

Problem

Within the Payroll Canada System, a critical void was identified in the T4 Slip functionality, impacting the accuracy and completeness of crucial financial documentation. The existing structure lacked two pivotal fields—EI Insurable Earnings (Box #24) and CPP Pensionable Earnings (Box #26) in the T4 Slip.

The image shows a T4 Statement of Remuneration Paid form. The form is titled "T4 Statement of Remuneration Paid / État de la rémunération payée". It includes fields for Employer's name, Year, Employment income, Income tax deducted, and various contribution boxes (14, 16, 17, 18, 20, 22, 24, 26, 28, 29, 44, 46, 50, 52, 55, 56). It also features a section for Employee's name and address, and a bottom section for other information and amounts. The form is labeled "Protected B when completed / Protégé B une fois rempli" on the left side.

This absence posed a substantial challenge as it hindered the system's ability to comprehensively capture and reflect employee earnings, particularly in relation to Employment Insurance (EI) and Canada Pension Plan (CPP) contributions.

The subsequent sections will provide insights into the process of integrating EI Insurable Earnings and CPP Pensionable Earnings into the T4 Slip and the transformative impact it had on resolving this inherent problem.

Objective

The primary objective of this initiative is to seamlessly integrate two indispensable fields, namely EI Insurable Earnings (Box #24) and CPP Pensionable Earnings (Box #26), into the T4 Slip of the Payroll Canada System. These fields play a pivotal role in reflecting the accurate financial picture of employees, specifically concerning Employment Insurance (EI) and Canada Pension Plan (CPP) contributions.

Our focus is on establishing meticulous formulas for the computation of these fields, ensuring precision and compliance. The EI Insurable Earnings (Box #24) will be derived by dividing the Employee EI Premium (Box #18) by the EI Premium Rate. This rate is dynamically influenced by changes in the

Employee EI Premium, which represents the amount deducted from an employee's earnings in a given year.

The formula for calculating the same are given below,

$$\text{EI Insurable Earnings (24)} = \text{Employee EI Premium (18)} / \text{EI Premium Rate}$$
$$\text{CPP Pensionable Earnings (26)} = (\text{Employee CPP Contribution (16)} / \text{CPP Contribution Rate}) + \text{CPP Exempt Amt}$$

Simultaneously, the CPP Pensionable Earnings (Box #26) will be computed by combining two critical components. Firstly, the Employee CPP Contribution (Box #16) divided by the CPP Contribution Rate, representing the amount contributed to the Canada Pension Plan during the year. Secondly, the CPP Exempt Amount will be factored into the calculation. It is important to note that both the EI insurable earnings rate and the CPP contribution percentage rate undergo annual variations.

The implementation of these refined formulas is not merely an enhancement; it is a strategic imperative for accurate and transparent T4 reporting. By seamlessly incorporating these calculations, our aim is to fortify the Payroll Canada System's ability to generate precise T4 Slips, thereby ensuring compliance and providing employees with a thorough and accurate representation of their EI and CPP-related financial information.

