

## **Problem**

The existing payroll processing system in the Payroll Canada System lacks a streamlined mechanism for calculating federal, provincial, and territorial payroll deductions directly from the Salary Tracking Screen. This deficiency necessitates users to manually navigate to the Payroll Deductions Online Calculator (PDOC) on the CRA website for these calculations.

In the absence of an integrated solution, users encounter inefficiencies and a potential for errors in the payroll deduction calculation process. Manually toggling between the Payroll Canada System and the PDOC on the CRA website not only consumes additional time but also introduces the risk of data inconsistencies.

To address this issue, we aim to implement a new feature that facilitates the seamless transmission of data to the PDOC on the CRA website directly from the Salary Tracking Screen. This enhancement will empower users to calculate federal, provincial, and territorial payroll deductions effortlessly within the Payroll Canada System, thereby eliminating the need for manual interventions and enhancing the overall efficiency of the payroll processing workflow.

## **Objective**

The primary objective of this enhancement is to establish a seamless process for transmitting essential payroll data to the Payroll Deductions Online Calculator (PDOC) on the CRA website directly from the Salary Tracking Screen within the Payroll Canada System. Specifically, our aim is to automate the calculation of federal and provincial tax deductions, considering changes triggered by exceeding CPP/EI yearly contributions.

This objective aligns with the following key goals:

1. **Efficiency Improvement:** Streamline the process of calculating federal and provincial tax deductions by integrating the functionality directly into the Salary Tracking Screen. This enhancement aims to eliminate the need for users to manually run calculations on the PDOC on the CRA website, thus enhancing overall workflow efficiency.
2. **Real-Time Deduction Updates:** Enable real-time updates of federal and provincial tax figures based on changes in CPP/EI yearly contributions. By automating this process within the system, users will receive instant and accurate deduction information, ensuring compliance with the latest regulatory requirements.
3. **User Convenience:** Enhance user experience by providing a user-friendly interface within the Payroll Canada System, allowing users to conveniently access and utilize the PDOC functionality without navigating to external websites.
4. **Error Reduction:** Minimize the potential for errors that may arise from manual calculations and data discrepancies between the Payroll Canada System and the PDOC. This objective aims to enhance accuracy and reliability in the payroll deduction process.

By achieving these goals, our objective is to significantly improve the payroll processing experience, making it more efficient, accurate, and user-friendly for all system users.

## Analysis

Our primary goal is to enhance the Payroll Process by implementing a system that notifies users promptly about any changes in deductions, prompting them to update salary information without delay. In the second phase of this enhancement, we aim to streamline the transmission of essential information to the Payroll Deductions Online Calculator (PDOC) on the CRA website and efficiently retrieve updated deduction figures for the salary tracking screen.

This change is motivated by the need for a more efficient process. Upon analyzing the current setup, we recognized an opportunity to optimize how data is transmitted to the CRA and how relevant information is retrieved. The objective is to create a smoother, more user-friendly experience that ensures timely updates and compliance with deduction changes.

The proposed modification, aimed at notifying users of any changes in deductions, demands a thorough analysis of the payroll screen, the central hub where users generate payroll for their members. This analysis will delve into the intricacies of the current setup, examining how it processes and displays deduction information.

The screenshot displays a payroll software interface for 'Test Company # 1' on October 10, 2023. The main section is titled 'Payrolls' and shows a summary for the period ending March 2023. The summary includes the following data:

Category	Value	Category	Value
Total Regular Pay:	20,000.01	Total Tax Federal:	2,598.58
Total Vacation Pay:	0.00	Total Tax Provincial:	1,304.40
Total Cpp Employer:	1,137.94	Total Cpp Employee:	1,137.94
Total EI Employer:	361.32	Total EI Employee:	250.00
Total Net Pay:	14,701.01	Total Employee Borne:	0.00

