12
2.3.2. Versand über einen SMTP-Server.....	12
2.4. Drucken mit kivitendo.....	13
2.4.1. Vorlagenverzeichnis anlegen.....	13
2.4.2. Der Druckvorlagensatz RB.....	14
2.4.3. f-tex.....	14
2.4.3.1. Feature-Übersicht.....	14
2.4.3.2. Die Installation.....	15
2.4.3.3. f-tex Funktionsübersicht.....	15
2.4.3.4. Bruttopreise für Endverbraucher.....	16
2.4.3.5. Lieferadressen.....	16
2.4.4. Der Druckvorlagensatz rev-odt.....	16
2.4.5. Allgemeine Hinweise zu LaTeX Vorlagen.....	17
2.5. OpenDocument-Vorlagen.....	17
2.6. Konfiguration zur Einnahmenüberschussrechnung/Bilanzierung: EUR.....	19
2.6.1. Einführung.....	19
2.6.2. Konfigurationsparameter.....	19
2.6.3. Festlegen der Parameter.....	19
2.6.4. Bemerkungen zur Bestandsmethode.....	19
2.6.5. Bekannte Probleme.....	20
2.7. SKR04 19% Umstellung für innergemeinschaftlichen Erwerb.....	20
2.7.1. Einführung.....	20
2.7.2. Konto 3804 manuell anlegen.....	20
2.8. Verhalten des Bilanzberichts.....	21
2.9. Einstellungen pro Mandant.....	21
2.10. kivitendo ERP verwenden.....	22
3. Features und Funktionen.....	22

3.1. Wiederkehrende Rechnungen.....	22
3.1.1. Einführung.....	22
3.1.2. Konfiguration.....	22
3.1.3. Spezielle Variablen.....	23
3.1.4. Auflisten.....	25
3.1.5. Erzeugung der eigentlichen Rechnungen.....	25
3.1.6. Erste Rechnung für aktuellen Monat erstellen.....	25
3.2. Bankerweiterung.....	25
3.2.1. Einführung.....	25
3.3. Dokumentenvorlagen und verfügbare Variablen.....	25
3.3.1. Einführung.....	25
3.3.2. Variablen ausgeben.....	26
3.3.3. Verwendung in Druckbefehlen.....	26
3.3.4. Anfang und Ende der Tags verändern.....	26
3.3.5. Zuordnung von den Dateinamen zu den Funktionen.....	27
3.3.6. Sprache, Drucker und E-Mail.....	27
3.3.7. Allgemeine Variablen, die in allen Vorlagen vorhanden sind.....	28
3.3.7.1. Metainformationen zur angeforderten Vorlage.....	28
3.3.7.2. Stammdaten von Kunden und Lieferanten.....	28
3.3.7.3. Informationen über den Bearbeiter.....	31
3.3.7.4. Informationen über den Verkäufer.....	31
3.3.7.5. Variablen für die einzelnen Steuern.....	31
3.3.7.6. Variablen für Lieferbedingungen.....	32
3.3.8. Variablen in Rechnungen.....	32
3.3.8.1. Allgemeine Variablen.....	32
3.3.8.2. Variablen für jeden Posten auf der Rechnung.....	33
3.3.8.3. Variablen für die einzelnen Zahlungseingänge.....	34
3.3.8.4. Benutzerdefinierte Kunden- und Lieferantenvariablen.....	35
3.3.9. Variablen in Mahnungen und Rechnungen über Mahngebühren.....	35
3.3.9.1. Namen der Vorlagen.....	35
3.3.9.2. Allgemeine Variablen in Mahnungen.....	35
3.3.9.3. Variablen für jede gemahnte Rechnung in einer Mahnung.....	35
3.3.9.4. Variablen in automatisch erzeugten Rechnungen über Mahngebühren. .	36
3.3.10. Variablen in anderen Vorlagen.....	36
3.3.10.1. Einführung.....	36
3.3.10.2. Angebote und Preisanfragen.....	37
3.3.10.3. Auftragsbestätigungen und Lieferantenaufträge.....	37
3.3.10.4. Lieferscheine (Verkauf und Einkauf).....	37
3.3.10.5. Variablen für Sammelrechnung.....	37
3.3.11. Blöcke, bedingte Anweisungen und Schleifen.....	38
3.3.11.1. Einführung.....	38
3.3.11.2. Der if-Block.....	38
3.3.11.3. Der foreach-Block.....	39
3.3.12. Markup-Code zur Textformatierung innerhalb von Formularen.....	40
3.4. Excel-Vorlagen.....	40
3.4.1. Zusammenfassung.....	40
3.4.2. Bedienung.....	40
3.4.3. Variablensyntax.....	41
3.4.4. Einschränkungen.....	41

1 Grundkonfiguration

1.0. Übersicht

Wenn sie dieses Dokument lesen ist die Installation von kivitendo bereits abgeschlossen und sie haben die Weboberfläche geöffnet. Bei der installation der Appliance ist die Authentifizierungs Datenbank bereits angelegt. Sie können direkt zu dem Kapitel Benutzer, Gruppen und Datenbanken anlegen gehen.

1. **Benutzerinformationen speichern können:** man benötigt mindestens eine Datenbank, in der Informationen zur Authentifizierung sowie die Nutzdaten gespeichert werden. Wie man das als Administrator macht, verrät "Abschnitt 2.8, „Benutzerauthentifizierung und Administratorpasswort““.
2. **Benutzer, Gruppen und Datenbanken anlegen:** wie dies alles zusammenspielt erläutert "Abschnitt 2.9, „Mandanten-, Benutzer- und Gruppenverwaltung““.
3. **Los geht's:** alles soweit erledigt? Dann kann es losgehen: "Abschnitt 2.18, „kivitendo ERP verwenden““

Alle weiteren Unterkapitel in diesem Kapitel sind ebenfalls wichtig und sollten vor einer ernsthaften Inbetriebnahme gelesen werden

1.1. Benutzerauthentifizierung und Administratorpasswort

Informationen über die Einrichtung der Benutzerauthentifizierung, über die Verwaltung von Gruppen und weitere Einstellungen

1.1.1. Grundlagen zur Benutzerauthentifizierung

kivitendo verwaltet die Benutzerinformationen in einer Datenbank, die im folgenden "Authentifizierungsdatenbank" genannt wird. Für jeden Benutzer kann dort eine eigene Datenbank für die eigentlichen Finanzdaten hinterlegt sein. Diese beiden Datenbanken können, müssen aber nicht unterschiedlich sein.

Im einfachsten Fall gibt es für kivitendo nur eine einzige Datenbank, in der sowohl die Benutzerinformationen als auch die Daten abgelegt werden.

Zusätzlich ermöglicht es kivitendo, dass die Benutzerpasswörter entweder gegen die Authentifizierungsdatenbank oder gegen einen LDAP-Server überprüft werden.

Welche Art der Passwortüberprüfung kivitendo benutzt und wie kivitendo die Authentifizierungsdatenbank erreichen kann, wird in der Konfigurationsdatei config/kivitendo.conf festgelegt. Diese muss bei der Installation und bei einem Upgrade von einer Version vor v2.6.0 angelegt werden. Eine Beispielkonfigurationsdatei config/kivitendo.conf.default existiert, die als Vorlage benutzt werden kann.

1.1.2. Administratorpasswort

Das Passwort, das zum Zugriff auf das Administrationsinterface benutzt wird, wird ebenfalls in dieser Datei gespeichert. Es kann auch nur dort und nicht mehr im Administrationsinterface selber geändert werden. Der Parameter dazu heißt admin_password im Abschnitt [authentication].

1.1.3. Authentifizierungsdatenbank

Die Verbindung zur Authentifizierungsdatenbank wird mit den Parametern in [authentication/database] konfiguriert. Hier sind die folgenden Parameter anzugeben:

host	Der Rechnername oder die IP-Adresse des Datenbankservers
------	--

Port	Die Portnummer des Datenbankservers, meist 5432
Db	Der Name der Authentifizierungsdatenbank
User	Der Benutzername, mit dem sich kivitendo beim Datenbankserver anmeldet (z.B. "postgres")
Password	Das Passwort für den Datenbankbenutzer

Die Datenbank muss noch nicht existieren. kivitendo kann sie automatisch anlegen (mehr dazu siehe unten).

1.1.4. Passwortüberprüfung

kivitendo unterstützt Passwortüberprüfung auf zwei Arten: gegen die Authentifizierungsdatenbank und gegen einen externen LDAP- oder Active-Directory-Server. Welche davon benutzt wird, regelt der Parameter `module` im Abschnitt `[authentication]`.

Sollen die Benutzerpasswörter in der Authentifizierungsdatenbank gespeichert werden, so muss der Parameter `module` den Wert `DB` enthalten. In diesem Fall können sowohl der Administrator als auch die Benutzer selber ihre Psaswörter in kivitendo ändern.

Soll hingegen ein externer LDAP- oder Active-Directory-Server benutzt werden, so muss der Parameter `module` auf `LDAP` gesetzt werden. In diesem Fall müssen zusätzliche Informationen über den LDAP-Server im Abschnitt `[authentication/ldap]` angegeben werden:

Host	Der Rechnername oder die IP-Adresse des LDAP- oder Active-Directory-Servers. Diese Angabe ist zwingend erforderlich.
Port	Die Portnummer des LDAP-Servers; meist 389
Tls	Wenn Verbindungsverschlüsselung gewünscht ist, so diesen Wert auf '1' setzen, andernfalls auf '0' belassen
Attribute	Das LDAP-Attribut, in dem der Benutzername steht, den der Benutzer eingegeben hat. Für Active-Directory-Server ist dies meist 'sAMAccountName', für andere LDAP-Server hingegen 'uid'. Diese Angabe ist zwingend erforderlich.
base_dn	Der Abschnitt des LDAP-Baumes, der durchsucht werden soll. Diese Angabe ist zwingend erforderlich.
Filter	Ein optionaler LDAP-Filter. Enthält dieser Filter das Wort <code><%login%></code> , so wird dieses durch den vom Benutzer eingegebenen Benutzernamen ersetzt. Andernfalls wird der LDAP-Baum nach einem Element durchsucht, bei dem das oben angegebene Attribut mit dem Benutzernamen identisch ist.
bind_dn und bind_password	Wenn der LDAP-Server eine Anmeldung erfordert, bevor er durchsucht werden kann (z.B. ist dies bei Active-Directory-Servern der Fall), so kann diese hier angegeben werden. Für Active-Directory-Server kann als 'bind_dn' entweder eine komplette LDAP-DN wie z.B. 'cn=Martin Mustermann,cn=Users,dc=firmendomain' auch nur der volle Name des Benutzers eingegeben werden; in diesem Beispiel also 'Martin Mustermann'.

1.1.5. Name des Session-Cookies

Sollen auf einem Server mehrere kivitendo-Installationen aufgesetzt werden, so müssen die Namen der Session-Cookies für alle Installationen unterschiedlich sein. Der Name des Cookies wird mit dem Parameter `cookie_name` im Abschnitt `[authentication]` gesetzt.

Diese Angabe ist optional, wenn nur eine Installation auf dem Server existiert.

1.1.6. Anlegen der Authentifizierungsdatenbank

Nachdem alle Einstellungen in config/kivitendo.conf vorgenommen wurden, muss kivitendo die Authentifizierungsdatenbank anlegen. Dieses geschieht automatisch, wenn Sie sich im Administrationsmodul anmelden, das unter der folgenden URL erreichbar sein sollte:

<http://localhost/kivitendo-erp/controller.pl?action=Admin/login>

2 Mandanten-, Benutzer- und Gruppenverwaltung

Nach der Installation müssen Mandanten, Benutzer, Gruppen und Datenbanken angelegt werden. Dieses geschieht im Administrationsmenü, das Sie unter folgender URL finden:

<http://localhost/kivitendo-erp/controller.pl?action=Admin/login>

Verwenden Sie zur Anmeldung das Passwort, das Sie in der Datei config/kivitendo.conf eingetragen haben.

2.1.1. Zusammenhänge

kivitendo verwaltet zwei Sets von Daten, die je nach Einrichtung in einer oder zwei Datenbanken gespeichert werden.

Das erste Set besteht aus Anmeldeinformationen: welche Benutzer und Mandanten gibt es, welche Gruppen, welche BenutzerIn hat Zugriff auf welche Mandanten, und welche Gruppe verfügt über welche Rechte. Diese Informationen werden in der Authentifizierungsdatenbank gespeichert. Dies ist diejenige Datenbank, deren Verbindungsparameter in der Konfigurationsdatei config/kivitendo.conf gespeichert werden.

Das zweite Set besteht aus den eigentlichen Verkehrsdaten eines Mandanten, wie beispielsweise die Stammdaten (Kunden, Lieferanten, Waren) und Belege (Angebote, Lieferscheine, Rechnungen). Diese werden in einer Mandantendatenbank gespeichert. Die Verbindungsinformationen einer solchen Mandantendatenbank werden im Administrationsbereich konfiguriert, indem man einen Mandanten anlegt und dort die Parameter einträgt. Dabei hat jeder Mandant eine eigene Datenbank.