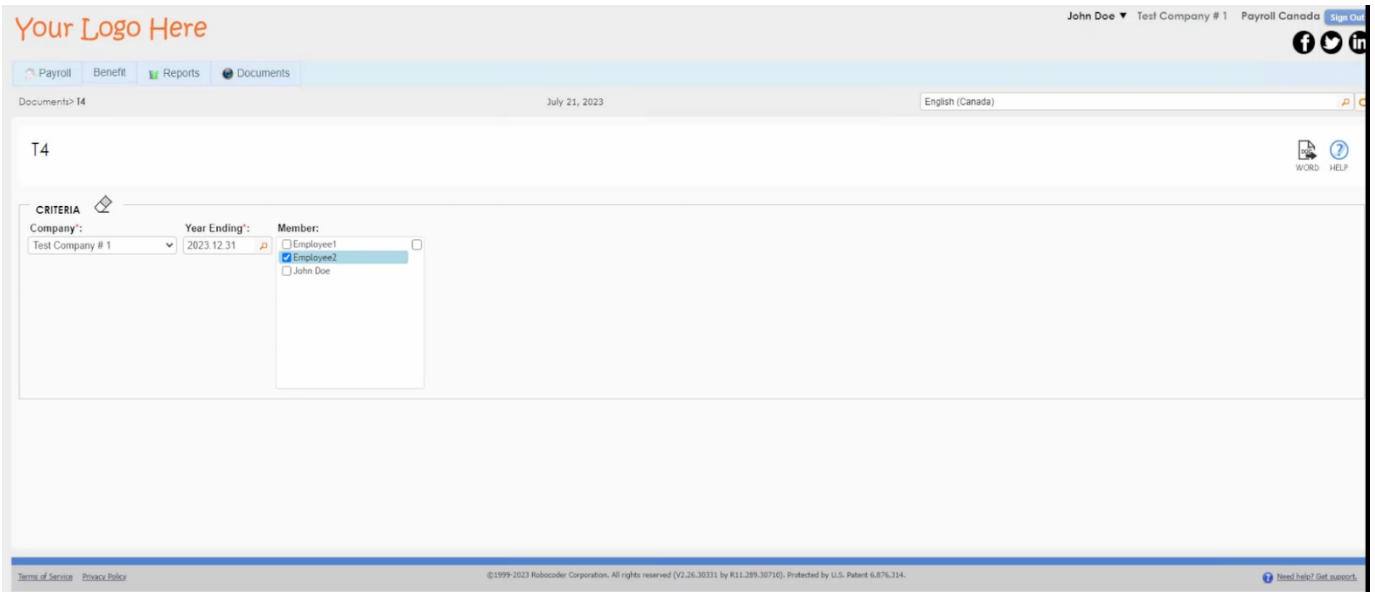
Analysis

In our pursuit of enhancing the T4 Slip in the Payroll Canada System to include EI Insurable Earnings (Box 24) and CPP Pensionable Earnings (Box 26), a thorough analysis of the current system setup is imperative.

Here's a step-by-step breakdown of our exploration:

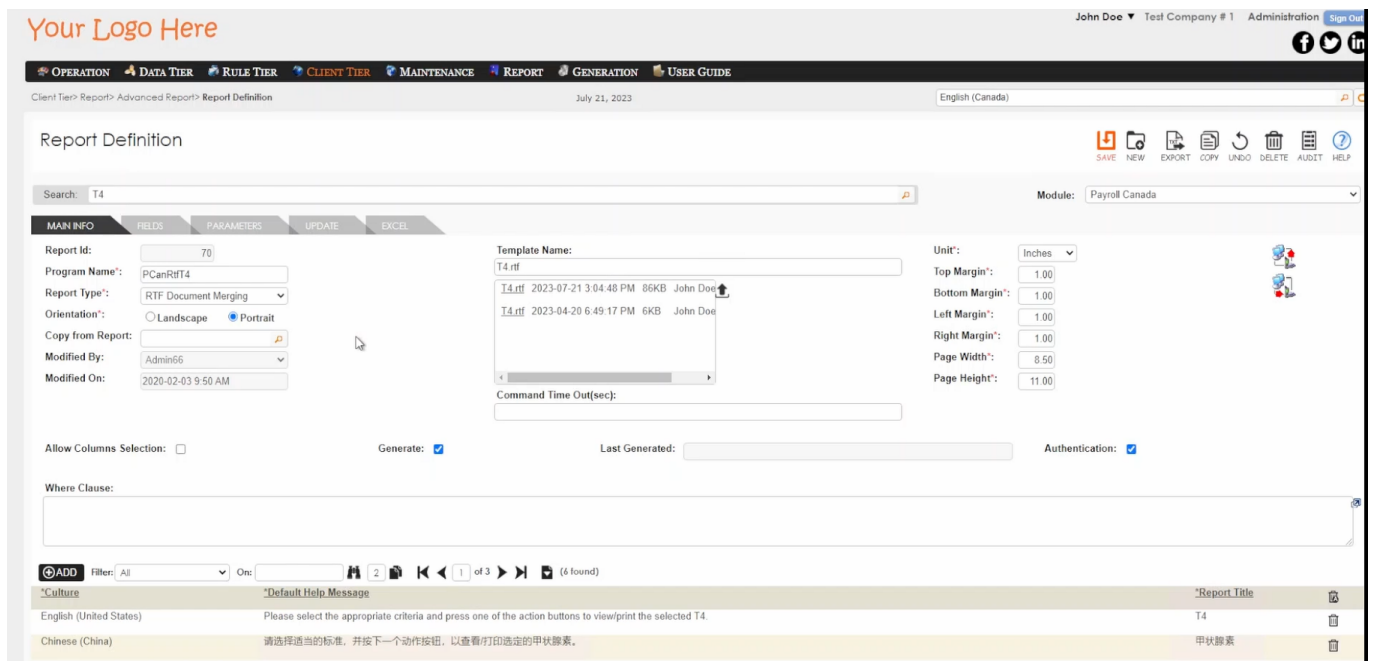
1. Review of T4 Report in the Payroll Canada System:

- We initiated the analysis by navigating to the Payroll Canada System and locating the T4 report under the Documents Menu.
- The T4 Slip for Employee 2, as of the Year Ending 2023, was generated for examination.



2. Report Setup Exploration:

The setup of the system was scrutinized in Administration-Client Tier\Report\Advanced Report\Report Definition.



The T4 report was identified as an RTF Document Merging type, utilizing a template with specific tags.

The image shows two side-by-side screenshots of a report template in a word processing application. The left screenshot shows a form with various fields for 'Test Company#1' and '[[T4Year]]'. The right screenshot shows the same form with data populated for the year 2023.

