
Delinquents? Whip them into shape with Oracle Receivables

Greg Fookes

Manager

CGEYNZ Ltd



INTRODUCTION.....	2
MOVE STAFF OUT OF THE BACK OFFICE AND INTO THE FRONT.....	3
AUTOMATIC CREDIT ALLOCATION	3
AUTOMATIC RECEIPT APPLICATION.....	4
CREDIT REFUNDS	6
GO GET THAT MONEY	8
TIME PAYMENT ARRANGEMENTS – ‘WE’RE HERE TO HELP YOU!’	8
PENALTIES – ‘THE BIG STICK’	11
DEBT TRANSFER – ‘PAY UP OR WE’LL ...’	13
CONCLUSION	15

Introduction

Oracle Receivables can be easily extended to improve your back office efficiencies, allowing you to focus on collecting delinquent debt to increase your cash flow. We have designed and built custom extensions to Receivables that allow you to:

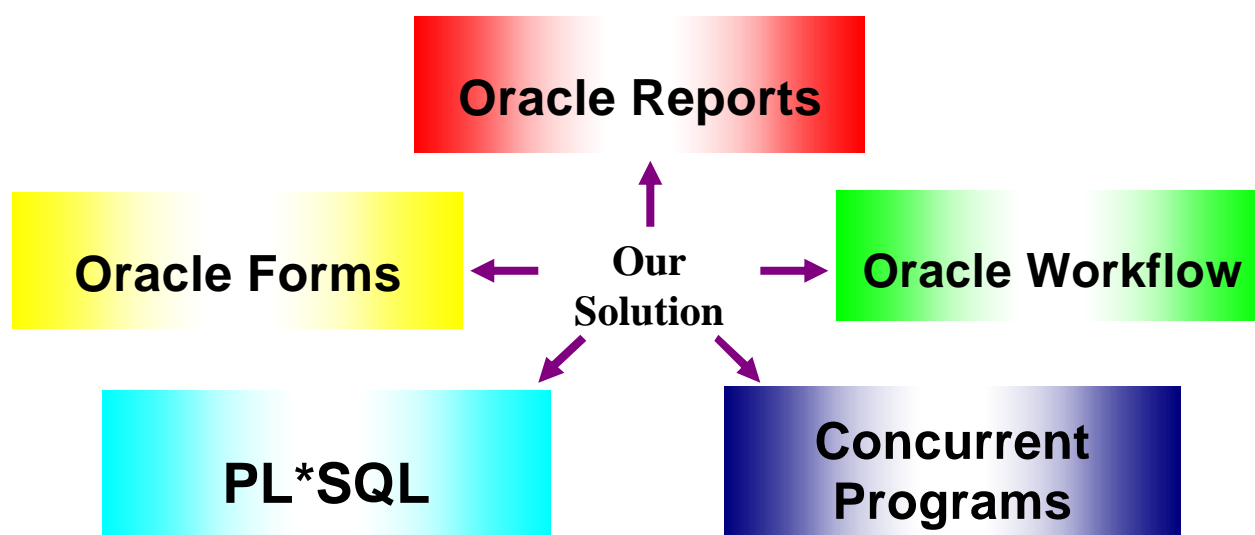
Improve Efficiencies

- Automatically allocate open credit items to open debit items.
- Apply on account and unapplied receipts to debit items automatically.
- Pay a credit refund out of Accounts Payable automatically with an integrated workflow driven approval system.

Increase Cash Flow

- Create deferred payment arrangement agreements with customer/accounts in order to facilitate the payment of outstanding debt on an instalment basis.
- Apply late payment penalties based on flexible, transactionally based business rules – not customer categories.
- Transfer outstanding debt at the level of the transaction to external debt collection agencies, or internal collectors based on parameter driven business rules.

We have used the following toolset to create this custom solution:



This presentation will explain in more detail the design of the above solutions and how they can benefit you in your organisation. We hope you will read on...

Move Staff Out of the Back Office and into the Front

To help us better collect our outstanding debt, we need our staff to focus more on 'front office' activities such as managing the relationship with our customers, by spending less time on 'back office' activities. Back office activities include tasks such as the allocation of credits and receipts to individual transactions, and issuing refunds to customers who have a credit balance. The following three solutions demonstrate how we have automated all three of these tasks by taking advantage of the latest Oracle Receivables integration points.