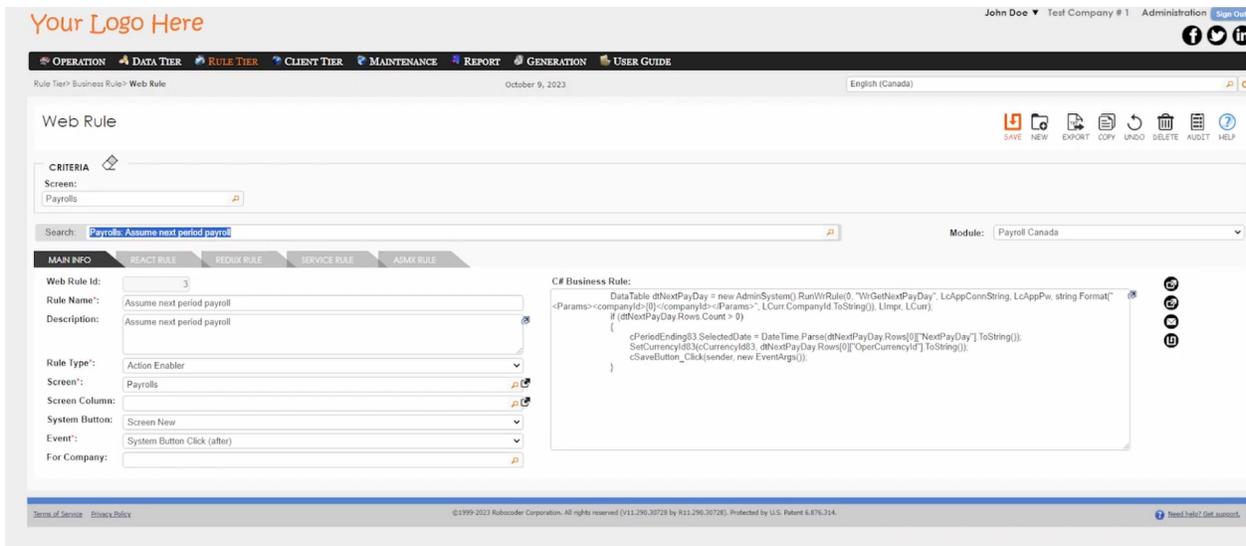
Below the summary is a table of employee payroll data:

Member	Employee Borne	Regular Hrs	Regular Pay	Vacation Hrs	Vacation Pay	Cpp Employee	EI Employee	Tax Federal	Tax Provincial	Deduction	Net Pay	Cpp Employer	EI Employer	Total Expense	Gross Pay
John Doe	0.00	104.00	7,916.67	0.00	0.00	453.69	129.04	1,106.53	552.24	2,241.50	5,675.17	453.69	180.66	634.35	7,916.67
Employee1	0.00	104.00	4,166.67	0.00	0.00	230.56	0.00	385.52	199.92	816.00	3,350.67	230.56	0.00	230.56	4,166.67
Employee2	0.00	104.00	7,916.67	0.00	0.00	453.69	129.04	1,106.53	552.24	2,241.50	5,675.17	453.69	180.66	634.35	7,916.67
	0.00	552.00	20,000.01	0.00	0.00	1,137.94	250.00	2,598.58	1,304.40	5,299.00	14,701.01	361.32	1,137.94	1,499.26	20,000.01

At the bottom of the interface, there is a footer with the text: 'Please be aware that CPP and EI can be a bit higher crossing a month-end due to taxable benefits recognition. Net pay can also be lower due to certain benefits borne by employee.'