Test Company#1		[[T4Year]]	
8XXXXX 0XXX RP0001		[[EmpIncome]]	[[TaxDeduction]]
		[[ProvState]]	[[CppEmployee]]
[[SocialNumber]]	[[CX]]	[[EX]]	
[[LastName]]		[[FirstName]]	[[EiEmployee]]
[[Address]]			
[[City]], [[ProvState]] [[Postal]]			
[[Country]]			
40	[[TaxableBenefit]]		

Test Company#1		2023	
8XXXXX 0XXX RP0001		25,750.01	4,976.31
		NS	1,361.07
1231312322			
2	Employee	387.12	
900 Main St HALIFAX, NS CANADA			
40	0.00		

3. Identification of Relevant Tags:

Tags "CppEmployee" and "EiEmployee" were identified in the report template, presumed to correspond to Employee CPP contribution and Employee EI premium. This was done through the validation through SQL Profiler. SQL Profiler was set up to capture and analyze database calls during T4 report execution.

The stored procedure (_GetPayRtft4R) was examined, confirming the retrieval of Employee EI premium and Employee CPP contribution from the T4 table.

```
USE [RCPCan]
GO
/***** Object: StoredProcedure [dbo].[_GetPayRtft4R]    Script Date: 2023-10-27 6:02:13 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[_GetPayRtft4R]
    @wClause      varchar(4000)
    ,@YearEnding   datetime -- not null
    ,@CompanyId    int -- not null
/* WITH ENCRYPTION */
AS
DECLARE @sClause  varchar(8000)
        ,@fClause  varchar(8000)
        ,@oClause  varchar(8000)
        ,@ReportTimeTzInfo nvarchar(100)
SET NOCOUNT ON

SELECT @ReportTimeTzInfo = CONVERT (VARCHAR, RCDesign.dbo.fConvertFromUTC(GETUTCDATE()),NULL,r.val))
                        + ' (UTC ' + CONVERT(varchar,r.val) + ')'
FROM #ReportTemp r
WHERE r.Name = 'TZutcOffset'

IF NOT EXISTS (SELECT 1 FROM master.dbo.sysdatabases WHERE name = 'RCHrs')
BEGIN RAISERROR('Please make sure database "RCHrs" exists and try again.',18,2) WITH SETERROR RETURN 1 END
EXEC dbo.MkT4 @CompanyId, @YearEnding

SELECT @sClause = 'SELECT g.CompanyDesc,T4Year=convert(varchar,datepart(yyyy,c.YearEnding)),d.CurrencyName,a.LastName,a.FirstName'
+ ',a.SocialNumber,Address=b.Address1 + ISNULL(SPACE(1)+b.Address2,SPACE(0))'
+ ',City=t.CityName,ProvState=e.StateCode,Postal=b.PostalZip,Country=f.CountryName'
+ ',CX=CASE WHEN c.CppExempt=''Y'' THEN ''X'' ELSE '''' END,EX=CASE WHEN c.EiExempt=''Y'' THEN ''X'' ELSE '''' END'
+ ',c.CppEmployee,c.EiEmployee,c.TaxableBenefit,c.EmplIncome,c.TaxDeduction'
+ ',ReportTimeTzInfo = ''' + @ReportTimeTzInfo + ''''
SELECT @fClause = 'FROM RCCmon.dbo.Member a'
+ ' INNER JOIN RCCmon.dbo.Company g ON a.CompanyId = g.CompanyId'
+ ' INNER JOIN RCCmon.dbo.Addr b ON a.FirmId = b.FirmId AND DefAddr = ''Y'''
+ ' INNER JOIN RCCmon.dbo.State e ON b.StateId = e.StateId'
+ ' INNER JOIN RCCmon.dbo.City t ON b.CityId = t.CityId'
+ ' INNER JOIN RCCmon.dbo.Country f ON e.CountryId = f.CountryId'
+ ' INNER JOIN RCCmon.dbo.T4 c ON a.MemberId = c.MemberId'
+ ' INNER JOIN RCCmon.dbo.Currency d ON c.CurrencyId = d.CurrencyId'
+ ' INNER JOIN RcPCan.dbo.PayInfo i ON i.CompanyId = a.CompanyId AND i.PayYear = ''' + convert(varchar,YEAR(@YearEnding)) + ''''
SELECT @wClause = @wClause + ' AND a.CompanyId = ' + convert(varchar,@CompanyId)
SELECT @oClause = 'ORDER BY c.MemberId'
PRINT (@sClause + ' ' + @fClause + ' ' + @wClause + ' ' + @oClause)
EXEC (@sClause + ' ' + @fClause + ' ' + @wClause + ' ' + @oClause)
RETURN 0
```