Automatic Credit Allocation

Background

Oracle Receivables allows us to enter a credit memo 'on-account', or applied against individual transactions. Often some research is required into a customer's account history to determine which transactions should be credited before the credit memo can be created, so it is simply entered on-account and forgotten about. Unfortunately, in this case transactions are never closed, but linger with an outstanding balance. In addition, Autoinvoice (the transaction interface program) does not have the ability to apply a credit to multiple invoices, so credits must be created on-account and manually updated with the correct applications.

Solution

The Credit Allocation extension allows you to automatically allocate on-account credit memos to open debit items based on business rules according to transaction type. The extension takes advantage of standard Oracle PL/SQL packages to create the allocations. The process can be run through the concurrent manager and it produces a report showing the matches that have been performed for you to review.

Major features include:

- Automatic or selected processing
- Rules based matching
- Application of a single credit memo to multiple debit items

The solution for the Credit Allocation makes use of both PL/SQL and a Reports 6i Report. The solution involves:

Oracle Tools	Description of Use
Concurrent Program	Automatically locate and allocate open credit items
Reports	Show allocations made

Automatic or Selected Processing

The process can run automatically as a scheduled process, without user intervention (except for optionally viewing the report). Users can also manually run the process. In either case there is no need for parameters because the concurrent request retrieves eligible transactions based on the business rules.

Rules Based Matching Criteria

The matching between the credit transactions and debit transactions is based on a hierarchy of matching rules:

- where there is an open debit memo/invoice with the same transaction type and balance amount as the credit transaction
- where there is an open debit memo/invoice with the same transaction type as the credit transaction
- where there is an open debit memo/invoice with the same balance amount as the credit transaction
- match the transaction to an open debit memo/invoice with the earliest due date

The process is repeated until the credit transaction is fully matched, or until there are no debit transactions remaining to match the credit transaction to.

Multiple Credit Transactions for a Customer

The process can match a single credit to multiple debits based on the rules above. In some instances there may be more than one credit transaction being processed for the same receivable account within the same credit allocation run. In this case there is no attempt to rank these transactions before performing the matching algorithm. All transactions are processed independently according to the matching rules. Subsequent credit transactions for the same account ensure they do not over allocate against the same invoices/debit memos allocated by previous transaction(s).

Automatic Receipt Application

Background

To keep receipt processing costs down, there is a preference to accept electronic payments from customers such as direct debit and automatic payments, although we still need to accept more traditional means of payment such as cheques. Direct debit gives us the most control but many customers don't like giving it to us. Automatic payment gives us some certainty, but if the AP form is not filled out correctly with our transaction reference then automatic receipt creation and application becomes impossible. Cheques are fine if they are sent to us with the correct remittance advice, but if not, receipts must be manually applied.

Solution

The Receipt Application extension allows you to automatically allocate unapplied, and on account receipts to open debit items based on business rules. The process can be run through the concurrent manager and it produces a report showing the matches that have been performed for you to review.

Major features include:

- Automatic or selected processing
- Rules based matching
- Application of a single receipt to multiple debit items
- Treatment of Unallocated Balances

The solution for the Receipt Application modification makes use of both PL/SQL and a Reports 6i report. The solution involves:

Oracle Tools	Description of Use
Concurrent Program	Automatically locate and allocate receipts using the Receipt API's
Reports	Show allocations made

Automatic or Selected Processing

The process can run automatically as a scheduled process, without user intervention (except for optionally viewing the report). Users can also manually run the process. In either case there is no need for parameters because the concurrent request retrieves eligible transactions based on the business rules.

Rules Based Matching Criteria

The matching between the receipts and debit transactions is based on a hierarchy of matching rules:

- where there is an open debit memo/invoice with the same balance amount as the receipt transaction
- match the receipt to an open debit memo/invoice with the earliest due date

The process is repeated until the receipt is fully matched, or until there are no debit transactions remaining to match the receipt to.