## WebRule Analysis:

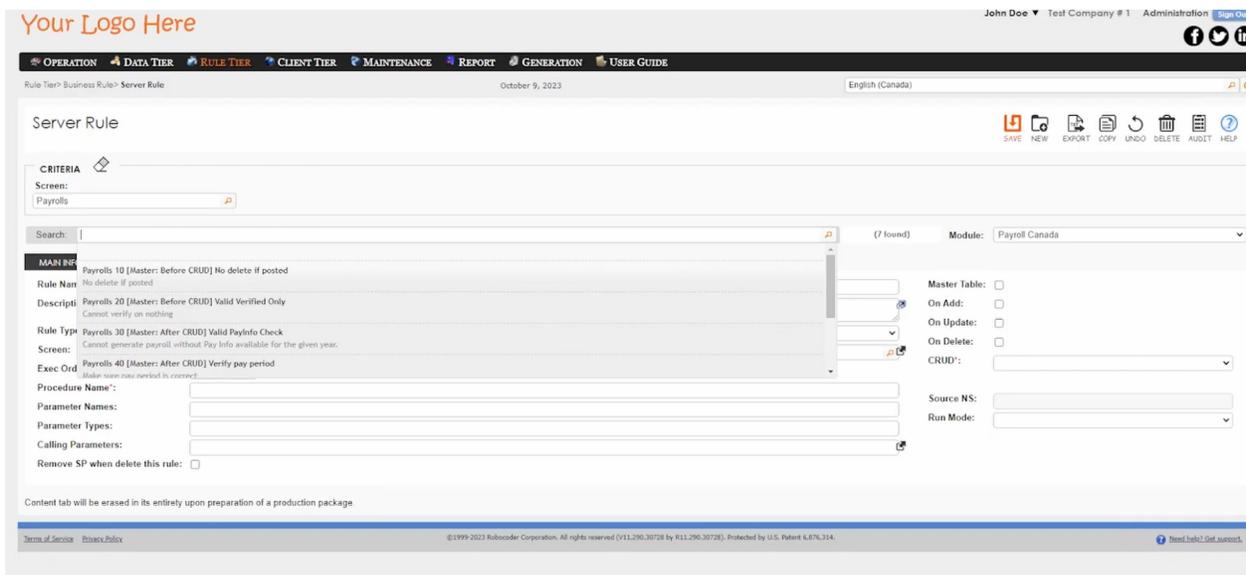
In our scrutiny of the system's rules, we pinpoint two noteworthy web rules. The initial one, "Assume next period payroll," is activated through the 'New' system button. This rule plays a crucial role in shaping how the system processes payroll for the upcoming period. Conversely, we encountered another WebRule associated with the transfer of payroll information to the Accounts Payable module. However, for the current analysis, this particular rule doesn't align with our primary objectives and will not be a focal point in our enhancement efforts.



With a comprehensive understanding of the webrules governing the periodic payroll screen, let's transition our focus to the examination of server rules. These rules play a critical role in shaping the backend processes that underpin the payroll functionalities. Let's delve into the intricacies of the server rule setup to glean further insights.

### Server Rules :

Now, let's dive into the detailed analysis of the server rules governing the Payroll Screen:



1. **CrValidVerifiedOnly:** This rule safeguards the integrity of payroll data by restricting changes to records that have been posted or finalized. This ensures the stability of payroll information once it has undergone necessary validations.

2. **CrVerifyPayPeriod:** A critical constraint, this rule guarantees that the correct pay period is being processed. By checking the last period completed in the PayInfo table, it maintains chronological accuracy in payroll processing.
3. **CrMaxCppEiExceed:** While currently inactive, this rule potentially plays a role in checking if the maximum CPP (Canada Pension Plan) is exceeded. Its non-utilization suggests a different mechanism or the absence of this specific check in the current payroll process.
4. **IrUpdPayrollDtl:** A pivotal server rule, this is responsible for intricate calculations. It fetches the CPP Contribution Percentage and EI (Employment Insurance) Insurable Percentage for the company, using this data to create detailed payroll information for the payroll record.
5. **IrPostToAp:** Focused on posting payroll to the Accounts Payable module, this rule is part of a subsequent system operation, beyond the primary concerns of our current analysis.

Now equipped with a thorough comprehension of the Payroll Screen intricacies, let's narrow our focus on the specific enhancement at hand. Our primary objective is to incorporate a warning system for users generating payroll when members approach or surpass the CPP/EI yearly maximum contributions. This warning mechanism will effectively communicate to users which members require salary updates.

To implement this change, our strategic approach involves expanding the functionality of the existing web rule, WebRule: Assume next period payroll. This rule, triggered by the 'New' button, will be modified to scrutinize CPP/EI yearly contributions for members, delivering a targeted warning when necessary.

The second phase of our enhancement initiative introduces a pivotal aspect—transmitting essential PDOC information to the CRA website and retrieving deduction figures to seamlessly update the salary tracking screen.