5. Exploration of T4 Table and its Population Process:

Upon analyzing the T4 table and its population process, we discovered a stored procedure called Mkt4. This procedure is responsible for deleting old records with the same date and replacing them with new ones. It achieves this by utilizing data from the Payroll and PayrollDtl tables for the selected year, allowing the calculation of the yearly Employee EI Premium and Employee CPP Contribution each time the T4 report is generated.

```
USE [RCPCan]
GO
/***** Object:  StoredProcedure [dbo].[Mkt4]    Script Date: 2023-10-27 6:04:13 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[Mkt4]
    @CompanyId smallint
    ,@YearEnding datetime
/* WITH ENCRYPTION */
AS
DECLARE @T4Year int
        ,@YearFr datetime
        ,@YearTo datetime
        ,@PeriodEnding datetime
SET NOCOUNT ON
SELECT @T4Year = datepart(yyyy,@YearEnding)
SELECT @PeriodEnding = MAX(PeriodEnding) FROM dbo.Payroll WHERE CompanyId = @CompanyId AND datepart(yyyy,PeriodEnding) = @T4Year
DELETE FROM dbo.T4 WHERE CompanyId = @CompanyId AND YearEnding = @YearEnding
/*
IF (SELECT COUNT(distinct BenefitDate) FROM dbo.Benefit WHERE CompanyId = @CompanyId AND datepart(yyyy,BenefitDate) = @T4Year AND Posted = 'Y') < 12
BEGIN
    RAISERROR('Some benefits has not been posted for calendar year %d, please try again later.',18,2,@T4Year) WITH SETERROR
    RETURN 1
END
*/
SELECT @YearFr = convert(datetime,convert(varchar,@T4Year) + '.01.01')
        ,@YearTo = convert(datetime,convert(varchar,@T4Year) + '.12.31')
INSERT INTO dbo.T4 (YearEnding,CurrencyId,MemberId,b.EiExempt,b.CppExempt,TaxableBenefit,RegularPay,VacationPay,EmplIncome,CppEmployee,EiEmployee,CppEmployer,EiEmployer,TaxFed,TaxProv,TaxDeduction,CompanyId)
SELECT @YearTo,c.CurrencyId,a.MemberId,b.EiExempt,b.CppExempt,TaxableBenefit=ISNULL(d.TaxableBenefit,0),SUM(a.RegularPay),SUM(a.VacationPay),ISNULL(d.TaxableBenefit,0) + SUM(a.GrossPay)
        ,SUM(a.CppEmployee),SUM(a.EiEmployee),SUM(a.CppEmployer),SUM(a.EiEmployer),SUM(a.TaxFed),SUM(a.TaxProv),SUM(a.TaxFed + a.TaxProv),@CompanyId
FROM dbo.PayrollDtl a
INNER JOIN dbo.Salary b ON a.MemberId = b.MemberId
INNER JOIN dbo.Payroll c ON a.PayrollId = c.PayrollId
LEFT OUTER JOIN (SELECT b2.MemberId, TaxableBenefit = SUM(b2.BenefitDetAmt + b2.BenefitDetGst)
FROM dbo.Benefit b1
INNER JOIN dbo.BenefitDet b2 ON b1.BenefitId = b2.BenefitId
INNER JOIN dbo.BenefitType b3 ON b2.BenefitTypeId = b3.BenefitTypeId AND b3.Taxable = 'Y'
WHERE datepart(yyyy,b1.BenefitDate) = @T4Year GROUP BY b2.MemberId) d ON a.MemberId = d.MemberId
WHERE c.CompanyId = @CompanyId AND c.PeriodEnding >= @YearFr AND c.PeriodEnding <= @YearTo AND b.ValidFrom <= @PeriodEnding AND b.ValidTo >= @PeriodEnding
GROUP BY c.CurrencyId,a.MemberId,b.EiExempt,b.CppExempt,TaxableBenefit
RETURN 0
```

6. Payroll Tables and Web Rules:

The Payroll tables play a crucial role in the Payroll Screen, as confirmed through the Administration/Client Tier/Screen/Screen Definition.

Your Logo Here

John Doe ▼ Test Company # 1 Payroll Canada Sign Out

Payroll Benefit Reports Documents

Payroll Periodic Payroll July 21, 2023 English (Canada)