Multiple Receipts for a Customer

The process can match a single receipt to multiple debits based on the rules above. In some cases there may be more than one receipt being processed for the same receivable account within the same application run. In this case there is no attempt to rank these transactions before performing the matching algorithm. All transactions will be processed independently according to the matching rules.

Treatment of Unallocated Balances

At the end of this process there may still be some open credit transactions that have either not been allocated, or have only been partially allocated. These transactions will only occur where the customer has overpaid their account. The Credit Refunds process (detailed below) will identify these transactions, unless they represent a small balance, which may then be written off through a separate process.

Credit Refunds

Background

Refunding a customer in Oracle is a labour intensive manual process. Firstly, customer accounts must be researched to determine if a refund should be paid. For each account in credit, offset debit accounting entries must be created using a general ledger clearing account, and matched to the credit. A supplier then needs to be created with the same details as the customer. A payables invoice must then be created for the amount of the refund (offsetting the clearing account), and finally, the payment must be made.

Solution

The Credit Refund extension allows you to automatically or manually select customers that are potentially due a refund using defined selection criteria, and then automates the approval and payment procedures.

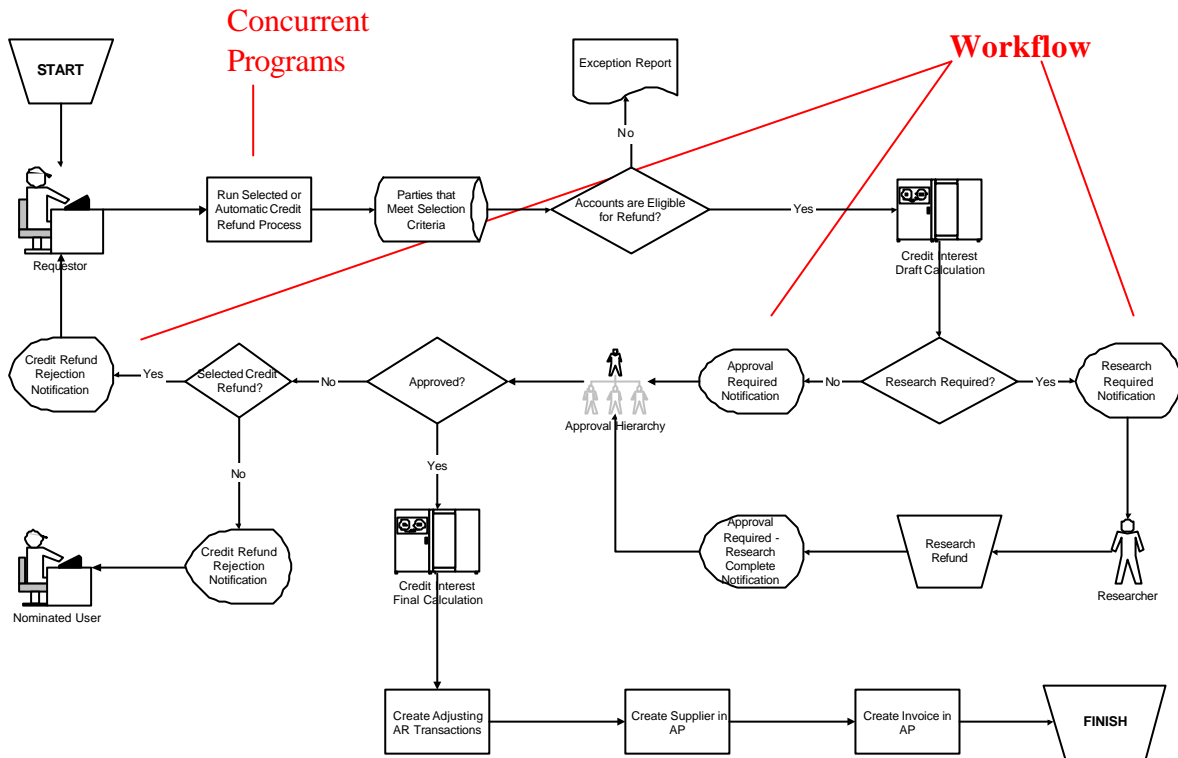
Major features include:

- Select potential refunds based on the overall account balance for a customer, taking into account different accounts belonging to the same Party, and any other accounts where a relationship has been established.
- Obtain the appropriate approvals required for a refund decision to be processed via multiple approval hierarchies.
- Refund a customer's account on request.
- Allow for credit interest to be applied where it is due.
- Automate the accounting transactions required to enable a payment to be recorded.