Aufgrund des Datenbankdesigns ist es für einfache Fälle möglich, die Authentifizierungsdatenbank und eine der Mandantendatenbanken in ein und derselben Datenbank zu speichern. Arbeitet man hingegen mit mehr als einem Mandanten, wird empfohlen, für die Authentifizierungsdatenbank eine eigene Datenbank zu verwenden, die nicht gleichzeitig für einen Mandanten verwendet wird.

2.1.2. Mandanten, Benutzer und Gruppen

kivitendos Administration kennt Mandanten, Benutzer und Gruppen, die sich frei zueinander zuordnen lassen.

kivitendo kann mehrere Mandanten aus einer Installation heraus verwalten. Welcher Mandant benutzt wird, kann direkt beim Login ausgewählt werden. Für jeden Mandanten wird ein eindeutiger Name vergeben, der beim Login angezeigt wird. Weiterhin benötigt der Mandant Datenbankverbindungsparameter für seine Mandantendatenbank. Diese sollte über die in der Administrationsoberfläche Datenbankadministration geschehen.

Ein Benutzer ist eine Person, die Zugriff auf kivitendo erhalten soll. Sie erhält einen Loginnamen sowie ein Passwort. Weiterhin legt der Administrator fest, an welchen Mandanten sich ein Benutzer anmelden kann, was beim Login verifiziert wird.

Gruppen dienen dazu, Benutzern innerhalb eines Mandanten Zugriff auf bestimmte Funktionen zu geben. Einer Gruppe werden dafür vom Administrator gewisse Rechte zugeordnet. Weiterhin legt der Administrator fest, für welche Mandanten eine Gruppe gilt, und welche Benutzer Mitglieder in dieser Gruppe sind. Meldet

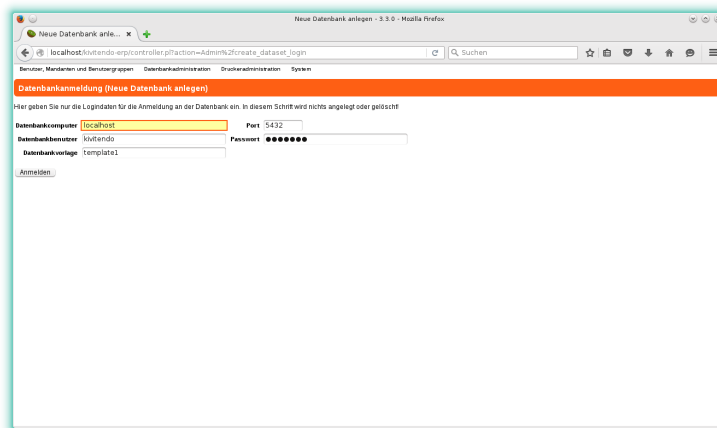
sich ein Benutzer dann an einem Mandanten an, so erhält er alle Rechte von allen denjenigen Gruppen, die zum Einen dem Mandanten zugeordnet sind und in denen der Benutzer zum Anderen Mitglied ist,

Die Reihenfolge, in der Datenbanken, Mandanten, Gruppen und Benutzer angelegt werden, kann im Prinzip beliebig gewählt werden. Die folgende Reihenfolge beinhaltet die wenigsten Arbeitsschritte:

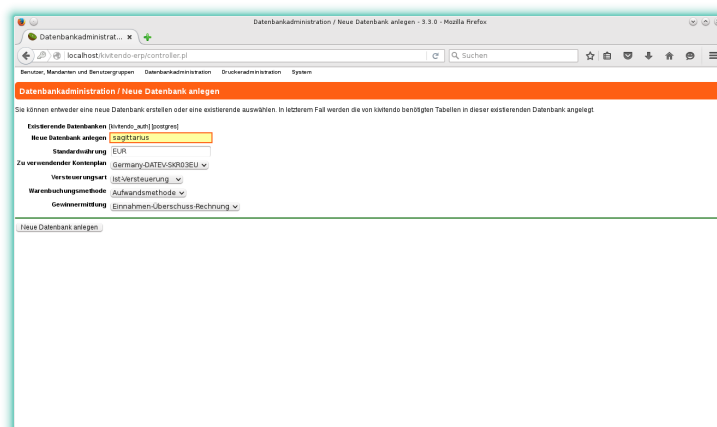
1. Datenbank anlegen
2. Gruppen anlegen
3. Benutzer anlegen und Gruppen als Mitglied zuordnen
4. Mandanten anlegen und Gruppen sowie Benutzer zuweisen

2.1.3. Datenbanken anlegen

Zuerst muss eine Datenbank angelegt werden. Verwenden Sie für den Datenbankzugriff den vorhin angelegten Benutzer (in unseren Beispielen ist dies 'kivitando').



Beim Anlegen der Datenbank wird auch zugleich der Kontenrahmen, Steuerungsart, Warenbuchungsmethode und Gewinnermittlung festgelegt. Diese Parameter können nur durch löschen und neu anlegen einer Datenbank

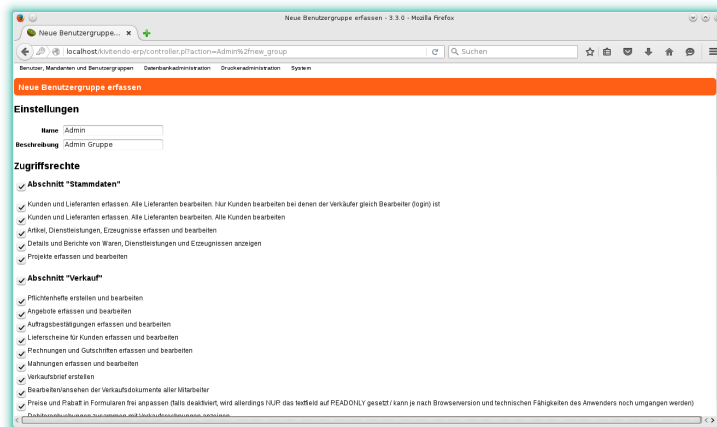


2.1.4. Gruppen anlegen

Eine Gruppe wird in der Gruppenverwaltung angelegt. Ihr muss ein Name gegeben werden, eine Beschreibung ist hingegen optional. Nach dem Anlegen können Sie die verschiedenen Bereiche wählen, auf die Mitglieder dieser Gruppe Zugriff haben sollen.

Benutzergruppen werden zwar in der Authentifizierungsdatenbank gespeichert, gelten aber nicht automatisch für alle Mandanten. Der Administrator legt vielmehr fest, für welche Mandanten eine Gruppe gültig ist. Dies kann entweder beim Bearbeiten der Gruppe geschehen ("diese Gruppe ist gültig für Mandanten X, Y und Z"), oder aber wenn man einen Mandanten bearbeitet ("für diesen Mandanten sind die Gruppen A, B und C gültig").

Wurden bereits Benutzer angelegt, so können hier die Mitglieder dieser Gruppe festgelegt werden ("in dieser Gruppe sind die Benutzer X, Y und Z Mitglieder"). Dies kann auch nachträglich beim Bearbeiten eines Benutzers geschehen ("dieser Benutzer ist Mitglied in den Gruppen A, B und C").

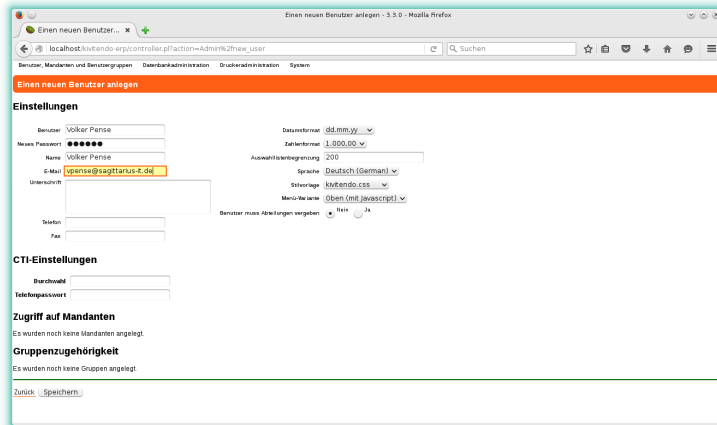


2.1.5. Benutzer anlegen

Beim Anlegen von Benutzern werden für viele Parameter Standardeinstellungen vorgenommen, die den Gepflogenheiten des deutschen Raumes entsprechen.

Zwingend anzugeben ist der Loginname. Wenn die Passwortauthentifizierung über die Datenbank eingestellt ist, so kann hier auch das Benutzerpasswort gesetzt bzw. geändert werden. Ist hingegen die LDAP-Authentifizierung aktiv, so ist das Passwort-Feld deaktiviert.

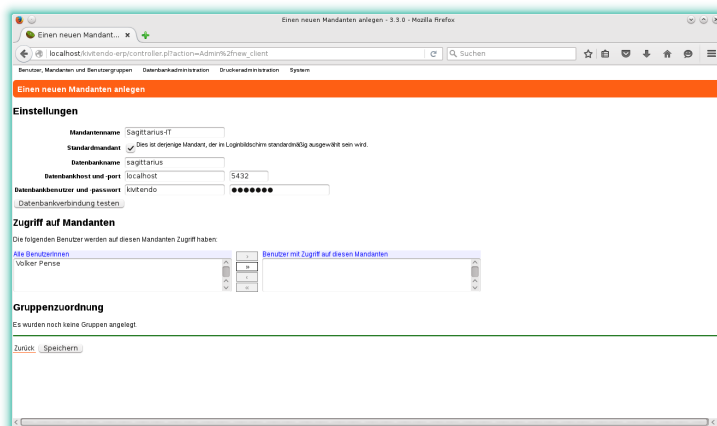
Hat man bereits Mandanten und Gruppen angelegt, so kann hier auch konfiguriert werden, auf welche Mandanten der Benutzer Zugriff hat bzw. in welchen Gruppen er Mitglied ist. Beide Zuweisungen können sowohl beim Benutzer vorgenommen werden ("dieser Benutzer hat Zugriff auf Mandanten X, Y, Z" bzw. "dieser Benutzer ist Mitglied in Gruppen X, Y und Z") als auch beim Mandanten ("auf diesen Mandanten haben Benutzer A, B und C Zugriff") bzw. bei der Gruppe ("in dieser Gruppe sind Benutzer A, B und C Mitglieder").



2.1.6. Mandanten anlegen

Ein Mandant besteht aus Administrationssicht primär aus einem eindeutigen Namen. Weiterhin wird hier hinterlegt, welche Datenbank als Mandantendatenbank benutzt wird. Hier müssen die Zugriffsdaten einer der eben angelegten Datenbanken eingetragen werden.

Hat man bereits Benutzer und Gruppen angelegt, so kann hier auch konfiguriert werden, welche Benutzer Zugriff auf den Mandanten haben bzw. welche Gruppen für den Mandanten gültig sind. Beide Zuweisungen können sowohl beim Mandanten vorgenommen werden ("auf diesen Mandanten haben Benutzer X, Y und Z Zugriff" bzw. "für diesen Mandanten sind die Gruppen X, Y und Z gültig") als auch beim Benutzer ("dieser Benutzer hat Zugriff auf Mandanten A, B und C") bzw. bei der Gruppe ("diese Gruppe ist für Mandanten A, B und C gültig").



2.2. Drucker- und Systemverwaltung

Im Administrationsmenü gibt es ferner noch die beiden Menüpunkte Druckeradministration und System.

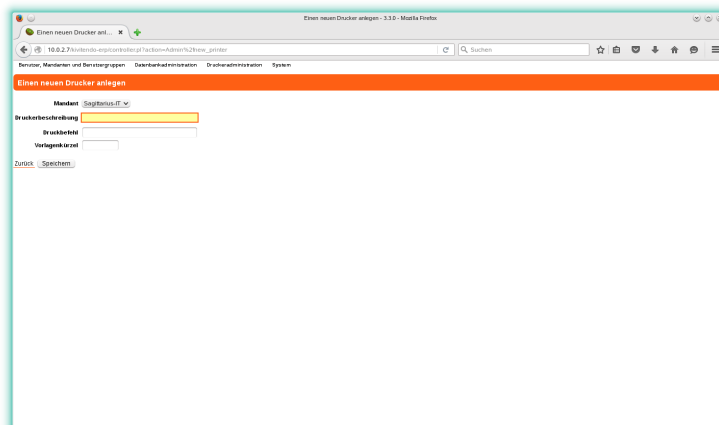
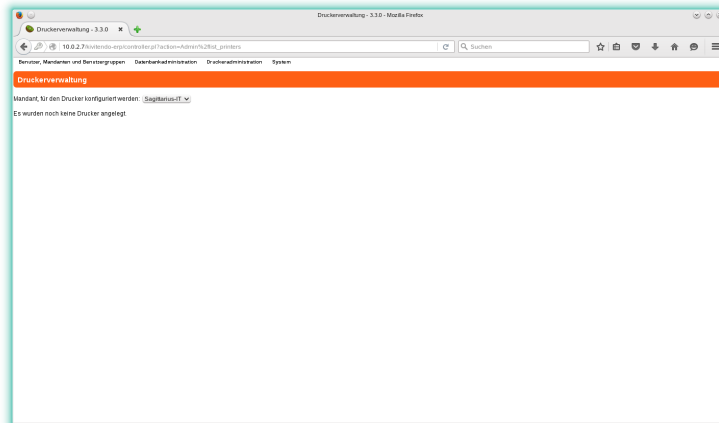
2.2.1. Druckeradministration

2.2.1.1 Drucker anlegen in YAST

Unter openSUSE/SLES werden Drucker am einfachsten in YAST angelegt. Die Verwendung von YAST geht nur als root. Die genaue Beschreibung erhalten sie in der SUSE Dokumentation.