Payrolls

Search: 0023.0351.KRC

Period Ending: March 2023

Company: Test Company # 1

Currency: Canadian Dollar

Finalized: ☐

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 Employee: 1,137.94

Total Cpp Employer: 1,137.94

Total EI Employee: 361.32

Total EI Employer: 258.00

Total Net Pay: 14,701.01

Total Employee Borne: 0.00

Member	Employee Borne	Regular Hrs	Vacation Hrs	Regular Pay	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	184.00	0.00	7,916.67	0.00	453.69	129.04	1,106.53	552.24	2,241.50	2,241.50	453.69	180.66	634.35	7,916.67
Employee1	0.00	184.00	0.00	4,166.67	0.00	230.56	0.00	385.52	199.92	816.00	3,350.67	0.00	0.00	4166.67	4,166.67
Employee2	0.00	184.00	0.00	7,916.67	0.00	453.69	129.04	1,106.53	552.24	2,241.50	2,241.50	453.69	180.66	634.35	7,916.67
	0.00	552.00	0.00	20,000.01	0.00	1,137.94	258.00	2,598.58	1,304.40	5,299.00	14,701.01	1,137.94	361.32	1,499.26	20,000.01

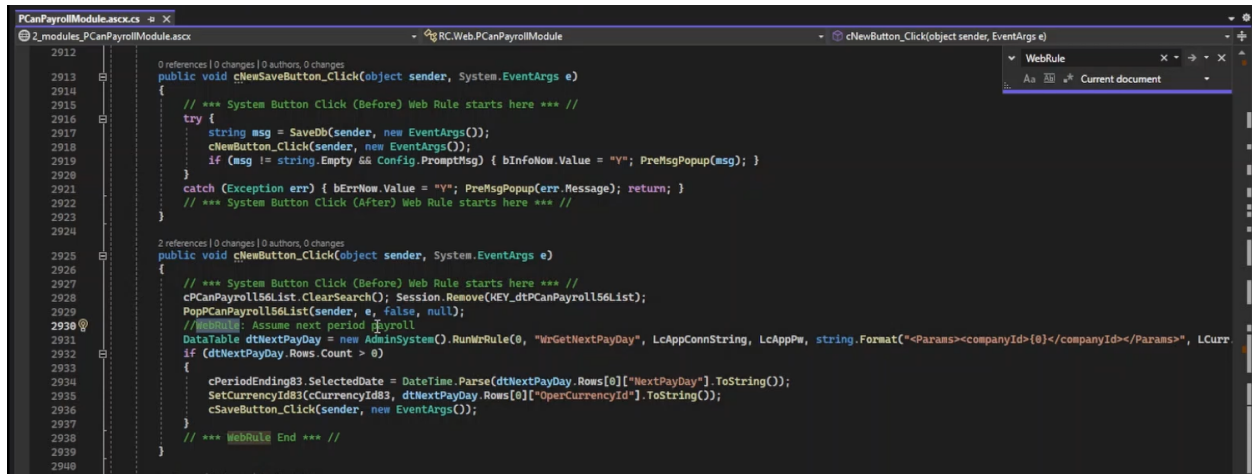
Please be aware that CPP and EI can be a bit higher crossing a month-end due to taxable benefits recognition; Net pay can also be lower due to certain benefits borne by employee.

Terms of Service Privacy Policy

©1999-2023 Robocorder Corporation. All rights reserved (V0.26.30321 by #513.289.30710). Protected by U.S. Patent 6,676,314.

Need help? Get support.

To understand the functionality and associated business rules, we can examine the webrules by opening the solution in Visual Studio and reviewing the PCanPayrollModule.ascx.cs file. Within this file, we find two web rules of interest.