To develop the solution we have used Oracle Workflow, PL/SQL Programs and a Reports 6i report. Details of the solution are listed below.

Oracle Tools	Description of Use
Workflow	Approvals, Research and Credit Refunds
Reports	Credit Refund Exclusions
Concurrent Programs	Credit Refund

The following diagram shows the Credit Refund Process



Selection of Potential Refunds

Customers with accounts in credit are automatically selected based on the overall account balance for a customer, taking into account different accounts belonging to the same Party, and any other accounts where a relationship has been established. The process also checks to ensure that the account does not have a refund hold (stored in a descriptive flexfield), has not had any debt written off, and does not have an active collector on any transaction.

Approval Hierarchy

Once a refund has been identified, a notification is routed to an appropriate approver, or sent to a staff member to research, before submitting for approval. The notifications contain all account transactional information required to make an informed decision. The extension allows for multiple hierarchies to be used based on the type of debt involved.

Refund on Request

If a customer calls and requests an immediate refund, a user can submit the credit refund concurrent program for a single party. Provided that approval and payment processes are run daily, the customer should be refunded on the same business day as the request.

Refund Credit Interest

If you pay interest on credit amounts held on account over a specified amount of time, the process will call a calculation routine to ascertain the amount due, and add it to the refund.

Automated Accounting Entries

Once a refund has been approved, the process calls standard Oracle Adjustment API's to create offset accounting entries, and creates a supplier and approved invoice in Accounts Payable ready for payment.

Go Get That Money

Once you have achieved operational efficiencies in the 'back office', staff can focus their attention on collecting overdue debt from your customers through direct contact. The extensions outlined below will directly assist this collection process as they offer your customers flexible repayment plans, provide the ability to flexibly apply penalties and administration charges, and transfer collection responsibility within your organisation, and to external debt collectors.

Time Payment Arrangements – 'We're Here to Help You!'

Background

One of the key reasons for late payment of invoices is that the amount of the debt exceeds the customer's ability to pay the lump sum by the due date. Rather than wait for the customer to 'save' and pay the debt after the normal payment terms have probably expired, it is far better for your cash flow to put the customer onto an instalment plan to pay the debt over time. It is also far better for the relationship with the customer, as you are then seen as being flexible and helpful.

Standard Oracle Receivables functionality limits us in the following ways:

- Once an invoice has been transferred to the General Ledger, the payment terms cannot be updated. Updating a payment term would therefore require the original invoice to be credited, and a new invoice created with new terms – not an option where you need to keep invoice numbering synchronised with external systems, and it can create confusion for a customer about what they are paying and when.
- Each payment term needs to be defined in setup – this prevents a collector easily tailoring an instalment plan to a customer's individual needs.
- Invoices are restricted to a one-to-one relationship with a payment schedule. A customer with multiple invoices can be presented with multiple dates upon which to pay each individual transaction instalment. Each time a new invoice is created for the same customer, it must be defined with its own payment terms.

- If a customer makes a non standard payment for an invoice on an instalment plan, they are likely to make mistakes with future payments. It also creates extra work in the Accounts Receivable department determining which instalments to apply receipts to, and additional cost in recalculating instalments and then notifying the customer.

Solution

The Time Payment Arrangement (TPA) modification allows a security enabled user to create, maintain or cancel deferred payment arrangements for a customer, on an instalment basis.

By using the Oracle development tools we have been able to create custom forms and reports for this solution as well as Oracle Workflow and concurrent programs.

Major features include:



- A single TPA can include multiple invoices.
- A TPA can be updated at any time manually, or can be automatically remodelled to include new debt, or in response to unexpected credits or payments.
- The generation of correspondence notifying a customer of their payment schedule is fully automatic, including when their plan is remodelled.
- Automatic notifications are routed to the collector who created the TPA for follow up, when more than one instalment has been missed.
- Full integration with standard Oracle direct debit processes, but flexible enough to cater for all other forms of payment.