Sagittarius-IT, Pfungstädter Str. 22, 64297 Darmstadt

Der Drucker wird im Administration → Druckeradministration angelegt.



Unter dem Menüpunkt Druckeradministration lassen sich beliebig viele "Druckbefehle" im System verwalten. Diese Befehle werden mandantenweise zugeordnet. Unter Druckerbeschreibung wird der Namen des Druckbefehls festgelegt, der dann in der Druckerauswahl des Belegs angezeigt wird.

Unter Druckbefehl definiert man den eigentlichen Druckbefehl, der direkt auf dem Webserver ausgeführt wird, bspw. 'lpr -P meinDrucker' oder ein kompletter Pfad zu einem Skript (/usr/local/src/kivitendo/scripts/pdf_druck_in_verzeichnis.sh). Wird ferner noch ein optionales Vorlagenkürzel verwendet, wird dieses Kürzel bei der Auswahl der Druckvorlagendatei mit einem Unterstrich ergänzt, ist bspw. das Kürzel 'epson_drucker' definiert, so wird beim Ausdruck eines Angebots folgende Vorlage geparkt: sales_quotation_epson_drucker.tex.

2.2.2. System sperren / entsperren

Unter dem Menüpunkt System gibt es den Eintrag 'Installation sperren/entsperren'. Setzt man diese Sperre so ist der Zugang zu der gesamten kivitendo Installation gesperrt.

Falls die Sperre gesetzt ist, erscheint anstelle der Anmeldemaske die Information: 'kivitendo ist momentan zwecks Wartungsarbeiten nicht zugänglich.'

Wichtig zu erwähnen ist hierbei noch, dass sich kivitendo automatisch 'sperrt', falls es bei einem Versionsupdate zu einem Datenbankfehler kam. Somit kann hier nicht aus Versehen mit einem

inkonsistenten Datenbestand weitergearbeitet werden.

2.3. E-Mail-Versand aus kivitendo heraus

kivitendo kann direkt aus dem Programm heraus E-Mails versenden, z.B. um ein Angebot direkt an einen Kunden zu verschicken. Damit dies funktioniert, muss eingestellt werden, über welchen Server die E-Mails verschickt werden sollen. kivitendo unterstützt dabei zwei Mechanismen: Versand über einen lokalen E-Mail-Server (z.B. mit Postfix™ oder Exim™, was auch die standardmäßig aktive Methode ist) sowie Versand über einen SMTP-Server (z.B. der des eigenen Internet-Providers).

Welche Methode und welcher Server verwendet werden, wird über die Konfigurationsdatei `config/kivitendo.conf` festgelegt. Dort befinden sich alle Einstellungen zu diesem Thema im Abschnitt `'[mail_delivery]'`.

2.3.1. Versand über lokalen E-Mail-Server

Diese Methode bietet sich an, wenn auf dem Server, auf dem kivitendo läuft, bereits ein funktionsfähiger E-Mail-Server wie z.B. Postfix™, Exim™ oder Sendmail™ läuft.

Um diese Methode auszuwählen, muss der Konfigurationsparameter `'method = sendmail'` gesetzt sein. Dies ist gleichzeitig der Standardwert, falls er nicht verändert wird.

Um zu kontrollieren, wie das Programm zum Einliefern gestartet wird, dient der Parameter `'sendmail = ...'`. Der Standardwert verweist auf das Programm `/usr/bin/sendmail`, das bei allen oben genannten E-Mail-Serverprodukten für diesen Zweck funktionieren sollte.

Die Konfiguration des E-Mail-Servers selber würde den Rahmen dieses sprengen. Hierfür sei auf die Dokumentation des E-Mail-Servers verwiesen.

2.3.2. Versand über einen SMTP-Server

Diese Methode bietet sich an, wenn kein lokaler E-Mail-Server vorhanden oder zwar einer vorhanden, dieser aber nicht konfiguriert ist.

Um diese Methode auszuwählen, muss der Konfigurationsparameter `'method = smtp'` gesetzt sein. Die folgenden Parameter dienen dabei der weiteren Konfiguration:

Hostname	Name oder IP-Adresse des SMTP-Servers. Standardwert: 'localhost'
Port	Portnummer. Der Standardwert hängt von der verwendeten Verschlüsselungsmethode ab. Gilt <code>'security = none'</code> oder <code>'security = tls'</code> , so ist 25 die Standardportnummer. Für <code>'security = ssl'</code> ist 465 die Portnummer. Muss normalerweise nicht geändert werden.
Security	Wahl der zu verwendenden Verschlüsselung der Verbindung mit dem Server. Standardwert ist 'none', wodurch keine Verschlüsselung verwendet wird. Mit 'tls' wird TLS-Verschlüsselung eingeschaltet, und mit 'ssl' wird Verschlüsselung via SSL eingeschaltet. Achtung: Für 'tls' und 'ssl' werden zusätzliche Perl-Module benötigt (siehe unten).
Login und password	Falls der E-Mail-Server eine Authentifizierung verlangt, so können mit diesen zwei Parametern der

	Benutzername und das Passwort angegeben werden. Wird Authentifizierung verwendet, so sollte aus Sicherheitsgründen auch eine Form von Verschlüsselung aktiviert werden.
--	---

Wird Verschlüsselung über TLS oder SSL aktiviert, so werden zusätzliche Perl-Module benötigt. Diese sind:

- TLS-Verschlüsselung: Modul Net::SSLGlue (Debian-Paketname libnet-sslglue-perl, Fedora Core: perl-Net-SSLGlue, openSUSE: perl-Net-SSLGlue)
- SSL-Verschlüsselung: Modul Net::SMTP::SSL (Debian-Paketname libnet-smtp-ssl-perl, Fedora Core: perl-Net-SMTP-SSL, openSUSE: perl-Net-SMTP-SSL)

2.4. Drucken mit kivitendo

Die Einrichtung eines Druckers ist in Kapitel 2.2 beschrieben. Dieses Kapitel erklärt die Erstellung von Vorlagen in LaTeX und OpenOffice/LibreOffice. Da OpenOffice nicht mehr Bestandteil von openSUSE/SLES ist, wird nur auf LibreOffice eingegangen.

Das Drucksystem von kivitendo benutzt von Haus aus LaTeX-Vorlagen. Um drucken zu können, braucht der Server ein geeignetes LaTeX System. Am einfachsten ist dazu eine texlive Installation.

Unter openSUSE kann man das System mit folgenden Befehl installieren (als root)

```
zypper install texlive-collection-latex kile
```

kile ist ein Programm zum erstellen und bearbeiten von LaTeX Dokumenten. Die Installation ist sehr umfangreich und benötigt ca. 1GB Platz auf der Festplatte.

kivitendo bringt drei alternative Vorlagensätze mit:

- RB
- f-tex
- rev-odt

Der ehemalige Druckvorlagensatz "Standard" wurde mit der Version 3.3 entfernt, da er nicht mehr gepflegt wurde.

2.4.1. Vorlagenverzeichnis anlegen

Es lässt sich ein initialer Vorlagensatz erstellen. Die LaTeX-System-Abhängigkeiten hierfür kann man prüfen mit:

```
./scripts/installation_check.pl -lv
```

Der Angemeldete Benutzer muss in einer Gruppe sein, die über das Recht "Konfiguration -> Mandantenverwaltung" verfügt. Siehe auch [Abschnitt 2.9.4, „Gruppen anlegen“](#).

Im Userbereich lässt sich unter: "System -> Mandantenverwaltung -> Verschiedenes" die Option "Neue Druckvorlagen aus Vorlagensatz erstellen" auswählen.

1. Vorlagen auswählen: Wählen Sie hier den Vorlagensatz aus, der kopiert werden soll (RB, f-

tex oder odt-rev.)

2. Neuer Name: Der Verzeichnisname für den neuen Vorlagensatz. Dieser kann im Rahmen der üblichen Bedingungen für Verzeichnisnamen frei gewählt werden.

Nach dem Speichern wird das Vorlagenverzeichnis angelegt und ist für den aktuellen Mandanten ausgewählt. Der gleiche Vorlagensatz kann, wenn er mal angelegt ist, bei mehreren Mandanten verwendet werden. Eventuell müssen Anpassungen (Logo, Erscheinungsbild, etc) noch vorgenommen werden. Den Ordner findet man im Dateisystem unter `./templates/[Neuer Name]`

2.4.2. Der Druckvorlagensatz RB

Hierbei handelt es sich um einen vollständigen LaTeX Dokumentensatz mit alternativem Design. Die odt oder html-Varianten sind nicht gepflegt.

Die konzeptionelle Idee der Vorlagen wird [hier](#) auf Folie 5 bis 10 vorgestellt. Informationen zur Anpassung an die eigenen Firmendaten finden sich in der Datei `Readme.tex` im Vorlagenverzeichnis.

Eine kurze Übersicht der Features:

- Mehrsprachenfähig, mit Deutscher und Englischer Übersetzung
- Zentrale Konfigurationsdateien, die für alle Belege benutzt werden, z.B. für Kopf- und Fußzeilen, und Infos wie Bankdaten
- mehrere vordefinierte Varianten für Logos/Hintergrundbilder
- Berücksichtigung für Steuerzonen "EU mit USt-ID Nummer" oder "Außerhalb EU"

2.4.3. f-tex

Ein Vorlagensatz, der in wenigen Minuten alle Dokumente zur Verfügung stellt.

2.4.3.1. Feature-Übersicht

- Keine Redundanz. Es wird ein- und dieselbe LaTeX-Vorlage für alle briefartigen Dokumente verwendet. Also Angebot, Rechnung, Proformarechnung, Lieferschein, aber eben nicht für Paketaufkleber etc.
- Leichte Anpassung an das Firmen-Layout durch Verwendung eines Hintergrund-PDFs. Dieses kann leicht mit dem eigenen Lieblingsprogramm erstellt werden (Openoffice, Inkscape, Gimp, Adobe*)
- Hintergrund-PDF umschaltbar auf "nur erste Seite" (Standard) oder "alle Seiten" (Option `"bgPdfFirstPageOnly"` in Datei `letter.lco`)
- Hintergrund-PDF für Ausdruck auf bereits bedrucktem Briefpapier abschaltbar. Es wird dann nur bei per E-Mail versendeten Dokumenten eingebunden (Option `"bgPdfEmailOnly"` in Datei `letter.lco`).
- Nutzung der Layout-Funktionen von LaTeX für Seitenumbruch, Wiederholung von Kopfzeilen, Zwischensummen etc. (danke an Kai-Martin Knaak für die Vorarbeit)
- Anzeige des Empfängerlandes im Adressfeld nur, wenn es vom Land des eigenen Unternehmens abweicht (also die Rechnung das Land verlässt).
- Multisprachfähig leicht um weitere Sprachen zu erweitern, alle Übersetzungen in der Datei `translatinos.tex`.
- Auflistung von Bruttopreisen für Endverbraucher.

2.4.3.2. Die Installation

- Vorlagenverzeichnis mit Option f-tex anlegen, siehe: [Vorlagenverzeichnis anlegen](#). Das Vorlagensystem funktioniert jetzt schon, hat allerdings noch einen Beispiel-Briefkopf.
- Erstelle eine pdf-Hintergrund Datei und verlinke sie nach `./letter_head.pdf`.
- Editiere den Bereich "settings" in der datei `letter.lco`.

oder etwas detaillierter:

Es wird eine Datei `sample.lco` erstellt und diese nach `letter.lco` verlinkt. Eigentlich ist dies die Datei die für die firmenspezifischen Anpassungen gedacht ist. Da die Einstiegshürde in LaTeX nicht ganz niedrig ist, wird in dieser Datei auf ein Hintergrund-PDF verwiesen. Ich empfehle über dieses PDF die persönlichen Layoutanpassungen vorzunehmen und `sample.lco` unverändert zu lassen. Die Anpassung über eine `*.lco`-Datei, die letztlich auf `letter.lco` verlinkt ist ist aber auch möglich.

