Saldierung von Fakturavorgängen

Daniel Ellermann

17. April 2013

Inhaltsverzeichnis

1 Betrifft

Dieses Dokument beschreibt die Berechnung von Salden von Rechnungen, Gutschriften und Mahnungen in SpringCRM. Es liefert wichtige Formeln für die Berechnung und eine Beschreibung der Herleitung dieser Formeln.

2 Definitionen

Zunächst definieren wir folgende Werte:

Definition 1 (Rechnungsbeträge)**.** *Die Summe einer Rechnung (einschließlich MwSt.) bezeichnen wir mit* σ_r*.*

Den Zahlungsbetrag einer Rechnung, d. h. den durch den Kunden bereits bezahlten Betrag, bezeichnen wir mit b_r*.*

Definition 2 (Gutschriftsbeträge)**.** *Die Summe einer Gutschrift (einschließlich MwSt.) bezeichnen wir mit* σ_g*.*

Den Zahlungsbetrag einer Gutschrift, d. h. den durch uns bereits bezahlten Betrag, bezeichnen wir mit b_g*.*

Im folgenden gelten alle Definitionen und Berechnungen von Mahnungen wie für Rechnungen, da eine Mahnung eine Art anschließende Rechnung darstellt.

3 Salden und Endsaldo

3.1 Salden

Zu jeder Rechnung und jeder Gutschrift kann ein Saldo errechnet werden. Wir berechnen den Saldo so, dass ein positiver Saldo (Werte größer als null) ein Guthaben für uns (bzw. eine Verbindungkeit unserseits) und ein negativer Saldo (Werte kleiner als null) eine Forderung an den Kunden darstellt. Ein Saldo von null stellt eine ausgeglichene Forderung dar.
Für Rechnungen ergibt sich somit der Saldo s_r wie folgt:

$$s_r := b_r - \sigma_r \tag{3.1}$$

Damit ergibt sich, dass eine nicht oder teilweise bezahlte Rechnung einen negativen Saldo aufweist, also $s_r < 0$. Hat der Kunde die Rechnung vollständig bezahlt, gilt $s_r = 0$.

Der Saldo einer Gutschrift s_g wird genau umgekehrt berechnet:

$$s_g := \sigma_g - b_g \tag{3.2}$$

Hier ergibt sich bei einer nicht oder teilweise bezahlten Gutschrift ein positiver Saldo, also $s_g > 0$. Wurde die Gutschrift vollständig bezahlt, gilt $s_g = 0$.

3.2 Endsaldo

Für eine abschließende Betrachtung der Salden müssen jedoch die Salden der Rechnung den Salden von eventuellen Gutschriften gegenübergestellt werden. Dies funktioniert wie die Darstellung in einer Bilanz. Wir berechnen das im Endsaldo s, der für Rechnungen und Gutschriften gleich ist:

$$s := s_r + \sum s_g \tag{3.3}$$

Es gelten folgende Zusammenhänge:

- Positiver Endsaldo $s > 0$. Der Kunde hat ein Guthaben bei uns, d. h. wir schulden ihm Geld.
- Negativer Endsaldo $s < 0$. Der Kunde schuldet uns Geld, d. h. wir haben Forderungen an ihn.
- Endsaldo $s = 0$. Die Rechnung und event. Gutschriften sind ausgeglichen.

3.3 Beispiele

Ein Kunde kauft Artikel über 1.000 EUR. Dann gibt es folgende Beispielfälle:

- Er hat die Rechnung nicht bezahlt. Dann gilt $s_r = 0 - 1000 = -1000$ und $s = -1000$.
- Er bezahlt 800 EUR. Dann gilt $s_r = 800 - 1000 = -200$ und $s = -200$.
- Er bezahlt die komplette Rechnung, d. h. 1.000 EUR. Dann gilt $s_r = 1000 - 1000 = 0$ und $s = 0$.