To delve into this, our immediate focus shifts to the Salary Tracking screen, a core component where members' salary information is meticulously recorded. This stage necessitates a meticulous inspection to ensure the seamless integration of PDOC data, subsequently updating deduction figures.

Populate necessary fields : this is the only web rule which sets the Valid To of the salary information that is being updated.

```
PCanSalaryModule.ascx.cs  PCanPayrollModule.ascx.cs
- RC.Web.PCanSalaryModule
- dPCanSalaryGrid_OnItemDataBound(object sender, ListViewItemEventArgs e)
2117 if ((cc.Attributes["OnClick"] == null || cc.Attributes["OnClick"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.
2118 cc = ((WebControl)lvi.FindControl("cCpExempt81"));
2119 if ((cc.Attributes["OnClick"] == null || cc.Attributes["OnClick"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.
2120 cc = ((WebControl)lvi.FindControl("cTDIFed81"));
2121 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2122 cc = ((WebControl)lvi.FindControl("cTDIProv81"));
2123 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2124 cc = ((WebControl)lvi.FindControl("cTaxFed81"));
2125 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2126 cc = ((WebControl)lvi.FindControl("cTaxProv81"));
2127 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2128 cc = ((WebControl)lvi.FindControl("cGrossPay81"));
2129 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2130 cc = ((WebControl)lvi.FindControl("cNetPay81"));
2131 if ((cc.Attributes["OnChange"] == null || cc.Attributes["OnChange"].IndexOf("ChkPgDirty") < 0) && cc.Visible && cc.Enabled) {cc.Attributes["OnChange"] += "document.
2132 }
2133 }
0 references | Douglas Thiesen, 27 days ago | 2 authors, 3 changes
2134 protected void cPCanSalaryGrid_OnItemDataBound(object sender, ListViewItemEventArgs e)
2135 {
2136 //WebRule: Populate necessary fields
2137 if (cPCanSalaryGrid.EditIndex > -1 && GetDataItemIndex(cPCanSalaryGrid.EditIndex) == e.Item.DataItemIndex)
2138 {
2139 //RoboCoder.WebControls.ComboBox cMid = (RoboCoder.WebControls.ComboBox)e.Item.FindControl("cMemberId81");
2140 //if (string.IsNullOrEmpty(cMid.SelectedValue)) { SetMemberId81(cMid, ?); }
2141 TextBox cTo = ((TextBox)e.Item.FindControl("cValidTo81"));
2142 if (string.IsNullOrEmpty(cTo.Text)) { cTo.Text = RO.Common3.Utils.fasShortDate("9999.12.31", base.LUSER.Culture); }
2143 }
2144 }
2145 // *** WebRule End *** //
2146 DataTable dt = (DataTable)Session[KEY_dtPCanSalaryGrid];
2147 bool isEditItem = false;
2148 DataView dvPCanSalaryGrid = dt != null ? dt.DefaultView : null;
2149 if (cPCanSalaryGrid.EditIndex > -1 && GetDataItemIndex(cPCanSalaryGrid.EditIndex) == e.Item.DataItemIndex)
2150 {
2151 isEditItem = true;

```

Our focus now narrows down to two key server rules, "Ir\_AddSalary" and "CrCheckValidDates," which play integral roles in updating and validating salary records for members. Let's delve into the specifics of each rule:

Server Rule:

**Ir\_AddSalary :** The primary objective of this server rule is to update the Valid To date of the salary record for a member.

**CrCheckValidDates :** This constraint server rule is designed to verify the validity of the dates associated with the salary record through a series of checks.

## Solution

Having concluded our thorough analysis of the screens slated for enhancement, we're now poised to move into the implementation phase, with a primary focus on the initial segment of our upgrade. Our objective is to actively alert users to any changes in deductions, prompting them to swiftly update salary information.

The implementation phase of the enhanced business rules for the Payroll Screen involves a strategic approach to integrating the necessary functionalities. Here's a step-by-step breakdown:

On the Payroll Screen, upon pressing the New button, conduct a check to determine if members for whom the payroll is generated are close to or exceeding the yearly CPP and EI exemption numbers.