Es wird eine Datei `sample_head.pdf` mit ausgeliefert, diese wird nach `letter_head.pdf` verlinkt. Damit gibt es schon mal eine funktionsfähige Vorlage. Schau Dir nach Abschluss der Installation die Datei `sample_head.pdf` an und erstelle ein entsprechendes PDF passend zum Briefkopf Deiner Firma, diese dann im Template Verzeichniss ablegen und statt `sample_head.pdf` nach `letter_head.pdf` verlinken.

Letzlich muss `letter_head.pdf` auf das passende Hintergrund-PDF verweisen, welches gewünschten Briefkopf enthält.

Es wird eine Datei `mydata.tex.example` ausgeliefert, die nach `mydata.tex` verlinkt ist. Bei verwendetem Hintergrund-PDF wird nur der Eintrag für das Land verwendet. Die Datei muss also nicht angefasst werden. Die anderen Werte sind für das Modul 'lp' (Label Print in erp - zur Zeit nicht im öffentlichen Zweig).

Alle Anpassungen zum Briefkopf, Fusszeilen, Firmenlogos, etc. sollten über die Hintergrund-PDF-Datei oder die `*.lco`-Datei erfolgen.

2.4.3.3. f-tex Funktionsübersicht

Das Konzept von kivitendo sieht vor, für jedes Dokument (Auftragsbestätigung, Lieferschein, Rechnung, etc.) eine LaTeX-Vorlage vorzuhalten, dies ist sehr wartungsunfreundlich. Auch das Einlesen einer einheitlichen Quelle für den Briefkopf bringt nur bedingte Vorteile, da hier leicht die Pflege der Artikel-Tabellen aus dem Ruder läuft. Bei dem vorliegenden Ansatz wird für alle briefartigen Dokumente mit Artikel-Tabellen eine einheitliche LaTeX-Vorlage verwendet, welche über Codeweichen die Besonderheiten der jeweiligen Dokumente berücksichtigt:

- Tabellen mit oder ohne Preis
- Sprache der Tabellenüberschriften etc.
- Anpassung der Bezugs-Zeile (z.B. Rechnungsnummer versus Angebotsnummer)
- Darstellung von Brutto oder Netto-Preisen in der Auflistung (Endverbraucher versus gewerblicher Kunde)

Nachteil:

LaTeX hat ohnehin eine sehr steile Lehrkurve. Die Datei `letter.tex` ist sehr komplex und verstärkt damit diesen Effekt noch einmal erheblich. Wer LaTeX-Erfahrung hat, oder geübt ist Scriptsprachen nachzuvollziehen kann natürlich auch innerhalb der Tabellendarstellung gut persönliche Anpassungen vornehmen. Aber man kann sich hier bei Veränderungen sehr schnell heftig in den Fuss schiessen.

Wer nicht so tief in die Materie einsteigen will oder leicht zu frustrieren ist, sollte sein Hintergrund-PDF auf Basis der mitgelieferten Datei `sample_head.pdf` erstellen, und sich an der Form der dargestellten Tabellen,

wie sie ausgeliefert werden, erfreuen.

Kleiner Tipp: Nicht zu viel auf einmal wollen, lieber kleine, kontinuierliche Schritte gehen.

2.4.3.4. Bruttopreise für Endverbraucher

Der auszuweisende Bruttopreis wird innerhalb der LaTeX-Umgebung berechnet. Es gibt zwar ein Feld, um bei Aufträgen "alle Preise Brutto" auszuwählen, aber:

- hierfür müssen die Preise auch in Brutto in der Datenbank stehen (ja - das lässt sich über die Preisgruppen und die Zuordnung einer Default-Preisgruppe handhaben)
- man darf beim Anlegen des Vorgangs nicht vergessen, dieses Häkchen zu setzen. (Das ist in der Praxis, wenn man sowohl Endverbraucher als auch Gewerbekunden beliefert, der eigentliche Knackpunkt)

Es gibt mit f-tex eine weitere Alternative. Die Information ob Brutto oder Nettorechnung wird mit den Zahlarten verknüpft. Zahlarten bei denen Rechnungen, Angebote, etc, in Brutto ausgegeben werden sollen, enden mit "_E" (für Endverbraucher). Falls identische Zahlarten für Gewerbekunden und Endverbraucher vorhanden sind, legt man diese einfach doppelt an (einmal mit der Namensendung "_E"). Gewinn:

- Die Entscheidung, ob Nettopreise ausgewiesen werden, ist nicht mehr fix mit einer Preisliste verbunden.
- Die Default-Zahlart kann im Kundendatensatz hinterlegt werden, und man muss nicht mehr daran denken, "alle Preise Netto" auszuwählen.
- Die Entscheidung, ob Netto- oder Bruttopreise ausgewiesen werden, kann direkt beim Drucken revidiert werden, ohne dass sich der Auftragswert ändert.

2.4.3.5. Lieferadressen

In Lieferscheinen kommen shipto*-Variablen im Adressfeld zum Einsatz. Wenn die shipto*-Variable leer ist, wird die entsprechende Adressvariable eingesetzt. Wenn also die Lieferadresse in Straße, Hausnummer und Ort abweicht, müssen auch nur diese Felder in der Lieferadresse ausgefüllt werden. Für den Firmennamen wird der Wert der Hauptadresse angezeigt.

2.4.4. Der Druckvorlagensatz rev-odt

Hierbei handelt es sich um einen Dokumentensatz der mit odt-Vorlagen erstellt wurde. Es gibt in dem Verzeichnis eine Readme-Datei, die eventuell aktueller als die Dokumentation hier ist. Die odt-Vorlagen in diesem Verzeichnis "rev-odt" wurden von revamp-it, Zürich erstellt und werden laufend aktualisiert. Ein paar der Formulierungen in den Druckvorlagen entsprechen dem Schweizer Sprachgebrauch, z.B. "Offerte" oder "allfällig".

Hinweis zum Einsatz des Feldes "Land" bei den Stammdaten für KundInnen und LieferantInnen, sowie bei Lieferadressen: Die in diesem Vorlagensatz vorhandenen Vorlagen erwarten für "Land" das entsprechende Kürzel, das in Adressen vor die Postleitzahl gesetzt wird. Das Feld kann auch komplett leer bleiben. Wer dies anders handhaben möchte, muss die Vorlagen entsprechend anpassen.

odt-Vorlagen können mit LibreOffice oder OpenOffice editiert und den eigenen Bedürfnissen angepasst werden. Wichtig beim Editieren von if-Blöcken ist, dass immer der gesamte Block überschrieben werden muss und nicht nur Teile davon, da dies sonst oft zu einer odt-Datei führt, die vom Parser nicht korrekt gelesen werden kann.

Zur Zeit gibt es in kivitendo noch keine Möglichkeit, odt-Vorlagen bei Mahnungen einzusetzen. Entsprechende Vorlagen sind deshalb nicht vorhanden.

Inwieweit es möglich ist, für die in Version 3.2.0 neu eingeführten Pflichtenhefte odt-Vorlagen zu erstellen,

sind wir am abklären. Wenn dies möglich ist, werden wir in Zukunft auch eine odt-Vorlage für Pflichtenhefte in diesem Vorlagensatz zur Verfügung stellen.

Fehlermeldungen, Anregungen und Wünsche bitte senden an: empfang@revamp-it.ch

2.4.5. Allgemeine Hinweise zu LaTeX Vorlagen

In den allermeisten Installationen sollte das Drucken jetzt schon funktionieren. Sollte ein Fehler auftreten, wirft TeX sehr lange Fehlerbeschreibungen, der eigentliche Fehler ist immer die erste Zeile, die mit einem Ausrufezeichen anfängt. Häufig auftretende Fehler sind zum Beispiel:

- ! LaTeX Error: File `eurosym.sty' not found. Die entsprechende LaTeX-Bibliothek wurde nicht gefunden. Das tritt vor allem bei Vorlagen aus der Community auf. Installieren Sie die entsprechenden Pakete.
- ! Package inputenc Error: Unicode char \u8:... set up for use with LaTeX. Dieser Fehler tritt auf, wenn sie versuchen mit einer Standardinstallation exotische utf8 Zeichen zu drucken. TeXLive unterstützt von Haus nur romanische Schriften und muss mit diversen Tricks dazu gebracht werden andere Zeichen zu akzeptieren. Adere TeX Systeme wie XeTeX schaffen hier Abhilfe.

Wird gar kein Fehler angezeigt, sondern nur der Name des Templates, heißt das normalerweise, dass das LaTeX Binary nicht gefunden wurde. Prüfen Sie den Namen in der Konfiguration (Standard: pdflatex), und stellen Sie sicher, dass pdflatex (oder das von Ihnen verwendete System) vom Webserver ausgeführt werden darf.

Wenn sich das Problem nicht auf Grund der Ausgabe im Webbrowser verifizieren lässt:

- editiere `[kivitendo-home]/config/kivitendo.conf` und ändere "keep_temp_files" auf 1
`keep_temp_files = 1;`
- bei fastcgi oder mod_perl den Webserver neu Starten
- Nochmal einen Druckversuch im Webfrontend auslösen
- wechsel in das users Verzeichnis von kivitendo
`cd [kivitendo-home]/users`
- LaTeX Suchpfad anpassen:
`export TEXINPUTS=".: [kivitendo-home]/templates/[aktuelles_template_verzeichniss]:"`
- Finde heraus, welche Datei kivitendo beim letzten Durchlauf erstellt hat
`ls -lahtr ./1*.tex`
Es sollte die letzte Datei ganz unten sein
- für besseren Hinweis auf Fehler texdatei nochmals übersetzen
`pdflatex ./1*.tex`
in der *.tex datei nach dem Fehler suchen.

2.5. OpenDocument-Vorlagen

kivitendo unterstützt die Verwendung von Vorlagen im OpenDocument-Format, wie es OpenOffice.org ab Version 2 erzeugt. kivitendo kann dabei sowohl neue OpenDocument-Dokumente als auch aus diesen direkt

Sagittarius-IT, Pfungstädter Str. 22, 64297 Darmstadt

PDF-Dateien erzeugen. Um die Unterstützung von OpenDocument-Vorlagen zu aktivieren muss in der Datei config/kivitendo.conf die Variable opendocument im Abschnitt print_templates auf '1' stehen. Dieses ist die Standardeinstellung.

Während die Erzeugung von reinen OpenDocument-Dateien keinerlei weitere Software benötigt, wird zur Umwandlung dieser Dateien in PDF OpenOffice.org benötigt. Soll dieses Feature genutzt werden, so muss neben OpenOffice.org ab Version 2 auch der "X virtual frame buffer" (xvfb) installiert werden. Bei Debian ist er im Paket "xvfb" enthalten. Andere Distributionen enthalten ihn in anderen Paketen.

Nach der Installation müssen in der Datei config/kivitendo.conf zwei weitere Variablen angepasst werden: openofficeorg_writer muss den vollständigen Pfad zur OpenOffice.org Writer-Anwendung enthalten. xvfb muss den Pfad zum "X virtual frame buffer" enthalten. Beide stehen im Abschnitt applications.

Zusätzlich gibt es zwei verschiedene Arten, wie kivitendo mit OpenOffice kommuniziert. Die erste Variante, die benutzt wird, wenn die Variable \$openofficeorg_daemon gesetzt ist, startet ein OpenOffice, das auch nach der Umwandlung des Dokumentes gestartet bleibt. Bei weiteren Umwandlungen wird dann diese laufende Instanz benutzt. Der Vorteil ist, dass die Zeit zur Umwandlung deutlich reduziert wird, weil nicht für jedes Dokument ein OpenOffice gestartet werden muss. Der Nachteil ist, dass diese Methode Python und die Python-UNO-Bindings benötigt, die Bestandteil von OpenOffice 2 sind.



Anmerkung

Für die Verbindung zu OpenOffice wird normalerweise der Python-Interpreter /usr/bin/python benutzt. Sollte dies nicht der richtige sein, so kann man mit zwei Konfigurationsvariablen entscheiden, welcher Python-Interpreter genutzt wird. Mit der Option python_uno aus dem Abschnitt applications wird der Interpreter selber festgelegt; sie steht standardmäßig auf dem eben erwähnten Wert /usr/bin/python.

Zusätzlich ist es möglich, Pfade anzugeben, in denen Python neben seinen normalen Suchpfaden ebenfalls nach Modulen gesucht wird, z.B. falls sich diese in einem gesonderten OpenOffice-Verzeichnis befinden. Diese zweite Variable heißt python_uno_path und befindet sich im Abschnitt environment. Sie ist standardmäßig leer. Werden hier mehrere Pfade angegeben, so müssen diese durch Doppelpunkte voneinander getrennt werden. Der Inhalt wird an den Python-Interpreter über die Umgebungsvariable PYTHONPATH übergeben.