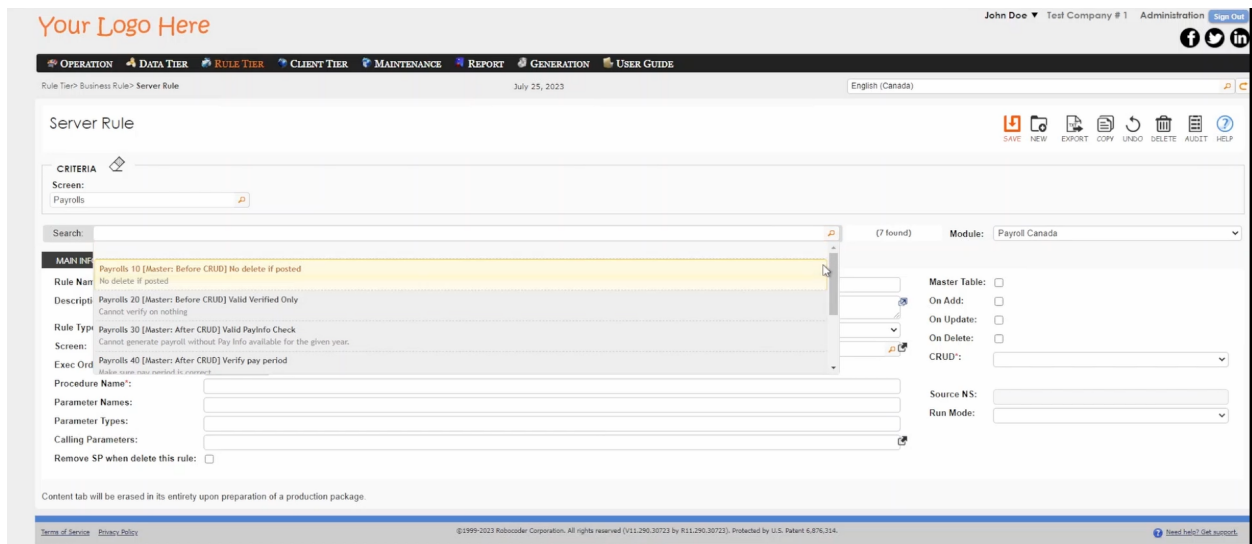


```
2912
2913 0 references | 0 changes | 0 authors, 0 changes
2914 public void cNewSaveButton_Click(object sender, System.EventArgs e)
2915 {
2916     // *** System Button Click (Before) Web Rule starts here *** //
2917     try {
2918         string msg = SaveDb(sender, new EventArgs());
2919         cNewButton_Click(sender, new EventArgs());
2920         if (msg != string.Empty && Config.PromptMsg) { bInfoNow.Value = "Y"; PrefMsgPopup(msg); }
2921     }
2922     catch (Exception err) { bErrNow.Value = "Y"; PrefMsgPopup(err.Message); return; }
2923     // *** System Button Click (After) Web Rule starts here *** //
2924 }
2925
2926 2 references | 0 changes | 0 authors, 0 changes
2927 public void cNewButton_Click(object sender, System.EventArgs e)
2928 {
2929     // *** System Button Click (Before) Web Rule starts here *** //
2930     cPCanPayroll1561List.ClearSearch(); Session.Remove(KEY_dtPCanPayroll1561List);
2931     PopPCanPayroll1561List(sender, e, false, null);
2932     //WebRule: Assume next period payroll
2933     DataTable dtNextPayDay = new AdminSystem().RunMrRule(0, "WrGetNextPayDay", LcAppConnString, LcAppPw, string.Format("<Params><companyId>{0}</companyId></Params>", LcCur
2934     if (dtNextPayDay.Rows.Count > 0)
2935     {
2936         cPeriodEnding33.SelectedDate = DateTime.Parse(dtNextPayDay.Rows[0]["NextPayDay"].ToString());
2937         SetCurrencyId33(cCurrencyId33, dtNextPayDay.Rows[0]["OperCurrencyId"].ToString());
2938         cSaveButton_Click(sender, new EventArgs());
2939     }
2940     // *** WebRule End *** //
2941 }
```

The first is triggered by the New Button Click and calls the S.Proc "WrGetNextPayDay," which is essential for determining the next applicable pay period.

The second WebRule, cPostToAp_Click, finalizes, and posts the payroll to A/P, serving as a secondary step after payroll creation which is not of interest to us at this time.

Additionally, for further insights, we can review the server rules in Administration System under the Rule Tier Menu - Server Rule.



These are the business rules associated with the payroll screen. We can get more information on reading the description of each rule. The first three rules are constraints doing various check to ensure proper payroll is generated.

a. CrVerifyPayPeriod is a constraint that ensures the correct pay period is being processed by checking the last period completed in the PayInfo table.

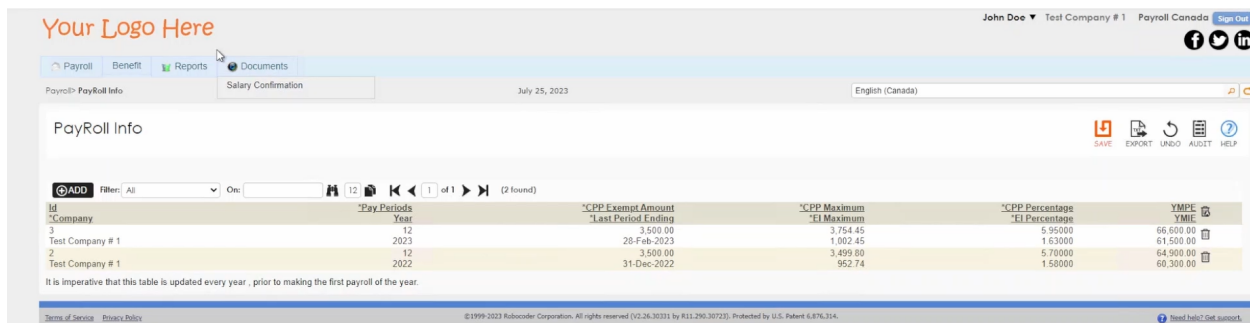
b. CrMaxCppEiExceed is a constraint that checks if the maximum CPP (Canada Pension Plan) is not exceeded.

c. IrUpdPayrollDtl is a stored procedure that utilizes the PayInfo table to fetch the CPP Contribution Percentage and EI (Employment Insurance) Insurable Percentage for the company. It then creates detailed payroll information for the payroll record.

d. IrPostToAp posts the payroll to the AP (Accounts Payable) module if the payroll is finalized. However, this is not a primary concern for us at the moment.