Web Rule Enhancement:

- Expand the existing web rule (WebRule: Assume next period payroll) to integrate logic for checking CPP/EI yearly contributions.
- Introduce a mechanism to identify members close to or exceeding the CPP/EI maximum.
- Display a warning message for affected members, advising users to promptly update their salary information.

*Assume Next Period Payroll WebRule*



Microsoft Word  
Document

Changes to the Next Period Payroll WebRule accomplished:

- Navigate to the RC solution file in Visual Studio and open the cs file for PCanPayrollModule.ascx.cs to initiate revisions in the web rule.
- Revise the web rule to enhance the SaveDb method message, incorporating additional information regarding CPP and EI yearly exemption numbers.
- Establish an S.proc (**WrGetMaxCppEi**) to collect information on members from the database. This stored procedure provides essential details for members, including Year-to-Date (YTD) CPP and EI numbers required for subsequent comparisons.

*WrGetMaxCppEi Server Rule*



Microsoft Word  
Document

- Capture the result, which is a message containing information about CPP and EI and store it in the 'msg' variable.
- Iterate through the query results, examining conditions linked to CPP and EI contributions for each member. If certain conditions are met, such as reaching maximum contributions, add members to warning lists (warnCloseCPPToMax, warnCPPMax, warnCloseEIToMax, warnEIMax).
- If there are members in the warning lists, append messages about them to the 'msg' variable.
- Present the comprehensive message to the user as an advisory.

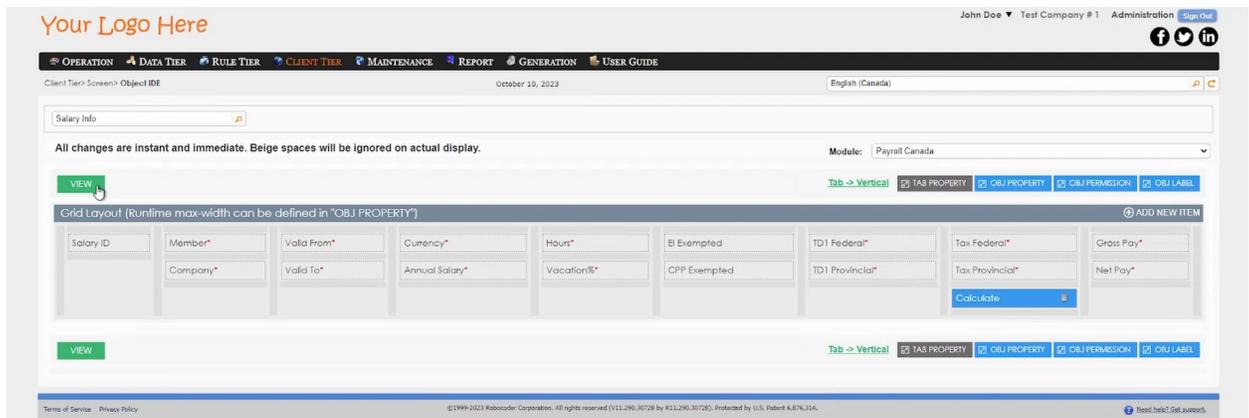
- Notify users that salary information needs updating for members mentioned in the message, emphasizing the critical importance of having updated deduction numbers.

This meticulous implementation ensures the successful integration of enhanced business rules, facilitating a seamless and proactive approach to managing CPP and EI contributions for employees within the Payroll Canada System.

In the second phase of our enhancement, the focus shifts towards seamlessly transmitting crucial PDOC information to the CRA website and updating deduction figures on the Salary Info screen.

### 1. Setting Up the 'Calculate' Button:

- A 'Calculate' button is introduced on the Salary Info Screen to facilitate the transmission of data to the PDOC Calculator.
- Navigate to the Object IDE to configure this button.



Screen Object Property Setup:



- Develop a method to interact with an external service (a Canadian tax-related service) to calculate tax-related information.
- Simulate user interaction with a web service, handling HTTP requests, cookies, and passing JSON payloads.