Ist \$openofficeorg_daemon nicht gesetzt, so wird für jedes Dokument OpenOffice neu gestartet und die Konvertierung mit Hilfe eines Makros durchgeführt. Dieses Makro muss in der Dokumentenvorlage enthalten sein und "Standard.Conversion.ConvertSelfToPDF()" heißen. Die Beispielvorlage 'templates/mastertemplates/German/invoice.odt' enthält ein solches Makro, das in jeder anderen Dokumentenvorlage ebenfalls enthalten sein muss.

Als letztes muss herausgefunden werden, welchen Namen OpenOffice.org Writer dem Verzeichnis mit den Benutzereinstellungen gibt. Unter Debian ist dies momentan ~/.openoffice.org2. Sollte der Name bei Ihrer OpenOffice.org-Installation anders sein, so muss das Verzeichnis users/.openoffice.org2 entsprechend umbenannt werden. Ist der Name z.B. einfach nur .openoffice, so wäre folgender Befehl auszuführen:

```
mv users/.openoffice.org2 users/.openoffice
```

Dieses Verzeichnis, wie auch das komplette users-Verzeichnis, muss vom Webserver beschreibbar sein. Dieses wurde bereits erledigt (siehe [Manuelle Installation des Programmpaketes](#)), kann aber erneut überprüft werden, wenn die Konvertierung nach PDF fehlschlägt.

2.6. Konfiguration zur Einnahmenüberschussrechnung/Bilanzierung: EUR

2.6.1. Einführung

kivitendo besaß bis inklusive Version 2.6.3 einen Konfigurationsparameter namens eur, der sich in der Konfigurationsdatei config/kivitendo.conf (damals noch config/lx_office.conf) befand. Somit galt er für alle Mandanten, die in dieser Installation benutzt wurden.

Mit der nachfolgenden Version wurde der Parameter zum Einen in die Mandantendatenbank verschoben und dabei auch gleich in drei Einzelparameter aufgeteilt, mit denen sich das Verhalten genauer steuern lässt.

2.6.2. Konfigurationsparameter

Es gibt drei Parameter, die die Gewinnermittlungsart, Steuerungsart und die Warenbuchungsmethode regeln:

profit_determination	Dieser Parameter legt die Berechnungsmethode für die Gewinnermittlung fest. Er enthält entweder balance für Betriebsvermögensvergleich/Bilanzierung oder income für die Einnahmen-Überschuss-Rechnung.
accounting_method	Dieser Parameter steuert die Buchungs- und Berechnungsmethoden für die Steuerungsart. Er enthält entweder accrual für die Soll-Versteuerung oder cash für die Ist-Versteuerung.
inventory_system	Dieser Parameter legt die Warenbuchungsmethode fest. Er enthält entweder perpetual für die Bestandsmethode oder periodic für die Aufwandsmethode.

Zum Vergleich der Funktionalität bis und nach 2.6.3: eur = 1 bedeutete Einnahmen-Überschuss-Rechnung, Ist-Versteuerung und Aufwandsmethode. eur = 0 bedeutete hingegen Bilanzierung, Soll-Versteuerung und Bestandsmethode.

Die Konfiguration "eur" unter [system] in der [Konfigurationsdatei](#) config/kivitendo.conf wird nun nicht mehr benötigt und kann entfernt werden. Dies muss manuell geschehen.

2.6.3. Festlegen der Parameter

Beim Anlegen eines neuen Mandanten bzw. einer neuen Datenbank in der Administration können diese Optionen nun unabhängig voneinander eingestellt werden.

Beim Upgrade bestehender Mandanten wird eur ausgelesen und die Variablen werden so gesetzt, daß sich an der Funktionalität nichts ändert.

Die aktuelle Konfiguration wird unter Nummernkreise und Standardkonten unter dem neuen Punkt "Einstellungen" (read-only) angezeigt. Unter System -> Mandantenkonfiguration können die Einstellungen auch geändert werden. Dabei ist zu beachten, dass eine Änderung vorhandene Daten so belässt und damit evtl. die Ergebnisse verfälscht. Dies gilt vor Allem für die Warenbuchungsmethode (siehe auch [Bemerkungen zur Bestandsmethode](#)).

2.6.4. Bemerkungen zur Bestandsmethode

Die Bestandsmethode ist eigentlich eine sehr elegante Methode, funktioniert in kivitendo aber nur unter bestimmten Bedingungen: Voraussetzung ist, daß auch immer alle Einkaufsrechnungen gepflegt werden,

und man beim Jahreswechsel nicht mit einer leeren Datenbank anfängt, da bei jedem Verkauf anhand der gesamten Rechnungshistorie der Einkaufswert der Ware nach dem FIFO-Prinzip aus den Einkaufsrechnungen berechnet wird.

Die Bestandsmethode kann vom Prinzip her also nur funktionieren, wenn man mit den Buchungen bei Null anfängt, und man kann auch nicht im laufenden Betrieb von der Aufwandsmethode zur Bestandsmethode wechseln.

2.6.5. Bekannte Probleme

Bei bestimmten Berichten kann man derzeit noch individuell einstellen, ob man nach Ist- oder Sollversteuerung auswertet, und es werden im Code Variablen wie \$accrual oder \$cash gesetzt. Diese Codestellen wurden noch nicht angepasst, sondern nur die, wo bisher die Konfigurationsvariable `$.:lx_office_conf{system}->{eur}` ausgewertet wurde.

Es fehlen Hilfetext beim Neuanlegen eines Mandanten, was die Optionen bewirken, z.B. mit zwei Standardfällen.

2.7. SKR04 19% Umstellung für innergemeinschaftlichen Erwerb

2.7.1. Einführung

Die Umsatzsteuerumstellung auf 19% für SKR04 für die Steuerschlüssel "EU ohne USt-ID Nummer" ist erst 2010 erfolgt. kivitendo beinhaltet ein Upgradeskript, das das Konto 3804 automatisch erstellt und die Steuereinstellungen korrekt einstellt. Hat der Benutzer aber schon selber das Konto 3804 angelegt, oder gab es schon Buchungen im Zeitraum nach dem 01.01.2007 auf das Konto 3803, wird das Upgradeskript vorsichtshalber nicht ausgeführt, da der Benutzer sich vielleicht schon selbst geholfen hat und mit seinen Änderungen zufrieden ist. Die korrekten Einstellungen kann man aber auch per Hand ausführen. Nachfolgend werden die entsprechenden Schritte anhand von Screenshots dargestellt.

Für den Fall, daß Buchungen mit der Steuerschlüssel "EU ohne USt.-IdNr." nach dem 01.01.2007 erfolgt sind, ist davon auszugehen, dass diese mit dem alten Umsatzsteuersatz von 16% gebucht worden sind, und diese Buchungen sollten entsprechend kontrolliert werden.

2.7.2. Konto 3804 manuell anlegen

Die folgenden Schritte sind notwendig, um das Konto manuell anzulegen und zu konfigurieren. Zuerst wird in System -> Kontenübersicht -> Konto erfassen das Konto angelegt.



Als Zweites muss Steuergruppe 13 für Konto 3803 angepasst werden. Dazu unter System -> Steuern -> Bearbeiten den Eintrag mit Steuerschlüssel 13 auswählen und ihn wie im folgenden Screenshot angezeigt anpassen.



Als Drittes wird ein neuer Eintrag mit Steuerschlüssel 13 für Konto 3804 (19%) angelegt. Dazu unter System -> Steuern -> Erfassen auswählen und die Werte aus dem Screenshot übernehmen.



Als Nächstes sind alle Konten anzupassen, die als Steuerautomatikkonto die 3803 haben, sodass sie ab dem 1.1.2007 auch Steuerautomatik auf 3804 bekommen. Dies betrifft in der Standardkonfiguration die Konten 4315 und 4726. Als Beispiel für 4315 müssen Sie dazu unter System -> Kontenübersicht -> Konten anzeigen das Konto 4315 anklicken und die Einstellungen wie im Screenshot gezeigt vornehmen.



Als Letztes sollte die Steuerliste unter System -> Steuern -> Bearbeiten kontrolliert werden. Zum Vergleich der Screenshot.



2.8. Verhalten des Bilanzberichts

Bis Version 3.0 wurde "closedto" ("Bücher schließen zum") als Grundlage für das Startdatum benutzt. Schließt man die Bücher allerdings monatsweise führt dies zu falschen Werten.

In der Mandantenkonfiguration kann man dieses Verhalten genau einstellen indem man:

- weiterhin closed_to benutzt (Default, es ändert sich nichts zu vorher)
- immer den Jahresanfang nimmt (1.1. relativ zum Stichtag)
- immer die letzte Eröffnungsbuchung als Startdatum nimmt
 - mit Jahresanfang als Alternative wenn es keine EB-Buchungen gibt
 - oder mit "alle Buchungen" als Alternative"
- mit Jahresanfang als Alternative wenn es keine EB-Buchungen gibt
- immer alle Buchungen seit Beginn der Datenbank nimmt

Folgende Hinweise zu den Optionen: Das "Bücher schließen Datum" ist sinnvoll, wenn man nur komplette Jahre schließt. Bei Wirtschaftsjahr = Kalenderjahr entspricht dies aber auch dem Jahresanfang. "Alle Buchungen" kann z.B. sinnvoll sein wenn man ohne Jahresabschluß durchbucht. Eröffnungsbuchung mit "alle Buchungen" als Fallback ist z.B. sinnvoll, wenn man am sich Anfang des zweiten Buchungsjahres befindet, und noch keinen Jahreswechsel und auch noch keine EB-Buchungen hat. Bei den Optionen mit EB-Buchungen wird vorausgesetzt, daß diese immer am 1. Tag des Wirtschaftsjahres gebucht werden. Zur Sicherheit wird das Startdatum im Bilanzbericht jetzt zusätzlich zum Stichtag mit angezeigt. Das hilft auch bei der Kontrolle für den Abgleich mit der GuV.

2.9. Einstellungen pro Mandant

Einige Einstellungen können von einem Benutzer mit dem [Recht](#) "Administration (Für die Verwaltung der aktuellen Instanz aus einem Userlogin heraus)" gemacht werden. Diese Einstellungen sind dann für die aktuellen Mandanten-Datenbank gültig. Die Einstellungen sind unter System - > Mandantenkonfiguration erreichbar.

Bitte beachten Sie die Hinweise zu den einzelnen Einstellungen. Einige Einstellungen sollten nicht ohne Weiteres im laufenden Betrieb geändert werden (siehe auch [Bemerkungen zu Bestandsmethode](#)).

Die Einstellungen `show_bestbefore` und `payments_changeable` aus dem Abschnitt `features` und die Einstellungen im Abschnitt `datev_check` (sofern schon vorhanden) der [kivitendo-Konfigurationsdatei](#) werden bei einem Datenbankupdate einer älteren Version automatisch übernommen. Diese Einträge können danach aus der Konfigurationsdatei entfernt werden.

2.10. kivitendo ERP verwenden

Nach erfolgreicher Installation ist der Loginbildschirm unter folgender URL erreichbar:

<http://localhost/kivitendo-erp/login.pl>

Die Administrationsseite erreichen Sie unter:

<http://localhost/kivitendo-erp/controller.pl?action=Admin/login>

3. Features und Funktionen

3.1. Wiederkehrende Rechnungen

3.1.1. Einführung

Wiederkehrende Rechnungen werden als normale Aufträge definiert und konfiguriert, mit allen dazugehörigen Kunden- und Artikelangaben. Die konfigurierten Aufträge werden später automatisch in Rechnungen umgewandelt, so als ob man den Workflow benutzen würde, und auch die Auftragsnummer wird übernommen, sodass alle wiederkehrenden Rechnungen, die aus einem Auftrag erstellt wurden, später leicht wiederzufinden sind.

3.1.2. Konfiguration

Um einen Auftrag für wiederkehrende Rechnung zu konfigurieren, findet sich beim Bearbeiten des Auftrags ein neuer Knopf "Konfigurieren", der ein neues Fenster öffnet, in dem man die nötigen Parameter einstellen kann. Hinter dem Knopf wird außerdem noch angezeigt, ob der Auftrag als wiederkehrende Rechnung konfiguriert ist oder nicht.