7. PayInfo Screen Exploration:

The PayInfo Screen, capturing yearly information, holds crucial data like EI Premium Rate and CPP Contribution Rate needed for our calculations.



Id	Company	Pay Periods	Year	*CPP Exempt Amount	*Last Period Ending	*CPP Maximum	*EI Maximum	*CPP Percentage	*EI Percentage	YMPE	YIME
3	Test Company #1	12	2023	3,500.00	28-Feb-2023	3,754.45	1,002.45	5.95000	1.63000	66,600.00	61,500.00
2	Test Company #1	12	2022	3,500.00	31-Dec-2022	3,499.80	952.74	5.70000	1.58000	64,900.00	60,300.00

Having analyzed the existing setup, including database structure, report generation, and associated business rules, we are well-equipped to proceed with the strategic enhancement of integrating EI Insurable Earnings and CPP Pensionable Earnings into the T4 report. This comprehensive analysis ensures a clear understanding of the system's intricacies and informs our implementation strategy.

Solution:

Having gained a clear understanding of how the payroll is generated and how the T4 report utilizes this information to create the slip, we are now ready to proceed with our implementation. This involves including the two additional columns in the T4 report as per our requirements.

Creation of New Report Objects:

- Navigate to the Client Tier > Report > Report Obj. section in the Payroll Canada System.
- Duplicate the existing 'EiEmployee' object to create two new report objects: 'EilnsEarnings' (EI Insurable Earnings) and 'CppPenEarnings' (CPP Pensionable Earnings).
- Modify the new objects by adjusting the Column Name and setting the Tab Order to 1700 and 1800, respectively.
- Update the Column Header in the detail section to display "EI Insurable Earnings / CPP Pensionable Earnings."

The screenshot shows the 'Report Object' configuration page in the Payroll Canada System. The page has a top navigation bar with tabs: OPERATION, DATA TIER, RULE TIER, CLIENT TIER, MAINTENANCE, REPORT, GENERATION, and USER GUIDE. The 'REPORT' tab is selected. Below the navigation bar, there's a breadcrumb trail: Client Tier > Report > Advanced Report > Report Object. The main content area is titled 'Report Object' and contains a form for configuring a report object. The form includes fields for 'Report' (set to 'T4'), 'Column Name' (set to 'CppPenEarnings'), 'Column Type' (set to 'Field'), 'Tab Order' (set to '1800'), 'Column Format' (set to '#,##0.00'), 'Padding Size' (empty), 'With Character' (empty), 'Data Type' (set to 'Money'), 'Criteria Operator' (empty), and 'Criteria Column' (empty). There are also buttons for 'SAVE', 'NEW', 'EXPORT', 'COPY', 'UNDO', 'DELETE', 'ADGET', and 'HELP'. At the bottom, there's a table with columns 'Column Header' and 'Width'. The table has one row with the header 'EI Insurable Earnings' and a width of '100%'. The footer of the page contains copyright information: ©1999-2023 RoboSource Corporation. All rights reserved (V11.290.30723 by R11.290.30723). Protected by U.S. Patent 6,876,214.

Modification of Stored Procedure (_GetPayRtft4R):