Details of the solution are listed below:

Oracle Tools	Description of Use
Forms	Time Payment Arrangement
Workflow	Payment 'dishonour' notification
Reports	<ul style="list-style-type: none"> • Instalment schedule and bank account confirmation letter • Instalment variation schedule • Payment monitoring program • Direct debit file for banking software
Programs	Payment Calculation

Time Payment Arrangement Form

In support of the core time payment arrangement functionality a new time payment arrangement form was created. This form facilitates the creation and maintenance of new and existing time payment arrangements and includes the ability for a user to view all current (active and quote) and historic (complete and defaulted) time payment arrangements.

The screenshot displays the 'Time Payment Arrangement' application window. It features a table titled 'Current Arrangements' with the following data:

Plan	Type	Status	Acct Collector	Start Date	End Date	Initial Total
10000000	Instalment	Active	Jamie	28-NOV-2002	28-AUG-2003	6,511.90

Below the table are two buttons: 'Create New' and 'Maintain'. The 'Details' tab is selected, showing the following information:

Plan Type	Instalment Plan	Bank Account	921.112.00012
Status	Active	First Interest Date	28-DEC-2002
Plan Total	6,511.90	Payment Frequency	Monthly
Prior Payments		Lesser Penalty Rate	5.030
Remaining Balance		First Repayment Date	28-NOV-2002
Instalments	10	Plan Creation Date	27-FEB-2003
Payment Method	Direct Debit		

Automated Routing of 'Non Payment' Notifications

As part of the ongoing maintenance of time payment arrangements, either the assigned account level collector, or the default collector will receive notifications via Oracle Workflow when:

- an anticipated schedule payment for a time payment arrangement is missed, and
- direct debit payments are dishonoured.

Automated Time Payment Arrangement Maintenance

As a result of payment matching and credit allocation processes within Oracle, transactions covered by an active instalment plan may have an unexpected credit(s) allocated to a transaction(s). This unexpected credit allocation effectively means that the customer's agreed-upon instalment schedule is no longer valid therefore the customer must be advised of the change in the repayment schedule. Instalment plan correspondence is then issued to the customer in support of the new agreement.

Automated Completing of a Time Payment Arrangement

As part of a regular scheduled process, when the time payment arrangement has been fully paid the application will change the state of a time payment arrangement to recognise that it is now completed.

Issuance of Time Payment Arrangement Supporting Correspondence

Where time payment arrangements are created as part of the manual or automated process, the system can trigger the generation of the supporting collateral via a file to an external mail-house.

Penalties – ‘The Big Stick’

Background

If our customers don't pay their bills, even after we've offered them flexible payment terms, then it may be time to impose penalties for late payments. Sometimes just the threat of penalties can be enough to induce a customer to pay their debt on time.

Standard Oracle Receivables functionality limits us in the following ways:

- Finance charge functionality is based on a customer or site, not at a transactional level.
- If you decide not to levy a penalty at any time after reaching an agreement with a customer, but choose to resume later, the process 'catches up' even if you don't want it to.
- You have to maintain the rate of the charge at the customer or site level – that rate applies to all of a customer's transactions.
- There is no facility to pass on any of the administration costs associated with chasing overdue debt.
- Reversing finance charges is a very time consuming, manual process.

Solution

The Penalties application extension determines applicable penalties on overdue debt based on transaction types, allowing a user to review and approve the results before actual creation. It also provides a streamlined mechanism for reversing a penalty once it has been created.

By using the Oracle development tools we have been able to create a custom form and report for this solution to work in conjunction with a custom concurrent program.

Major features include:

- Automatic selection of transactions eligible for a penalty to be imposed.
- User definable minimum penalty amount.
- An online review and update of penalties before creation.
- Variable penalty rates and administration fees can be imposed based on transaction type.
- Automatic creation of penalties using Oracle's Adjustment API's.
- Penalties can be automatically reversed with the selection of a custom option in the 'Tools' menu.