Folgende Parameter kann man konfigurieren:

Status	Bei aktiven Rechnungen wird automatisch eine Rechnung erstellt, wenn die Periodizität erreicht ist (z.B. am Anfang eines neuen Monats). Ist ein Auftrag nicht aktiv, so werden für ihn auch keine wiederkehrenden Rechnungen
--------	---

	<p>erzeugt. Stellt man nach längerer nicht-aktiver Zeit einen Auftrag wieder auf aktiv, wird beim nächsten Periodenwechsel für alle Perioden, seit der letzten aktiven Periode, jeweils eine Rechnung erstellt. Möchte man dies verhindern, muss man vorher das Startdatum neu setzen.</p> <p>Für gekündigte Aufträge werden nie mehr Rechnungen erstellt. Man kann sich diese Aufträge aber gesondert in den Berichten anzeigen lassen.</p>
Periodizität	Ob monatlich, quartalsweise oder jährlich auf neue Rechnungen überprüft werden soll. Für jede Periode seit dem Startdatum wird überprüft, ob für die Periode (beginnend immer mit dem ersten Tag der Periode) schon eine Rechnung erstellt wurde. Unter Umständen können bei einem Startdatum in der Vergangenheit gleich mehrere Rechnungen erstellt werden.
Buchen auf	Das Forderungskonto, in der Regel "Forderungen aus Lieferungen und Leistungen". Das Gegenkonto ergibt sich aus den Buchungsgruppen der betreffenden Waren.
Startdatum	ab welchem Datum auf Rechnungserstellung geprüft werden soll
Enddatum	ab wann keine Rechnungen mehr erstellt werden sollen
Automatische Verlängerung um x Monate	Sollen die wiederkehrenden Rechnungen bei Erreichen des eingetragenen Enddatums weiterhin erstellt werden, so kann man hier die Anzahl der Monate eingeben, um die das Enddatum automatisch nach hinten geschoben wird.
Drucken	Sind Drucker konfiguriert, so kann man sich die erstellten Rechnungen auch gleich ausdrucken lassen.

Nach Erstellung der Rechnungen kann eine E-Mail mit Informationen zu den erstellten Rechnungen verschickt werden. Konfiguriert wird dies in der [Konfigurationsdatei](#) config/kivitendo.conf im Abschnitt [periodic_invoices].

3.1.3. Spezielle Variablen

Um die erzeugten Rechnungen individualisieren zu können, werden beim Umwandeln des Auftrags in eine Rechnung einige speziell formatierte Variablen durch für die jeweils aktuelle Abrechnungsperiode gültigen Werte ersetzt. Damit ist es möglich, z.B. den Abrechnungszeitraum explizit auszuweisen. Eine Variable hat dabei die Syntax `<%variablenname%>`.

Sofern es sich um eine Datumsvariable handelt, kann das Ausgabeformat weiter bestimmt werden, indem an den Variablenamen Formatoptionen angehängt werden. Die Syntax sieht dabei wie folgt aus: `<%variablenname FORMAT=Formatinformation%>`. Die zur Verfügung stehenden Formatinformationen werden unten genauer beschrieben.

Diese Variablen werden in den folgenden Elementen des Auftrags ersetzt:

- Bemerkungen
- Interne Bemerkungen
- Vorgangsbezeichnung
- In den Beschreibungs- und Langtextfeldern aller Positionen

Die zur Verfügung stehenden Variablen sind die Folgenden:

<%current_quarter%>, <%previous_quarter%>, <%next_quarter%>	Aktuelles, vorheriges und nächstes Quartal als Zahl zwischen 1 und 4.
<%current_month%>, <%previous_month%>, <%next_month%>	Aktueller, vorheriger und nächster Monat als Zahl zwischen 1 und 12.
<%current_month_long%>, <%previous_month_long%>, <%next_month_long%>	Aktueller, vorheriger und nächster Monat als Name (Januar, Februar etc.).
<%current_year%>, <%previous_year%>, <%next_year%>	Aktuelles, vorheriges und nächstes Jahr als vierstellige Jahreszahl (2013 etc.).
<%period_start_date%>, <%period_end_date%>	Formatiertes Datum des ersten und letzten Tages im Abrechnungszeitraum (z.B. bei quartalsweiser Abrechnung und im ersten Quartal von 2013 wären dies der 01.01.2013 und 31.03.2013).

Die individuellen Formatinformationen bestehen aus Paaren von Prozentzeichen und einem Buchstaben, welche beide zusammen durch den dazugehörigen Wert ersetzt werden. So wird z.B. %Y durch das vierstellige Jahr ersetzt. Alle möglichen Platzhalter sind:

%a	Der abgekürzte Wochentagsname.
%A	Der ausgeschriebene Wochentagsname.
%b	Der abgekürzte Monatsname.
%B	Der ausgeschriebene Monatsname.
%C	Das Jahrhundert (Jahr/100) als eine zweistellige Zahl.
%d	Der Monatstag als Zahl zwischen 01 und 31.
%D	Entspricht %m/%d/%y (amerikanisches Datumsformat).
%e	Wie %d (Monatstag als Zahl zwischen 1 und 31), allerdings werden führende Nullen durch Leerzeichen ersetzt.
%F	Entspricht %Y-%m-%d (das ISO-8601-Datumsformat).
%j	Der Tag im Jahr als Zahl zwischen 001 und 366 inklusive.
%m	Der Monat als Zahl zwischen 01 und 12 inklusive.
%u	Der Wochentag als Zahl zwischen 1 und 7 inklusive, wobei die 1 dem Montag entspricht.
%U	Die Wochennummer als Zahl zwischen 00 und 53 inklusive, wobei der erste Sonntag im Jahr das Startdatum von Woche 01 ist.
%V	Die ISO-8601:1988-Wochennummer als Zahl zwischen 01 und 53 inklusive, wobei Woche 01 die erste Woche, von der mindestens vier Tage im Jahr liegen

%w	Der Wochentag als Zahl zwischen 0 und 6 inklusive, wobei die 0 dem Sonntag entspricht.
%W	Die Wochennummer als Zahl zwischen 00 und 53 inklusive, wobei der erste Montag im Jahr das Startdatum von Woche 01 ist.
%y	Das Jahr als zweistellige Zahl zwischen 00 und 99 inklusive.
%Y	Das Jahr als vierstellige Zahl.
%%	Das Prozentzeichen selber.

Anwendungsbeispiel für die Ausgabe, von welchem Monat und Jahr bis zu welchem Monat und Jahr die aktuelle Abrechnungsperiode dauert: Abrechnungszeitraum: <%period_start_date FORMAT=%m/%Y%> bis <%period_end_date FORMAT=%m/%Y%>

3.1.4. Auflisten

Unter Verkauf->Berichte->Aufträge finden sich zwei neue Checkboxen, "Wiederkehrende Rechnungen aktiv" und "Wiederkehrende Rechnungen inaktiv", mit denen man sich einen Überblick über die wiederkehrenden Rechnungen verschaffen kann.

3.1.5. Erzeugung der eigentlichen Rechnungen

Die zeitliche und periodische Überprüfung, ob eine wiederkehrende Rechnung automatisch erstellt werden soll, geschieht durch den [Taskserver](#), einen externen Dienst, der automatisch beim Start des Servers gestartet werden sollte.

3.1.6. Erste Rechnung für aktuellen Monat erstellen

Will man im laufenden Monat eine monatlich wiederkehrende Rechnung inkl. des laufenden Monats starten, stellt man das Startdatum auf den Monatsanfang und wartet ein paar Minuten, bis der Taskserver den neu konfigurieren Auftrag erkennt und daraus eine Rechnung generiert hat. Alternativ setzt man das Startdatum auf den Monatsersten des Folgemonats und erstellt die erste Rechnung direkt manuell über den Workflow.

3.2. Bankerweiterung

3.2.1. Einführung

Die Beschreibung der Bankerweiterung befindet sich derzeit noch im Wiki und soll von dort später hierhin übernommen werden:

<http://redmine.kivitando-premium.de/projects/forum/wiki/Bankerweiterung>

3.3. Dokumentenvorlagen und verfügbare Variablen

3.3.1. Einführung

Dies ist eine Auflistung der Standard-Dokumentenvorlagen und aller zur Bearbeitung verfügbaren Variablen. Eine Variable wird in einer Vorlage durch ihren Inhalt ersetzt, wenn sie in der Form <%variablenname%> verwendet

wird. Für LaTeX- und HTML-Vorlagen kann man die Form dieser Tags auch verändern (siehe [Anfang und Ende der Tags verändern](#)).

Früher wurde hier nur über LaTeX gesprochen. Inzwischen unterstützt kivitendo aber auch OpenDocument-Vorlagen. Sofern es nicht ausdrücklich eingeschränkt wird, gilt das im Folgenden Gesagte für alle Vorlagenarten.

Insgesamt sind technisch gesehen eine ganze Menge mehr Variablen verfügbar als hier aufgelistet werden. Die meisten davon können allerdings innerhalb einer solchen Vorlage nicht sinnvoll verwendet werden. Wenn eine Auflistung dieser Variablen gewünscht ist, so kann diese wie folgt erhalten werden:

- SL/Form.pm öffnen und am Anfang die Zeile "use Data::Dumper;" einfügen.
- In Form.pm die Funktion parse_template suchen und hier die Zeile print(STDERR Dumper(\$self)); einfügen.
- Einmal per Browser die gewünschte Vorlage "benutzen", z.B. ein PDF für eine Rechnung erzeugen.
- Im error.log Apache steht die Ausgabe der Variablen \$self in der Form 'key' => 'value'.. Alle keys sind verfügbar.

3.3.2. Variablen ausgeben

Um eine Variable auszugeben, müssen sie einfach nur zwischen die Tags geschrieben werden, also z.B. <%variablenname%>.

Optional kann man auch mit Leerzeichen getrennte Flags angeben, die man aber nur selten brauchen wird. Die Syntax sieht also so aus: <%variablenname FLAG1 FLAG2%>. Momentan werden die folgenden Flags unterstützt:

- NOFORMAT gilt nur für Zahlenwerte und gibt den Wert ohne Formatierung, also ohne Tausendertrennzeichen mit mit einem Punkt als Dezimaltrennzeichen aus. Nützlich z.B., wenn damit in der Vorlage z.B. von LaTeX gerechnet werden soll.
- NOESCAPE unterdrückt das Escapen von Sonderzeichen für die Vorlagensprache. Wenn also in einer Variablen bereits gültiger LaTeX-Code steht und dieser von LaTeX auch ausgewertet und nicht wortwörtlich angezeigt werden soll, so ist dieses Flag sinnvoll.

Beispiel:

```
<%quototal NOFORMAT%>
```

3.3.3. Verwendung in Druckbefehlen

In der Administration können Drucker definiert werden. Auch im dort eingebbaren Druckbefehl können die hier aufgelisteten Variablen und Kontrollstrukturen verwendet werden. Ihr Inhalt wird dabei nach den Regeln der gängigen Shells formatiert, sodass Sonderzeichen wie `...` nicht zu unerwünschtem Verhalten führen.

Dies erlaubt z.B. die Definition eines Faxes als Druckerbefehl, für das die Telefonnummer eines Ansprechpartners als Teil der Kommandozeile verwendet wird. Für ein fiktives Kommando könnte das z.B. wie folgt aussehen:

```
send_fax --number <%if cp_phone2%><%cp_phone2%><%else%><%cp_phone1%><%end%>
```

3.3.4. Anfang und Ende der Tags verändern

Der Standardstil für Tags sieht vor, dass ein Tag mit dem Kleinerzeichen und einem Prozentzeichen beginnt und mit dem Prozentzeichen und dem Größerzeichen endet, beispielsweise <%customer%>. Da diese Form aber z.B. in LaTeX zu Problemen führen kann, weil das Prozentzeichen dort Kommentare einleitet, kann pro HTML- oder LaTeX-Dokumentenvorlage der Stil umgestellt werden.

Dazu werden in die Datei Zeilen geschrieben, die mit dem für das Format gültigen Kommentarzeichen anfangen, dann config: enthalten, die entsprechende Option setzen und bei HTML-Dokumentenvorlagen mit dem Kommentarendzeichen enden. Beispiel für LaTeX:

```
% config: tag-style=($ $)
```

Dies würde kivitendo dazu veranlassen, Variablen zu ersetzen, wenn sie wie folgt aussehen: (\$customer\$). Das äquivalente Beispiel für HTML-Dokumentenvorlagen sieht so aus:

```
<!-- config: tag-style=($ $) -->
```

3.3.5. Zuordnung von den Dateinamen zu den Funktionen

Diese folgende kurze Auflistung zeigt, welche Vorlage bei welcher Funktion ausgelesen wird. Dabei ist die Dateierdung ".ext" geeignet zu ersetzen: ".tex" für LaTeX-Vorlagen und ".odt" für OpenDocument-Vorlagen.

bin_list.ext	Lagerliste
check.ext	?
invoice.ext	Rechnung
packing_list.ext	Packliste
pick_list.ext	Sammelliste
purchase_delivery_order.ext	Lieferschein (Einkauf)
purchase_order.ext	Bestellung an Lieferanten
request_quotation.ext	Anfrage an Lieferanten
sales_delivery_order.ext	Lieferschein (Verkauf)
sales_order.ext	Bestellung
sales_quotation.ext	Angebot an Kunden
zahlungserinnerung.ext	Mahnung (Dateiname im Programm konfigurierbar)
zahlungserinnerung_invoice.ext	Rechnung über Mahngebühren (Dateiname im Programm konfigurierbar)

3.3.6. Sprache, Drucker und E-Mail

Angeforderte Sprache und Drucker Kürzel in den Dateinamen mit eingearbeitet. So wird aus der Vorlage sales_order.ext bei Sprache de und Drucker Kürzel lpr2 der Vorlagename sales_order_de_lpr2.ext. Zusätzlich können für E-Mails andere Vorlagen erstellt werden, diese bekommen dann noch das Kürzel_email, der vollständige Vorlagename wäre dann sales_order_email_de_lpr2.ext. In allen Fällen kann eine Standarddatei default.ext hinterlegt werden. Diese wird verwendet, wenn keine der anderen Varianten gefunden wird.

