

Lohn-Berechnung

Die Lohn-Berechnung setzt sich grundsätzlich zusammen aus

- Lohndaten vorbereiten
- Lohndaten berechnen

Verallgemeinert ausgedrückt, liegt der Unterschied bei Lohndaten *vorbereiten* und *berechnen* darin, dass es für jede Abrechnungsperiode Daten gibt, die sich nicht oder nur selten ändern; diese können *vorbereitet*, d.h. gecached werden. Alle übrigen Daten müssen in Echtzeit ermittelt (*berechnet*) werden.

Lohndaten vorbereiten

Vorbereite Lohndaten können aus manuell erfassten Daten und vom System ermittelten Daten bestehen.

Dazu gehören z.Bsp.:

- Manuell erfasster Monatslohn im Personalstamm
- Manuell erfasste Anzahl Stunden bei Stundenlöhnern
- Manuell erfasste Sonderzulagen, etc.
- Feldmodifikatoren, die zwar manuell erfasst, aber für jede Abrechnungsperiode neu durch das System ausgewertet und *vorbereitet* werden
- Versicherungsprämien (Sozialversicherungen), die pro Abrechnungsperiode und Person neu ermittelt werden: z.Bsp. muss das System automatisch reagieren, wenn ein Mitarbeiter das AHV-Eintrittsalter erreicht. Ab diesem Zeitpunkt müssen bei der betreffenden Person die AHV-, ALV-Beitragsprämien *vorbereitet* werden.
- Renteneintritt
- etc...

Alle diese vorbereiteten Daten werden durch das Stored Procedure (SP) **payroll_prc_empl_acc** ermittelt und zentral in der Tabelle **payroll_employee_account** abgelegt.

Das SP wird im Prinzip nur in zwei Situationen aufgerufen:

1. IMMER: Nachdem eine Abrechnungsperiode abgeschlossen wird und eine neue Abrechnungsperiode angelegt wurde
2. NACH BEDARF: Wenn sich durch einen Usereingriff ein Parameter geändert hat, der auf die vorbereiteten Daten einen Einfluss haben kann. Der Code in *code_logic/changeManager.php* entscheidet, ob nach einer Änderung das SP aufzurufen ist.
Beispiel: User ändert im Personalstamm den Monatslohn oder einen Versicherungscode.

Im ersten Fall wird das SP **payroll_prc_empl_acc** auf allen aktiven Mitarbeitern angewendet. Hingegen wird das SP im zweiten Fall nur bei denjenigen Mitarbeitern laufen gelassen, wo eine Änderung vorgenommen wurde.

Auf welche Mitarbeiter das SP **payroll_prc_empl_acc** angewendet wird, lässt sich bestimmen, indem die Record-IDs der betreffenden Mitarbeiter (payroll_employee.id) in der Tabelle **payroll_tmp_change_mng** eingetragen werden. Hier ein SQL-Beispiel:

DELETE FROM payroll_tmp_change_mng WHERE core_user_id=9;	Sicherstellen, dass alte Mitarbeiter-IDs gelöscht werden. core_user_id muss der RecID des aktuell angemeldeten User entsprechen, weil die Einträge der Session, resp. dem User zugeordnet sind.
INSERT INTO payroll_tmp_change_mng (core_user_id, numID) VALUES(9,456),(9,789);	Record-IDs von zwei Mitarbeitern (456 u. 789) eintragen, bei denen die Lohndaten vorbereitet werden sollen.
Call payroll_prc_empl_acc (9, 1, 1, 1, 1, 1, 1, 1) ;	SP aufrufen. arg1 = core_user_ID arg2 = START TRANSACTION / COMMIT (1=SP soll sich darum kümmern, 0=Entwickler kümmert sich selber um Transaktionsstart/-ende) Mit den übrigen arg's lassen sich Teilprozesse innerhalb des SP ein-/ausschalten. Diese arg's akzeptieren die Werte 0 oder 1. Wenn wie in diesem Beispiel überall 1 steht, werden alle Teilprozesse ausgeführt. Details dazu können dem Quelltext des SP entnommen werden.

Lohndaten berechnen

Das Grundprinzip der Lohnartensteuerung ist relativ simpel: Jede Lohnart kann quasi als „Konto“ mit mehreren Zahlenwerten (z.Bsp. Menge, Anzahl, Betrag) verstanden werden. Innerhalb einer Lohnart können die einzelnen Zahlenwerte miteinander verrechnet werden und das Resultat kann an Nachfolge-Lohnarten weitergereicht werden.

Ein Beispiel:

Für den aktuellen Monat, erfassen wir bei der Mitarbeiterin „Aebi, Anna“ unter der Lohnart 1310 („Anzahl gearbeitete Stunden“) den Wert 100.