- Modify the stored procedure (_GetPayRtft4R) to integrate the newly added report objects using the provided formulas.
- Include the PayInfo table in the stored procedure, which houses crucial data such as EI Premium Rate and CPP contribution Rates, necessary for accurate calculations.


```

/* WITH ENCRYPTION */
AS
DECLARE @sClause      varchar(8000)
        ,@fClause      varchar(8000)
        ,@oClause      varchar(8000)
        ,@ReportTimeTzInfo nvarchar(100)
SET NOCOUNT ON

SELECT @ReportTimeTzInfo = CONVERT (VARCHAR, RCDesign.dbo.fConvertFromUTC(GETUTCDATE(),NULL,r_val))
        + ' (UTC ' + CONVERT(varchar,r_val) + ')'
FROM #ReportTemp r
WHERE r.Name = 'TZUtcOffset'

IF NOT EXISTS (SELECT 1 FROM master.dbo.sysdatabases WHERE name = 'RCHrs')
BEGIN RAISERROR('Please make sure database "RCHrs" exists and try again.',18,2) WITH SETERROR RETURN 1 END
EXEC dbo.MKT4 @CompanyId, @YearEnding
SELECT @sClause = 'SELECT g.CompanyDesc,T4Year=convert(varchar,datepart(yyyy,c.YearEnding)),d.CurrencyName,a.LastName,a.FirstName'
+ ',a.SocialNumber,Address=b.Address1 + ISNULL(SPACE(1)+b.Address2,SPACE(0))'
+ ',City=t.CityName,ProvState=e.StateCode,Postal=b.PostalZip,Country=f.CountryName'
+ ',CX=CASE WHEN c.CppExempt=''Y'' THEN ''X'' ELSE '''' END,EX=CASE WHEN c.EiExempt=''Y'' THEN ''X'' ELSE '''' END'
+ ',c.CppEmployee,c.EiEmployee,c.TaxableBenefit,c.EmplIncome,c.TaxDeduction'
+ ',ReportTimeTzInfo =''' + @ReportTimeTzInfo + ''''
+ ',EiInsEarnings = ROUND(Convert(MONEY,c.EiEmployee/(i.EiPerc/100)),2), CppPenEarnings = ROUND(Convert(MONEY,(c.CppEmployee/(i.CppPerc/100)) + CppExemptAmt),2) '
SELECT @fClause = 'FROM RCCmon.dbo.Member a'
+ ' INNER JOIN RCCmon.dbo.Company g ON a.CompanyId = g.CompanyId'
+ ' INNER JOIN RCCmon.dbo.Address b ON a.FirmId = b.FirmId AND DefAddr = ''Y'''
+ ' INNER JOIN RCCmon.dbo.State e ON b.StateId = e.StateId'
+ ' INNER JOIN RCCmon.dbo.City t ON b.CityId = t.CityId'
+ ' INNER JOIN RCCmon.dbo.Country f ON e.CountryId = f.CountryId'
+ ' INNER JOIN dbo.T4 c ON a.MemberId = c.MemberId'
+ ' INNER JOIN RCCmon.dbo.Currency d ON c.CurrencyId = d.CurrencyId'
+ ' INNER JOIN RCCmon.dbo.PayInfo i ON i.CompanyId = a.CompanyId AND i.PayYear = ''' + YEAR(@YearEnding) + ''''

SELECT @wClause = @sClause + ' AND a.CompanyId = ' + convert(varchar,@CompanyId)
SELECT @oClause = 'ORDER BY c.MemberId'
PRINT (@sClause + ' ' + @fClause + ' ' + @wClause + ' ' + @oClause)
EXEC (@sClause + ' ' + @fClause + ' ' + @wClause + ' ' + @oClause)
RETURN 0

```

3. Template Update:

- Update the template by including tags with the same Object Name ('EiInsEarnings' and 'CppPenEarnings').

Test Company#1				[[T4Year]]	
			[[EmpIncome]]		[[TaxDeduction]]
8XXXXXX 0XXX RP0001					
			[[ProvState]]	[[CppEmployee]]	[[EiInsEarnings]]
[[SocialNumber]]	[[CX]]	[[EX]]	[[CppPenEarnings]]		
[[LastName]]		[[FirstName]]		[[EiEmployee]]	
[[Address]]					
[[City]], [[ProvState]]		[[Postal]]			
[[Country]]					
40	[[TaxableBenefit]]				

- Upload the modified template file to Administration-Client Tier\Report\Advanced Report\Report Definition. The system automatically utilizes the latest template during report execution.

Report Definition

Search: T4

Module: Payroll Canada

Report Id: 70

Program Name: PCanRptT4

Report Type: RTF Document Merging

Orientation: ☐ Landscape ☒ Portrait

Copy from Report:

Modified By: Admin66

Modified On: 2020-02-03 9:50 AM

Template Name: T4.rtf

Unit: Inches

Top Margin: 1.00

Bottom Margin: 1.00

Left Margin: 1.00

Right Margin: 1.00

Page Width: 8.50

Page Height: 11.00

Command Time Out(sec): 0

Allow Columns Selection: ☐ Generate: ☒ Last Generated: Authentication: ☒

Where Clause:

Report Title	Report ID	Report Name
*Culture	*Default Help Message	
English (United States)	Please select the appropriate criteria and press one of the action buttons to view/print the selected T4.	T4
Chinese (China)	请选择适当的标准，并按下一个动作按钮，以查看打印选定的甲状腺素。	甲状腺素

After implementing the changes, we verify that the modifications produce the expected results and that the newly added columns ('EI Insurable Earnings' and 'CPP Pensionable Earnings') accurately reflect the calculations based on EI Premium Rate and CPP Contribution Rates.

By following these steps, we ensure the seamless integration of the new report objects, EI Insurable Earnings and CPP Pensionable Earnings, into the T4 report. This systematic approach, from object creation to testing, guarantees the reliability and functionality of the enhanced T4 Slip in the Payroll Canada System.

This concludes the addition of columns on the T4 Tax Slip in the Payroll Canada System.

Final run of the report containing the new fields.

Test Company#1

2023

				23,750.01	4,976.31
8XXXXXX 0XXX RP0001					
			NS	1,361.07	23,749.69
1231312322					26,375.13
2		Employee		387.12	
900 Main St HALIFAX, NS CANADA					
40	0.00				