Die vollständige Suchreihenfolge für einen Verkaufsauftrag mit der Sprache "de" und dem Drucker "lpr2", der per E-Mail im Format PDF verschickt wird, ist:

1. sales_order_email_de_lpr2.tex
2. sales_order_de_lpr2.tex

3. sales_order.tex

4. default.tex

Die kurzen Varianten dieser Vorlagentitel müssen dann entweder Standardwerte anzeigen, oder die angeforderten Werte selbst auswerten, siehe dazu [Metainformationen zur angeforderten Vorlage](#).

3.3.7. Allgemeine Variablen, die in allen Vorlagen vorhanden sind

3.3.7.1. Metainformationen zur angeforderten Vorlage

Diese Variablen liefern Informationen darüber welche Variante einer Vorlage der Benutzer angefragt hat. Sie sind nützlich für Vorlagenautoren, die aus einer zentralen Layoutvorlage die einzelnen Formulare einbinden möchten.

template_meta.formname	Basisname der Vorlage. Identisch mit der Zurordnung zu den Dateinamen ohne die Erweiterung. Ein Verkaufsauftrag enthält hier sales_order.
template_meta.language.description	Beschreibung der verwendeten Sprache
template_meta.language.template_code	Vorlagenürzel der verwendeten Sprache, identisch mit dem Kürzel das im Dateinamen verwendetet wird.
template_meta.language.output_numberformat	Zahlenformat der verwendeten Sprache in der Form "1.000,00". Experimentell! Nur interessant für Vorlagen die mit unformatierten Werten arbeiten.
template_meta.language.output_dateformat	Datumsformat der verwendeten Sprache in der Form "dd.mm.yyyy". Experimentell! Nur interessant für Vorlagen die mit unformatierten Werten arbeiten.
template_meta.format	Das angeforderte Format. Kann im Moment die Werte pdf, postscript, html, opendocument, opendocume nt_pdf und excel enthalten.
template_meta.extension	Dateierweiterung, wie im Dateinamen. Wird aus format entschieden.
template_meta.media	Ausgabemedium. Kann zur Zeit die Werte screen für Bildschirm, email für E-Mail (triggert das_email Kürzel im Dateinamen), printer für Drucker, und queue für Warteschlange enthalten.
template_meta.printer.description	Beschreibung des ausgewählten Druckers
template_meta.printer.template_code	Vorlagenürzel des ausgewählten Druckers, identisch mit dem Kürzel das im Dateinamen verwendetet wird.
template_meta.tmpfile	Datei-Prefix für temporäre Dateien.

3.3.7.2. Stammdaten von Kunden und Lieferanten

account_number	Kontonummer
----------------	-------------

bank	Name der Bank
bank_code	Bankleitzahl
bic	Bank-Identifikations-Code (Bank Identifier Code, BIC)
business	Kunden-/Lieferantentyp
city	Stadt
contact	Kontakt
country	Land
c_vendor_id	Lieferantenummer beim Kunden (nur Kunden)
v_customer_id	Kundenummer beim Lieferanten (nur Lieferanten)
cp_email	Email des Ansprechpartners
cp_givenname	Vorname des Ansprechpartners
cp_greeting	Anrede des Ansprechpartners
cp_name	Name des Ansprechpartners
cp_phone1	Telefonnummer 1 des Ansprechpartners
cp_phone2	Telefonnummer 2 des Ansprechpartners
cp_title	Titel des Ansprechpartners
creditlimit	Kreditlimit
customeremail	Email des Kunden
customerfax	Faxnummer des Kunden
customernotes	Bemerkungen beim Kunden
customernumber	Kundenummer
customerphone	Telefonnummer des Kunden
discount	Rabatt
email	Emailadresse
fax	Faxnummer
greeting	Anrede
homepage	Homepage
iban	Internationale Kontonummer (International Bank Account Number,

	IBAN)
language	Sprache
name	Firmenname
payment_description	Name der Zahlart
payment_terms	Zahlungskonditionen
phone	Telefonnummer
shiptocity	Stadt (Lieferadresse) *
shiptocontact	Kontakt (Lieferadresse) *
shiptocountry	Land (Lieferadresse) *
shiptodepartment1	Abteilung 1 (Lieferadresse) *
shiptodepartment2	Abteilung 2 (Lieferadresse) *
shiptoemail	Email (Lieferadresse) *
shiptofax	Fax (Lieferadresse) *
shiptoname	Firmenname (Lieferadresse) *
shiptophone	Telefonnummer (Lieferadresse) *
shiptostreet	Straße und Hausnummer (Lieferadresse) *
shiptozipcode	Postleitzahl (Lieferadresse) *
street	Straße und Hausnummer
taxnumber	Steuernummer
ustid	Umsatzsteuer-Identifikationsnummer
vendoremail	Email des Lieferanten
vendorfax	Faxnummer des Lieferanten
vendornotes	Bemerkungen beim Lieferanten
vendornumber	Lieferantennummer
vendorphone	Telefonnummer des Lieferanten
zipcode	Postleitzahl



Anmerkung

Anmerkung: Sind die shipto*-Felder in den Stammdaten nicht eingetragen, so

haben die Variablen shipto* den gleichen Wert wie die die entsprechenden Variablen der Lieferdaten. Das bedeutet, dass sich einigeshipto*-Variablen so nicht in den Stammdaten wiederfinden sondern schlicht Kopien der Lieferdatenvariablen sind (z.B. shiptocontact).

3.3.7.3. Informationen über den Bearbeiter

employee_address	Adressfeld
employee_businessnumber	Firmennummer
employee_company	Firmenname
employee_co_ustid	Umsatzsteuer-Identifikationsnummer
employee_duns	DUNS-Nummer
employee_email	Email
employee_fax	Fax
employee_name	voller Name
employee_signature	Signatur
employee_taxnumber	Steuernummer
employee_tel	Telefonnummer

3.3.7.4. Informationen über den Verkäufer

salesman_address	Adressfeld
salesman_businessnumber	Firmennummer
salesman_company	Firmenname
salesman_co_ustid	Umsatzsteuer-Identifikationsnummer
salesman_duns	DUNS-Nummer
salesman_email	Email
salesman_fax	Fax
salesman_name	voller Name
salesman_signature	Signatur
salesman_taxnumber	Steuernummer
salesman_tel	Telefonnummer

3.3.7.5. Variablen für die einzelnen Steuern

tax	Steuer
-----	--------

taxbase	zu versteuernder Betrag
taxdescription	Name der Steuer
taxrate	Steuersatz

3.3.7.6. Variablen für Lieferbedingungen

delivery_term	Datenbank-Objekt der Lieferbedingung
delivery_term.description	Beschreibung der Lieferbedingung
delivery_term.description_long	Langtext bzw. übersetzter Langtext der Lieferbedingung

3.3.8. Variablen in Rechnungen

3.3.8.1. Allgemeine Variablen

creditremaining	Verbleibender Kredit
currency	Währung
cusordnumber	Bestellnummer beim Kunden
deliverydate	Lieferdatum
duedate	Fälligkeitsdatum
globalprojectnumber	Projektnummer des ganzen Beleges
globalprojectdescription	Projekbeschreibung des ganzen Beleges
intnotes	Interne Bemerkungen
invdate	Rechnungsdatum
invnumber	Rechnungsnummer
invtotal	gesamter Rechnungsbetrag
notes	Bemerkungen der Rechnung
orddate	Auftragsdatum
ordnumber	Auftragsnummer, wenn die Rechnung aus einem Auftrag erstellt wurde
payment_description	Name der Zahlart
payment_terms	Zahlungskonditionen
quodate	Angebotsdatum
quonumber	Angebotsnummer
shippingpoint	Versandort
shipvia	Transportmittel

subtotal	Zwischensumme aller Posten ohne Steuern
total	Restsumme der Rechnung (Summe abzüglich bereits bezahlter Posten)
transaction_description	Vorgangsbezeichnung
transdate	Auftragsdatum wenn die Rechnung aus einem Auftrag erstellt wurde

3.3.8.2. Variablen für jeden Posten auf der Rechnung

bin	Stellage
description	Artikelbeschreibung
cusordnumber_oe	Bestellnummer des Kunden aus dem Auftrag, aus dem der Posten ursprünglich stammt (nur Verkauf)
discount	Rabatt als Betrag
discount_sub	Zwischensumme mit Rabatt
donumber_do	Lieferscheinnummer des Lieferscheins, aus dem die Position ursprünglich stammt, wenn die Rechnung im Rahmen des Workflows aus einem Lieferschein erstellt wurde.
drawing	Zeichnung
ean	EAN-Code
image	Grafik
linetotal	Zeilensumme (Anzahl * Einzelpreis)
longdescription	Langtext
microfiche	Mikrofilm
netprice	Alternative zu sellprice, aber netprice entspricht dem effektiven Einzelpreis und beinhaltet Zeilenrabatt und Preisfaktor. netprice wird rückgerechnet aus Zeilensumme / Menge. Diese Variable ist nützlich, wenn man den gewährten Rabatt in der Druckvorlage nicht anzeigen möchte, aber Menge * Einzelpreis trotzdem die angezeigte Zeilensumme ergeben soll. netprice hat nichts mit Netto/Brutto im Sinne von Steuern zu tun.
nodiscount_linetotal	Zeilensumme ohne Rabatt
nodiscount_sub	Zwischensumme ohne Rabatt
number	Artikelnummer
ordnumber_oe	Auftragsnummer des Originalauftrags, aus dem der Posten ursprünglich stammt. Nützlich, wenn die Rechnung aus mehreren

	Lieferscheinen zusammengefasst wurde, oder wenn zwischendurch eine Sammelauftrag aus mehreren Aufträgen erstellt wurde. In letzterem Fall wird die ursprüngliche Auftragsnummer angezeigt.
p_discount	Rabatt in Prozent
partnotes	Die beim Artikel gespeicherten Bemerkungen
partsgroup	Warengruppe
price_factor	Der Preisfaktor als Zahl, sofern einer eingestellt ist
price_factor_name	Der Name des Preisfaktors, sofern einer eingestellt ist
projectnumber	Projektnummer
projectdescription	Projektbeschreibung
qty	Anzahl
reqdate	Lieferdatum
runningnumber	Position auf der Rechnung (1, 2, 3...)
sellprice	Verkaufspreis
serialnumber	Seriennummer
tax_rate	Steuersatz
transdate_do	Datum des Lieferscheins, wenn die Rechnung im Rahmen des Workflows aus einem Lieferschein stammte.
transdate_oe	Datum des Auftrags, wenn die Rechnung im Rahmen des Workflows aus einem Auftrag erstellt wurde. Wenn es Sammelaufträge gab wird das Datum des ursprünglichen Auftrags genommen.
transdate_quo	Datum des Angebots, wenn die Position im Rahmen des Workflows aus einem Angebot stammte.
unit	Einheit
weight	Gewicht

Für jeden Posten gibt es ein Unterarray mit den Informationen über Lieferanten und Lieferantenartikelnummer. Diese müssen mit einer foreach-Schleife ausgegeben werden, da für jeden Artikel mehrere Lieferanteninformationen hinterlegt sein können. Die Variablen dafür lauten:

make	Lieferant
model	Lieferantenartikelnummer

3.3.8.3. Variablen für die einzelnen Zahlungseingänge

payment	Betrag
---------	--------

paymentaccount	Konto
paymentdate	Datum
paymentmemo	Memo
paymentsource	Beleg

3.3.8.4. Benutzerdefinierte Kunden- und Lieferantenvariablen

Die vom Benutzer definierten Variablen für Kunden und Lieferanten stehen beim Ausdruck von Einkaufs- und Verkaufsbelegen ebenfalls zur Verfügung. Ihre Namen setzen sich aus dem Präfix vc_cvar_ und dem vom Benutzer festgelegten Variablennamen zusammen.

Beispiel: Der Benutzer hat eine Variable namens number_of_employees definiert, die die Anzahl der Mitarbeiter des Unternehmens enthält. Diese Variable steht dann unter dem Namen vc_cvar_number_of_employees zur Verfügung.