Gültigk.	LOM Nr.	LOM Text	Menge	Ansatz	Betrag	Kostestelle
∞	1055	Wegentschädigung	1.00000	200.00000	0.00000	
∞	1976	Vom Arbeitgeber übernom. Säule 3b	150.00000	0.00000	0.00000	
∞	1977	Vom Arbeitgeber übernom. Säule 3a	350.00000	0.00000	0.00000	
∞	6002	Verpflegungsspesen	320.00000	0.00000	0.00000	
∞	6030	Übrige effektive Spesen	155.00000	0.00000	0.00000	
∞	6040	Pauschale Repräsentationsspesen	300.00000	0.00000	0.00000	

Dieser manuell erfasste Wert wird in der Tabelle **payroll_employee_account** gespeichert, wo auch alle übrigen vorbereiteten Daten abgelegt sind.

Weil bei „Aebi, Anna“ im Personalstamm der Lohn-Code „1 = Stundenlohn“ und „Basislohn = 59.30“ erfasst wurde, wird bei *Lohndaten vorbereiten* auch automatisch die Lohnart 1005 („Stundenlohn“) mit dem Stundenansatz 59.30 in der Tabelle **payroll_employee_account** gespeichert.

Tabelle **payroll_employee_account** (Auszug):

payroll_account_ID	PayrollDataType	account_text	quantity	rate	amount	TargetField	max_limit	min_limit	deduction	CostCenter
1005	2		0.00000	59.30000	0.00000	0	0.00	0.00	0.00	
1310	1		100.00000	0.00000	0.00000	0	0.00	0.00	0.00	

Um die Berechnung zu starten, müssen – wie bei *Lohndaten vorbereiten* – zuerst die Mitarbeiter definiert werden, bei denen die Lohnberechnung durchgeführt werden soll:

```
DELETE FROM payroll_tmp_change_mng WHERE core_user_id=9;  
INSERT INTO payroll_tmp_change_mng(core_user_id, numID) VALUES(9,456),(9,789);
```

Anschließend ist die Funktion *calculate* in der Datei *code_logic/payroll_calculation.php* aufzurufen. Dieser Funktion kann das optionale Argument *calculateAll* übergeben werden. Wenn *calculateAll=true* (default!), dann werden alle Mitarbeiter berechnet (ungeachtet der zuvor festgelegten Mitarbeitern), wenn *calculateAll=false* werden nur die zuvor festgelegten Mitarbeiter mit den IDs 456 und 789 berechnet.

Innerhalb der Funktion *calculate* wird das SP **payroll_prc_calculate** aufgerufen, wo der Hauptteil der Verarbeitungslogik rund um die Berechnung untergebracht ist.

In **payroll_prc_calculate** passiert folgendes:

Die für die Berechnung benötigten Lohnarten (LOA) werden in die temporäre Tabelle **payroll_tmp_calculation** geladen. Welche LOA das sind, wird pro Mitarbeiter anhand der vorbereiteten Daten in **payroll_employee_account**, den Quellensteuercodes, den LOA-Einstellungen im Lohnartenstamm (**payroll_account**) und der Systemlohnarten (**payroll_account_mapping**) entschieden. Beispielsweise werden LOA mit dem Wert **payroll_account.mandatory=1** immer eingefügt.

Im Folgenden ist zu sehen, wie sich die Werte in der Tabelle **payroll_tmp_calculation** schrittweise verändern...

Schritt 1: LOA-Basisdaten werden aus der LOA-Konfiguration **payroll_account** geladen. Nebst vielen anderen Feldern, werden die Felder quantity (MENGE), rate (ANSATZ) und amount (BETRAG) mit den Default-Werten gefüllt (hier überall 0.00; könnte aber je nach LOA-Konfiguration auch ein Wert ungleich 0.00 sein).

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	0.00	0.00	0.00	-- keine --	
1005	0.00	0.00	0.00	quantity x rate	
9010	0.00	0.00	0.00	-- keine --	

The screenshot shows the 'Lohnart bearbeiten (1005 - Stundenlohn)' window. The 'Berechnung' tab is selected. The table below shows the calculation results:

FAKTOR	ZUSCHLAG	MENGE	ANSATZ	BETRAG
0.00000	0.00000	0.00000	0.00000	0.00000

The 'Formel' field contains 'MENGE x ANSATZ'. The 'Zu ueberschreibendes Feld' is set to 'BETRAG'.