Zu dieser Rechnung gibt es zwei Gutschriften über 100 EUR und 200 EUR. Dann gibt es folgende Beispielfälle:

- Er hat die Rechnung nicht bezahlt und die Gutschriften wurden auch nicht bezahlt. Dann gilt $s_r = 0 - 1000 = -1000$, $s_{g1} = 100 - 0 = 100$, $s_{g2} = 200 - 0 = 200$ sowie $s = -1000 + 100 + 200 = -700$. Der Kunde muss also noch 700 EUR bezahlen.

- Er bezahlt 400 EUR, die Gutschriften wurden nicht bezahlt. Dann gilt $s_r = 400 - 1000 = -600$, $s_{g1} = 100 - 0 = 100$, $s_{g2} = 200 - 0 = 200$ sowie $s = -600 + 100 + 200 = -300$. Der Kunde muss also noch 300 EUR bezahlen.

- Er bezahlt 700 EUR, die Gutschriften wurden nicht bezahlt. Dann gilt $s_r = 700 - 1000 = -300$, $s_{g1} = 100 - 0 = 100$, $s_{g2} = 200 - 0 = 200$ sowie $s = -300 + 100 + 200 = 0$. Die Rechnung wurde damit vollständig bezahlt.

- Er bezahlt 900 EUR, die Gutschriften wurden nicht bezahlt. Dann gilt $s_r = 900 - 1000 = -100$, $s_{g1} = 100 - 0 = 100$, $s_{g2} = 200 - 0 = 200$ sowie $s = -100 + 100 + 200 = 200$. Wir müssen noch 200 EUR an den Kunden auszahlen.

- Er bezahlt 900 EUR, die Gutschrift über 100 EUR wurde bezahlt, die andere Gutschrift nicht. Dann gilt $s_r = 900 - 1000 = -100$, $s_{g1} = 100 - 100 = 0$, $s_{g2} = 200 - 0 = 200$ sowie $s = -100 + 0 + 200 = 100$. Wir müssen noch 100 EUR an den Kunden auszahlen.

- Er bezahlt 1.000 EUR, die Gutschriften wurden bezahlt. Dann gilt $s_r = 1000 - 1000 = 0$, $s_{g1} = 100 - 100 = 0$, $s_{g2} = 200 - 200 = 0$ sowie $s = 0 + 0 + 0 = 0$. Die Rechnung und die Gutschriften wurde damit vollständig bezahlt.

4 Modifizierter Endsaldo

Die Formulare für Rechnungen und Gutschriften sollen den noch zu zahlenden Betrag dynamisch anzeigen können, d. h. bei Änderungen der Rechnungs- bzw. Gutschriftenposten oder des Zahlungsbetrages soll sich eine Angabe „noch zu zahlen“ dynamisch neu berechnen. Dies wird mit JavaScript erledigt.

Um die dynamische Berechnung zu vereinheitlichen muss der JavaScript-Funktion ein normierter Endsaldo mitgegeben werden, der sog. „modifizierte Endsaldo“. Der modifizierte Endsaldo muss zwei Funktionen erfüllen:

- Der Saldo der Rechnung bzw. Gutschrift muss vom Endsaldo subtrahiert werden damit die JavaScript-Funktion den Rechnungs- bzw. Gutschriftensaldo dynamisch hinzuaddieren kann.

- Das Vorzeichen des modifizierten Endsaldos muss geeignet gesetzt sein. Bei Rechnungen ist es positiv, damit vom Kunden zu zahlende Beträge positiv erscheinen. Bei Gutschriften ist es dagegen negativ, damit von uns zu zahlende Beträge positiv erscheinen.

Unter diesen Gesichtspunkten definieren wir den modifizierten Endsaldo für Rechnungen m_r wie folgt:

$$m_r := s - s_r \tag{4.1}$$

Den modifizierten Endsaldo für Gutschriften m_g definieren wir dagegen wie folgt:

$$m_g := -(s - s_g) = s_g - s \tag{4.2}$$

Durch das negative Vorzeichen erfüllen wir das zweite oben genannte Kriterium. In beiden Fällen wird vom Endsaldo s der Rechnungs- bzw. Gutschriftssaldo abgezogen (erstes Kriterium), damit ihn die JavaScript-Funktion für den „noch zu zahlenden Betrag“ (s. Abschnitt 5 auf Seite 4) dynamisch aufaddieren kann.