3.3.9. Variablen in Mahnungen und Rechnungen über Mahngebühren

3.3.9.1. Namen der Vorlagen

Die Namen der Vorlagen werden im System-Menü vom Benutzer eingegeben. Wird für ein Mahnlevel die Option zur automatischen Erstellung einer Rechnung über die Mahngebühren und Zinsen aktiviert, so wird der Name der Vorlage für diese Rechnung aus dem Vorlagennamen für diese Mahnstufe mit dem Zusatz _invoice gebildet. Weiterhin werden die Kürzel für die ausgewählte Sprache und den ausgewählten Drucker angehängt.

3.3.9.2. Allgemeine Variablen in Mahnungen

Die Variablen des Verkäufers stehen wie gewohnt als employee_... zur Verfügung. Die Adressdaten des Kunden stehen als Variablen name, street, zipcode, city, country, department_1, department_2, und email zur Verfügung.

Weitere Variablen beinhalten:

dunning_date	Datum der Mahnung
dunning_duedate	Fälligkeitsdatum für diese Mahnung
dunning_id	Mahnungsnummer
fee	Kummulative Mahngebühren
interest_rate	Zinssatz per anno in Prozent
total_amount	Gesamter noch zu zahlender Betrag als fee + total_interest + total_open_amount
total_interest	Zinsen per anno über alle Rechnungen
total_open_amount	Summe über alle offene Beträge der Rechnungen

3.3.9.3. Variablen für jede gemahnte Rechnung in einer Mahnung

dn_amount	Rechnungssumme (brutto)
dn_duedate	Originales Fälligkeitsdatum der Rechnung
dn_dunning_date	Datum der Mahnung
dn_dunning_duedate	Fälligkeitsdatum der Mahnung
dn_fee	Kummulative Mahngebühr
dn_interest	Zinsen per anno für diese Rechnung
dn_invnumber	Rechnungsnummer
dn_linetotal	Noch zu zahlender Betrag (ergibt sich aus dn_open_amount + dn_fee + dn_interest)
dn_netamount	Rechnungssumme (netto)
dn_open_amount	Offener Rechnungsbetrag
dn_ordnumber	Bestellnummer
dn_transdate	Rechnungsdatum
dn_curr	Währung, in der die Rechnung erstellt wurde. (Die Rechnungsbeträge sind aber immer in der Hauptwährung)

3.3.9.4. Variablen in automatisch erzeugten Rechnungen über Mahngebühren

Die Variablen des Verkäufers stehen wie gewohnt als employee_... zur Verfügung. Die Adresdaten des Kunden stehen als Variablen name, street, zipcode, city, country, department_1, department_2, und email zur Verfügung.

Weitere Variablen beinhalten:

duedate	Fälligkeitsdatum der Rechnung
dunning_id	Mahnungsnummer
fee	Mahngebühren
interest	Zinsen
invamount	Rechnungssumme (ergibt sich aus fee + interest)
invdate	Rechnungsdatum
invnumber	Rechnungsnummer

3.3.10. Variablen in anderen Vorlagen

3.3.10.1. Einführung

Die Variablen in anderen Vorlagen sind ähnlich wie in der Rechnung. Allerdings heißen die Variablen, die

mit inv beginnen, jetzt anders. Bei den Angeboten fangen sie mit quo für "quotation" an: quodate für Angebotsdatum etc. Bei Bestellungen wiederum fangen sie mit ord für "order" an: ordnumber für Bestellnummer etc.

Manche Variablen sind in anderen Vorlagen hingegen gar nicht vorhanden wie z.B. die für bereits verbuchte Zahlungseingänge. Dies sind Variablen, die vom Geschäftsablauf her in der entsprechenden Vorlage keine Bedeutung haben oder noch nicht belegt sein können.

Im Folgenden werden nur wichtige Unterschiede zu den Variablen in Rechnungen aufgeführt.

3.3.10.2. Angebote und Preisfragen

quonumber	Angebots- bzw. Anfragenummer
reqdate	Gültigkeitsdatum (bei Angeboten) bzw. Lieferdatum (bei Preisfragen)
transdate	Angebots- bzw. Anfragedatum

3.3.10.3. Auftragsbestätigungen und Lieferantenaufträge

ordnumber	Auftragsnummer
reqdate	Lieferdatum
transdate	Auftragsdatum

3.3.10.4. Lieferscheine (Verkauf und Einkauf)

cusordnumber	Bestellnummer des Kunden (im Verkauf) bzw. Bestellnummer des Lieferanten (im Einkauf)
donumber	Lieferscheinnummer
transdate	Lieferscheindatum

Für jede Position eines Lieferscheines gibt es ein Unterarray mit den Informationen darüber, von welchem Lager und Lagerplatz aus die Waren verschickt wurden (Verkaufslieferscheine) bzw. auf welchen Lagerplatz sie eingelagert wurden. Diese müssen mittels einer foreach-Schleife ausgegeben werden. Diese Variablen sind:

si_bin	Lagerplatz
si_chargenumber	Chargennummer
si_bestbefore	Mindesthaltbarkeit
si_number	Artikelnummer
si_qty	Anzahl bzw. Menge
si_runningnumber	Positionsnummer (1, 2, 3 etc)
si_unit	Einheit
si_warehouse	Lager

--	--

3.3.10.5. Variablen für Sammelrechnung

c0total	Gesamtbetrag aller Rechnungen mit Fälligkeit < 30 Tage
c30total	Gesamtbetrag aller Rechnungen mit Fälligkeit >= 30 und < 60 Tage
c60total	Gesamtbetrag aller Rechnungen mit Fälligkeit >= 60 und < 90 Tage
c90total	Gesamtbetrag aller Rechnungen mit Fälligkeit >= 90 Tage
total	Gesamtbetrag aller Rechnungen

Variablen für jede Rechnungsposition in Sammelrechnung:

invnumber	Rechnungsnummer
invdate	Rechnungsdatum
duedate	Fälligkeitsdatum
amount	Summe der Rechnung
open	Noch offener Betrag der Rechnung
c0	Noch offener Rechnungsbetrag mit Fälligkeit < 30 Tage
c30	Noch offener Rechnungsbetrag mit Fälligkeit >= 30 und < 60 Tage
c60	Noch offener Rechnungsbetrag mit Fälligkeit >= 60 und < 90 Tage
c90	Noch offener Rechnungsbetrag mit Fälligkeit >= 90 Tage

3.3.11. Blöcke, bedingte Anweisungen und Schleifen

3.3.11.1. Einführung

Der Parser kennt neben den Variablen einige weitere Konstrukte, die gesondert behandelt werden. Diese sind wie Variablennamen in spezieller Weise markiert: `<%anweisung%> ... <%end%>`

Anmerkung zum `<%end%>`: Der besseren Verständlichkeit halber kann man nach dem end noch beliebig weitere Wörter schreiben, um so zu markieren, welche Anweisung (z.B. if oder foreach) damit abgeschlossen wird.

Beispiel: Lautet der Beginn eines Blockes z.B. `<%if type == "sales_quotation"%>`, so könnte er mit `<%end %>` genauso abgeschlossen werden wie mit `<%end if%>` oder auch `<%end type == "sales_quotation"%>`.

3.3.11.2. Der if-Block

```
<%if variablenname%>
...
<%end%>
```

Eine normale "if-then"-Bedingung. Die Zeilen zwischen dem "if" und dem "end" werden nur ausgegeben,

wenn die Variable variablenname gesetzt und ungleich 0 ist.

Handelt es sich bei der benannten Variable um ein Array, also um einen Variablennamen, über den man mit `<%foreach variablenname%>` iteriert, so wird mit diesem Konstrukt darauf getestet, ob das Array Elemente enthält. Somit würde im folgenden Beispiel nur dann eine Liste von Zahlungseingängen samt ihrer Überschrift "Zahlungseingänge" ausgegeben, wenn tatsächlich welche getätigt wurden:

```
<%if payment%>
Zahlungseingänge:
  <%foreach payment%>
    Am <%paymentdate%>: <%payment%> €
  <%end foreach%>
<%end if%>
```

Die Bedingung kann auch negiert werden, indem das Wort `not` nach dem `if` verwendet wird. Beispiel:

```
<%if not cp_greeting%>
...
<%end%>
```

Zusätzlich zu dem einfachen Test, ob eine Variable gesetzt ist oder nicht, bietet dieser Block auch die Möglichkeit, den Inhalt einer Variablen mit einer festen Zeichenkette oder einer anderen Variablen zu vergleichen. Ob der Vergleich mit einer Zeichenkette oder einer anderen Variablen vorgenommen wird, hängt davon ab, ob die rechte Seite des Vergleichsoperators in Anführungszeichen gesetzt wird (Vergleich mit Zeichenkette) oder nicht (Vergleich mit anderer Variablen). Zwei Beispiele, die beide Vergleiche zeigen:

```
<%if var1 == "Wert"%>
```

Testet die Variable `var1` auf Übereinstimmung mit der Zeichenkette `Wert`. Mittels `!=` anstelle von `==` würde auf Ungleichheit getestet.

```
<%if var1 == var2%>
```

Testet die Variable `var1` auf Übereinstimmung mit der Variablen `var2`. Mittels `!=` anstelle von `==` würde auf Ungleichheit getestet.

Erfahrene Benutzer können neben der Tests auf (Un-)Gleichheit auch Tests auf Übereinstimmung mit regulären Ausdrücken ohne Berücksichtigung der Groß- und Kleinschreibung durchführen. Dazu dient dieselbe Syntax wie oben nur mit `=~` und `!~` als Vergleichsoperatoren.

Beispiel für einen Test, ob die Variable `intnotes` (interne Bemerkungen) das Wort `schwierig` enthält:

```
<%if intnotes =~ "schwierig"%>
```

3.3.11.3. Der `foreach`-Block

```
<%foreach variablenname%>
...
<%end%>
```

Fügt die Zeilen zwischen den beiden Anweisungen so oft ein, wie das Perl-Array der Variablen `variablenname` Elemente enthält. Dieses Konstrukt wird zur Ausgabe der einzelnen Posten einer Rechnung / eines Angebots sowie zur Ausgabe der Steuern benutzt. In jedem Durchlauf werden die [zeilenbezogenen Variablen](#) jeweils auf den Wert für die aktuelle Position gesetzt.

Die Syntax sieht normalerweise wie folgt aus:

```
<%foreach number%>
Position: <%runningnumber%>
Anzahl: <%qty%>
Artikelnummer: <%number%>
Beschreibung: <%description%>
. . .
<%end%>
```

Besonderheit in OpenDocument-Vorlagen: Tritt ein `<%foreach%>`-Block innerhalb einer Tabellenzelle auf, so wird die komplette Tabellenzeile so oft wiederholt wie notwendig. Tritt er außerhalb auf, so wird nur der Inhalt zwischen `<%foreach%>` und `<%end%>` wiederholt, nicht aber die komplette Zeile, in der er steht.

3.3.12. Markup-Code zur Textformatierung innerhalb von Formularen

Wenn der Benutzer innerhalb von Formularen in kivitendo Text anders formatiert haben möchte, so ist dies begrenzt möglich. kivitendo unterstützt die Textformatierung mit HTML-ähnlichen Tags. Der Benutzer kann z.B. bei der Artikelbeschreibung auf einer Rechnung Teile des Texts zwischen Start- und Endtags setzen. Dieser Teil wird dann automatisch in Anweisungen für das ausgewählte Vorlagenformat (HTML oder PDF über LaTeX) umgesetzt.

Die unterstützten Formatierungen sind:

<code>Text</code>	Text wird in Fettdruck gesetzt.
<code><i>Text</i></code>	Text wird kursiv gesetzt.
<code><u>Text</u></code>	Text wird unterstrichen.
<code><s>Text</s></code>	Text wird durchgestrichen. Diese Formatierung ist nicht bei der Ausgabe als PDF über LaTeX verfügbar.
<code><bullet></code>	Erzeugt einen ausgefüllten Kreis für Aufzählungen (siehe unten).

Der Befehl `<bullet>` funktioniert momentan auch nur in Latex-Vorlagen.

3.4. Excel-Vorlagen

3.4.1. Zusammenfassung

Dieses Dokument beschreibt den Mechanismus, mit dem Exceltemplates abgearbeitet werden, und die Einschränkungen, die damit einhergehen.

3.4.2. Bedienung

Der Excel Mechanismus muss in der Konfigurationsdatei aktiviert werden. Die Konfigurationsoption heißt `excel_templates = 1` im Abschnitt `[print_templates]`.

Eine Excelvorlage kann dann unter dem Namen einer beliebigen anderen Vorlage mit der Endung `.xls` gespeichert werden. In den normalen Verkaufsmasken taucht nun Excel als auswählbares Format auf und kann von da an wie LaTeX- oder OpenOffice-Vorlagen benutzt werden.