Schritt 2: Die Daten aus den vorbereiteten Lohndaten werden geladen.

payroll_account_ID	PayrollDataType	account_text	quantity	rate	amount	TargetField	max_limit	min_limit	deduction	CostCenter
1005	2		0.00000	59.30000	0.00000	0	0.00	0.00	0.00	
1310	1		100.00000	0.00000	0.00000	0	0.00	0.00	0.00	

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	100.00	0.00	0.00	-- keine --	
1005	0.00	59.30	0.00	quantity x rate	
9010	0.00	0.00	0.00	-- keine --	

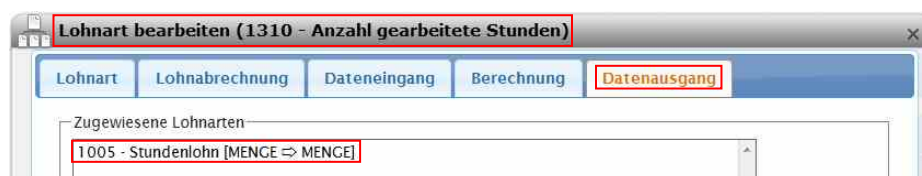
Schritt 3: Die Lohnarten werden der vorgegebenen Reihenfolge nach abgearbeitet (Reihenfolge wird definiert durch **payroll_account.processing_order** und diese wiederum wird nach jeder Änderung an der LOA-Konfiguration neu ermittelt mit dem SP **payroll_prc_group_accounts**).

Teilschritt 3a:

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	100.00	0.00	0.00	-- keine --	keine Verarbeitung: keine Formel, etc.
1005	0.00	59.30	0.00	quantity x rate	
9010	0.00	0.00	0.00	-- keine --	

Teilschritt 3b:

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	100.00	0.00	0.00	-- keine --	gemäss LOA-Verknüpfung muss der Wert von 1310, Feld MENGE in das Feld MENGE der LOA 1005 vorgetragen werden (s. Screenshot unten).
1005	100.00	59.30	0.00	quantity x rate	
9010	0.00	0.00	0.00	-- keine --	



Bemerkung: Die Verknüpfungsinformationen sind in der Tabelle **payroll_account_linker** abgelegt.

Teilschritt 3c:

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	100.00	0.00	0.00	-- keine --	
1005	100.00	59.30	5930.00	quantity x rate	Berechnung ausführen: MENGE x ANSATZ, Resultat auf 0.05 runden und in Feld BETRAG (amount) speichern → siehe Screenshot unten
9010	0.00	0.00	0.00	-- keine --	

Lohnart bearbeiten (1005 - Stundenlohn)

Limiten, Abzug/Reduktion:

Maximaler Betrag: 0.00
 Abzug/Reduktion: 0.00
 Minimaler Betrag: 0.00

Jahresausgleich: Per Ende Monat
 Hilfslohnart:

Verwendung des Zwischenresultats nach Limitenberechnung, Abzug/Reduktion:
 Zu ueberschreibendes Feld: keine Wertzuweisung

FAKTOR	ZUSCHLAG	MENGE	ANSATZ	BETRAG
0.00000	0.00000	0.00000	0.00000	0.00000

☒ Berechnung
 Formel: MENGE x ANSATZ

☒ Rundung
 Rundung: 0.0500

Verwendung des Resultats:
 Zu ueberschreibendes Feld: BETRAG

Teilschritt 3d:

payroll_account_ID	quantity	rate	amount	«Formel»	«Bemerkung»
1310	100.00	0.00	0.00	-- keine --	
1005	100.00	59.30	5930.00	quantity x rate	gemäss LOA-Verknüpfung muss der Wert von 1005, Feld BETRAG in das Feld BETRAG von mehreren Nachfolge-LOA vorgetragen werden (s. Screenshot unten).
9010	0.00	0.00	5930.00	-- keine --	
9030	0.00	0.00	5930.00	-- keine --	
9040	0.00	0.00	5930.00	-- keine --	
etc. ...					

Lohnart bearbeiten (1005 - Stundenlohn)

Zugewiesene Lohnarten:

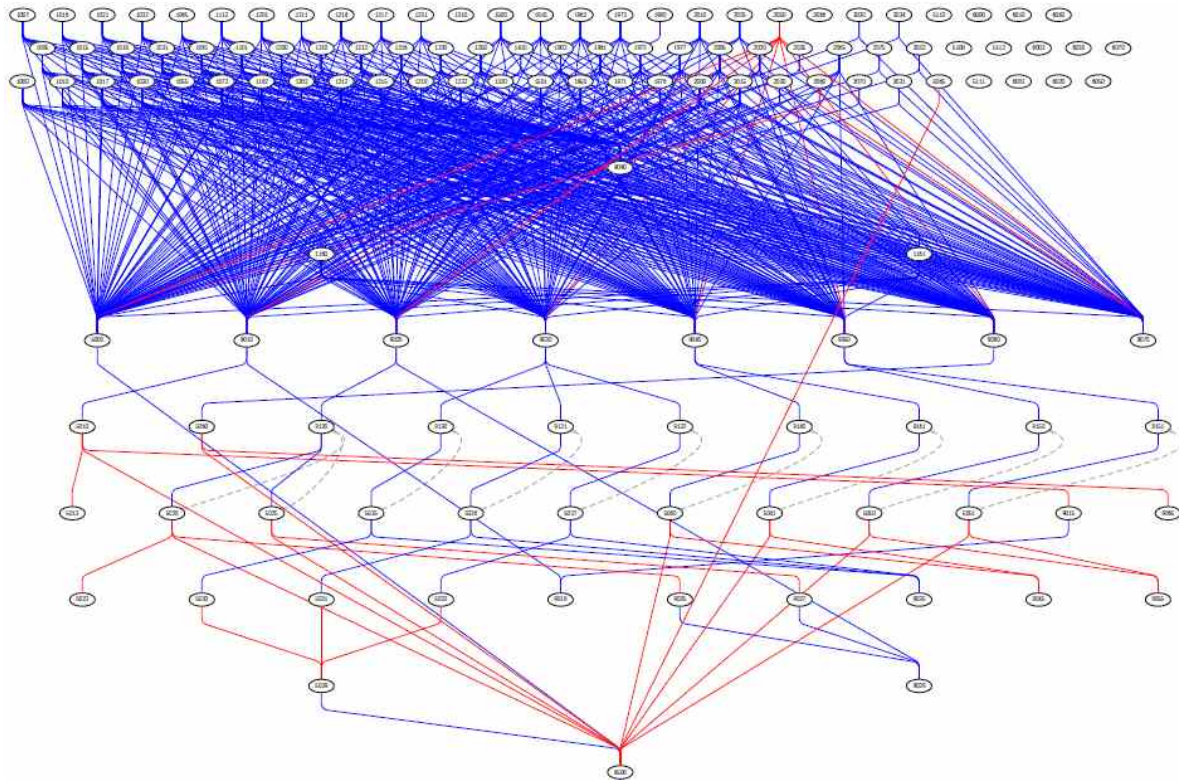
- 9010 - AHV-Basis [BETRAG ⇒ BETRAG]
- 9030 - UVG-Basis [BETRAG ⇒ BETRAG]
- 9040 - UVGZ-Basis [BETRAG ⇒ BETRAG]
- 9050 - KTG-Basis [BETRAG ⇒ BETRAG]
- 9060 - BVG-Basis [BETRAG ⇒ BETRAG]
- 9090 - Ferien-Basis [BETRAG ⇒ BETRAG]
- 9020 - ALV-Basis [BETRAG ⇒ BETRAG]
- 9070 - QST-Basis [BETRAG ⇒ BETRAG]

Hinzufügen Bearbeiten Löschen Öffnen

Wichtig: Beim Vortragen der Resultate auf Nachfolge-LOA wird das Zielfeld nicht durch den vorgetragenen Wert überschrieben, sondern der Wert wird dem Zielfeld addiert! Wäre also im

obigen Beispiel bei der LOA 9010 im Feld *amount* bereits der Wert 1'000.00 hinterlegt gewesen, wäre der *amount*-Wert bei LOA 9010 nach dem Vortragen neu $1'000.00 + 5'930.00 = \underline{6'930.00}$

Wenn alle LOA miteinander verknüpft sind, kann ein einfaches Berechnungsnetzwerk wie folgt aussehen:



Nachdem die Berechnung abgeschlossen ist, werden die Resultate von der Tabelle **payroll_tmp_calculation** in die Tabelle **payroll_calculation_current** verschoben. Die Tabelle **payroll_calculation_current** enthält also immer die neusten Resultate der aktuellen Periode. Diese werden beim Abschluss einer Periode (resp. schon beim Auszahlen) in die Tabelle **payroll_calculation_entry** geschrieben.

Zusammengefasst:

- **payroll_tmp_calculation:** Enthält nur während der paar wenigen Millisekunden/Sekunden (solange die Berechnung dauert) Daten.
- **payroll_calculation_current:** Enthält die Resultate der aktuellen Periode
- **payroll_calculation_entry:** Enthält die Resultate sämtlicher Perioden, ausser der aktuellen Periode

Bei einer Auswertung, welche sowohl die vergangenen als auch die aktuellste Periode berücksichtigen soll, ist daher darauf zu achten, dass die Daten der beiden Tabellen **payroll_calculation_current** und **payroll_calculation_entry** per UNION-Statement vereint werden. Dies ist problemlos möglich. Beispiele sind in der Datei *code_logic/payroll_reports.php* zu finden (nach „UNION“ suchen).