5 Noch zu zahlender Betrag

Wie oben beschrieben, wurde im modifizierten Endsaldo der Rechnungs- bzw. Gutschriftssaldo vom Endsaldo abgezogen. Die JavaScript-Funktion kann nun den Wert „noch zu zahlen“ basierend auf dem modifizierten Endsaldo und aktuellen Werten für Rechnungssumme und Zahlungsbetrag berechnen. Die aktuelle Rechnungssumme bezeichnen wir mit σ'_r und den aktuellen Zahlungsbetrag mit b'_r. Die Differenz zur Rechnungssumme σ_r bezeichnen wir mit $\Delta\sigma_r$ mit

$$\Delta\sigma_r := \sigma'_r - \sigma_r \tag{5.1}$$

Weiterhin bezeichnen wir die Differenz zwischen aktuellem Zahlungsbetrag b'_r und Zahlungsbetrag b_r mit Δb_r mit

$$\Delta b_r := b'_r - b_r \tag{5.2}$$

Beim Betrag „noch zu zahlen“ muss zwei Kriterien erfüllen:

- Das Vorzeichen muss umgekehrt werden, damit negative Salden, die ja eine Forderung an den Kunden bedeuten, als positive „Noch zu zahlen“-Werte erscheinen.
- Der Wert ergibt sich durch Addition des aktuellen Rechnungssaldos s'_r mit $s'_r := b'_r - \sigma'_r$ bzw. Subtraktion des aktuellen Gutschriftssaldos s'_g mit $s'_g := \sigma'_g - b'_g$ zum bzw. vom modifizierten Endsaldo m_r bzw. m_g. Durch Addition bzw. Subtraktion der *aktuellen* Werte zeigt der „noch zu zahlende“ Betrag immer den aktuellen offenen Zahlbetrag an.

Für Rechnungen ergibt sich die Angabe „noch zu zahlen“ z_r unter Beachtung der genannten Kriterien wie folgt:

$$\begin{aligned}
z_r :=& -(m_r + s'_r) \\
=& -(s - s_r + s'_r) \\
=& -(s - b_r + \sigma_r + b'_r - \sigma'_r) \\
=& -(s + b'_r - b_r - \sigma'_r + \sigma_r) \\
=& -(s + \Delta b_r - \Delta\sigma_r) \\
=& \Delta\sigma_r - \Delta b_r - s
\end{aligned} \tag{5.3}$$

Im speziellen Fall $\sigma'_r = \sigma_r$ und $b'_r = b_r$, also $\Delta\sigma_r = \Delta b_r = 0$ gilt:

$$z_r = -s \tag{5.4}$$

Für Gutschriften ergibt sich die Berechnung analog. Hier bezeichnen wir die aktuelle Gutschriftssumme mit σ'_g und den aktuellen Zahlungsbetrag mit b'_g. Die Differenz zwischen aktueller Gutschriftssumme σ'_g und Gutschriftssumme σ_g bezeichnen wir mit $\Delta\sigma_g$ mit

$$\Delta\sigma_g := \sigma'_g - \sigma_g \tag{5.5}$$

Weiterhin bezeichnen wir die Differenz zwischen aktuellem Zahlungsbetrag b'_g und Zahlungsbetrag b_g mit Δb_g mit

$$\Delta b_g := b'_g - b_g \tag{5.6}$$

Dann ergibt sich die Angabe „noch zu zahlen" z_g für Gutschriften wie folgt:

$$\begin{aligned} z_g &:= -\left(m_g - s'_g\right) \\ &= -\left(-(s - s_g) - s'_g\right) \\ &= -\left(s_g - s - s'_g\right) \\ &= -\left(\sigma_g - b_g - s - \sigma'_g + b'_g\right) \\ &= -\left(b'_g - b_g - \sigma'_g + \sigma_g - s\right) \\ &= -\left(\Delta b_g - \Delta\sigma_g - s\right) \\ &= \Delta\sigma_g - \Delta b_g + s \end{aligned} \tag{5.7}$$