#### **Number Formatting Methods:**

- Implement methods to convert numbers to a formatted string, removing unnecessary decimal places.

The required setup of the WebRule have been provided in the word document above.

Implement server rules for the Salary Info Screen:

**Calculate PDOC:** Call PDOC and retrieve deductions.

**Custom PDOC Functions:** Custom function/procedure to provide information to CRA and receive deduction information back.

**Hide PDOC Calculate Button:**

Utilize the grid item data-bound event (before) to hide the calculate button until the row is edited. This code is part of a data-bound control's item data-binding event, checking if the item being bound is in edit mode.

This implementation ensures a smooth flow of data between the Salary Info Screen, the PDOC Calculator, and the external service (CanadaCRAPDOC), enabling real-time tax-related calculations and updates on the salary tracking screen.

### **3. Server Rule Setup:**

The Custom PDOC WebRule needs the payroll information from the database to be passed on the CRA website, this is accomplished by the server rule "WrGetPDOInfo". The s.proc gathers information of members to be relayed to CRA's PDOC website, which will calculate the correct tax deductions for the coming pay period.

*Server Rule Information*



Microsoft Word  
Document

Screen with the setup complete and ready to send information over to CRA to retrieve the tax deduction information.

Before information is sent to the CRA website:

Your Logo Here John Doe Test Company # 1 Payroll Canada Sign Out

Payroll Benefit Reports Documents

Payroll Salary Tracking October 11, 2023 English (Canada)

### Salary Info

CRITERIA  
Member: John Doe

ADD Filter: All On: 15 of 1 (1 found)

Member	Company	Valid From	Valid To	Currency	Annual Salary	Hours	Vacation%	El Exempted	CPP Exempted	*TD1 Federal	*TD1 Provincial	*Tax Federal	*Tax Provincial	Gross Pay	Net Pay
John Doe	Test Company # 1	2023-08-01	9999-12-31	Canadian Dollar	95000	174	8	<input type="checkbox"/>	<input type="checkbox"/>	15000	11981			7916.67	
John Doe	Test Company # 1	2022-12-01	9999-12-31	Canadian Dollar	95,000.00	174.00	8	<input type="checkbox"/>	<input type="checkbox"/>	15,000.00	11,865.00	1,106.53	552.24	7,916.67	5,675.17

Unless this is the first/end salary, please make sure salary period starts on the first day of a pay period and ends on the last day of a pay period or 9999-12-31.

Terms of Service Privacy Policy © 1999-2023 Robocoder Corporation. All rights reserved (V2.27.30928 by R11.290.30728). Protected by U.S. Patent 6,876,314. Need help? Get support.

Fields populated with the response from the CRA PDOC page,

Your Logo Here John Doe Test Company # 1 Payroll Canada Sign Out

Payroll Benefit Reports Documents

Payroll Salary Tracking October 11, 2023 English (Canada)

### Salary Info

CRITERIA  
Member: John Doe

ADD Filter: All On: 15 of 1 (1 found)

Member	Company	Valid From	Valid To	Currency	Annual Salary	Hours	Vacation%	El Exempted	CPP Exempted	*TD1 Federal	*TD1 Provincial	*Tax Federal	*Tax Provincial	Gross Pay	Net Pay
John Doe	Test Company # 1	2023-08-01	9999-12-31	Canadian Dollar	95000	174	8	<input type="checkbox"/>	<input type="checkbox"/>	15000	11981	1119.15	448.07	7916.67	6224.52
John Doe	Test Company # 1	2022-12-01	9999-12-31	Canadian Dollar	95,000.00	174.00	8	<input type="checkbox"/>	<input type="checkbox"/>	15,000.00	11,865.00	1,106.53	552.24	7,916.67	5,675.17

Unless this is the first/end salary, please make sure salary period starts on the first day of a pay period and ends on the last day of a pay period or 9999-12-31.

Terms of Service Privacy Policy © 1999-2023 Robocoder Corporation. All rights reserved (V2.27.30928 by R11.290.30728). Protected by U.S. Patent 6,876,314. Need help? Get support.

This concludes the integration Payroll Deductions Online Calculator with the Payroll Canada Screen.