Auch hier gilt im speziellen Fall $\sigma'_g = \sigma_g$ und $b'_g = b_g$, also $\Delta\sigma_g = \Delta b_g = 0$:

$$z_g = s \tag{5.8}$$

6 Saldenausgleich

Der Benutzer soll durch Klick auf „noch zu zahlen" einen Saldenausgleich herbeiführen können. Dadurch soll der Zahlungsbetrag einer Rechnung bzw. Gutschrift berechnet werden, dass sich ein ausgeglichener Endsaldo $s = 0$ ergibt. Der Zahlungsbetrag c_r, der bei Klick auf „noch zu zahlen" bei einer Rechnung einzutragen ist, ergibt sich wie folgt:

$$\begin{aligned} c_r &:= \sigma_r - m_r \\ &= \sigma_r - s + s_r \\ &= \sigma_r - s + b_r - \sigma_r \\ &= b_r - s \\ &= b_r - s_r - \sum s_g \\ &= b_r - b_r + \sigma_r - \sum s_g \\ &= \sigma_r - \sum s_g \end{aligned} \tag{6.1}$$

Nach Klick auf „noch zu zahlen“ ist der Zahlungsbetrag also so hoch wie die Rechnungssumme abzüglich eventueller Gutschriftssalden. Dadurch ergibt sich ein ausgeglichener Endsaldo, wenn wir c_r für b_r einsetzen:

$$\begin{aligned} s &= s_r + \sum s_g \\ &= c_r - \sigma_r + \sum s_g \\ &= \sigma_r - \sum s_g - \sigma_r + \sum s_g \\ &= 0 \end{aligned} \tag{6.2}$$

Bei Gutschriften erfolgt die Berechnung analog. Der Zahlungsbetrag c_{g_i}, der bei Klick auf „noch zu zahlen“ bei einer Gutschrift Nr. i einzutragen ist, ergibt sich wie folgt:

$$\begin{aligned} c_{g_i} &:= \sigma_{g_i} - m_{g_i} \\ &= \sigma_{g_i} - s_{g_i} + s \\ &= \sigma_{g_i} - \sigma_{g_i} + b_{g_i} + s \\ &= b_{g_i} + s \\ &= b_{g_i} + s_r + \sum s_g \\ &= b_{g_i} + s_r + \sum \sigma_g - \sum b_g \\ &= b_{g_i} + s_r + \sum \sigma_g - b_{g_1} - \ldots - b_{g_n} \\ &= s_r + \sum \sigma_g - b_{g_1} - \ldots - b_{g_{i-1}} - b_{g_{i+1}} - \ldots - b_{g_n} \end{aligned} \tag{6.3}$$

Setzt man den berechneten Zahlungsbetrag c_{g_i} für den Zahlungsbetrag b_{g_i} der Gutschrift Nr. i ein, ergibt sich ein ausgeglichener Endsaldo wie folgt:

$$\begin{aligned} s &= s_r + \sum s_g \\ &= s_r + \sum \sigma_g - \sum b_g \\ &= s_r + \sum \sigma_g - b_{g_1} - \ldots - b_{g_{i-1}} - \\ &\quad (s_r + \sum \sigma_g - b_{g_1} - \ldots - b_{g_{i-1}} - b_{g_{i+1}} - \ldots - b_{g_n}) - \\ &\quad b_{g_{i+1}} - \ldots - b_{g_n} \\ &= s_r + \sum \sigma_g - b_{g_1} - \ldots - b_{g_{i-1}} - \\ &\quad s_r - \sum \sigma_g + b_{g_1} + \ldots + b_{g_{i-1}} + b_{g_{i+1}} + \ldots + b_{g_n} - \\ &\quad b_{g_{i+1}} - \ldots - b_{g_n} \\ &= 0 \end{aligned} \tag{6.4}$$