Details of the solution are listed below:

Oracle Tools	Description of Use
Forms	Review & Create Penalties
Reports	Review Penalties
Programs	Select Penalties
Programs	Select Penalties
Custom Library	Custom Menu Option to Reverse Penalties

Identification of Debt to Penalise

An Oracle concurrent process is the basis for this application extension. It allows the user to select transactions available for penalties based on entered search criteria. This process triggers the creation of a report. The results of this report are stored in the database for further review via a custom form.

Online Review of Proposed Penalties

The 'Penalties' form allows the online review of those transactions that have been identified by the initial process. By default, all transactions that have been identified are selected indicating that penalties should be created for these transactions.

The screenshot shows a web-based form titled "Penalty Batch". It features a table with the following columns: "Apply", "Customer Name", "Doc Number", "Transaction Type", "Due Date", "Balance Due", "Current Penalty", and "Liable Penalty". The "Apply" column contains a series of checkboxes, with the top-most checkbox selected. The table is currently empty of data. At the bottom right of the form, there are two buttons: "Submit" and "Cancel Batch".

User Definable Penalty Calculations

The variables for penalty imposition are definable for a particular transaction type. Each transaction type is associated to a penalty regime (if applicable). Each penalty regime can have a distinct penalty percentage rate and/or administration amount.

Automatic Creation of Penalty Transactions

From the 'Penalties' form, the user can approve the mass creation of penalty transactions (adjustments). They can also alter the value of the penalty from the form if required. These penalties are created via the Adjustment API upon clicking the submit button.

Automatically Create a Penalty Reversal

To simplify the penalty reversal process for end users, a simple penalty reversal function is available for end users to use. From the 'Adjustments' form, the user can select a specific penalty that they wish to reverse. By selecting a menu option, an offsetting transaction will be created effectively reversing the penalty from the customer's transaction.

Debt Transfer – 'Pay Up or We'll ...'

Background

If being flexible about payment terms, and applying penalties for overdue debt does not persuade your customer to pay their debt, it becomes necessary to manage transactions and customers on an individual basis. Your organisation may have a collections team to manage high value or sensitive debt, but it often doesn't make financial sense to have your own people manage high volume, low value transactions – this is where Debt Collection Agencies (DCA) can be useful. But to manage our debt effectively we need the right functionality.

Standard Oracle Receivables has no facility that allows us to transfer and manage debt at the level of the transaction, and no means of interacting with third party debt collection agencies.

Solution

The Manage Debt Transfer modification allows you to transfer individual transactional debt with no allocated collector to either an internal or an external collector.

Major features include:

- Automatic or selected processing to select transactions for the assignment of an internal or external collector based on pre-defined business rules.
- Proposed collector assignments can be reviewed in a report and updated individually via a form prior to the actual assignment becoming active.
- Transactions can be individually selected for re-assignment to a new collector.
- Electronic files are produced for DCAs to advise them of new transactions to manage, updates to debt already under management, and transactions that are being transferred to another DCA.
- Automatic creation of follow up tasks for each internal collector assigned new debt to investigate.

Details of the solution are listed below:

Oracle Tools	Description of Use
Descriptive Flexfields	<ul style="list-style-type: none"> • Name of Collector • Mark transaction for transfer
Profile Options	To determine which collector should be allocated to specific debt based on alpha split.
Forms	Manage Debt Transfer – review and update collector assignments
Reports	<ul style="list-style-type: none"> • Changes in debts allocated • Payments received • Historical changes in collectors allocated to a transaction
PL/SQL Programs	<ul style="list-style-type: none"> • Determine the appropriate collector for allocation of debt • Create customer call and follow up task for collector

Automatic or Selected Processing

The process can be run with either pre-defined default parameters or with user-specified parameters. The results of the process will not be updated against the database until confirmed by the user.

Online Review of Proposed Assignments

When the process has completed, the selected transactions and the proposed new collectors can be viewed online. The results can be adjusted, as required. Once the user is satisfied that the transactions / collectors are correct, the transfer will be confirmed and the transactions updated accordingly.

Manual Request for Reassignment

An internal collector who has been validly assigned to a transaction can manually mark that transaction for transfer to a DCA if they no longer wish to manage the debt internally. The Manage Debt Transfer process will select the appropriate DCA for the transaction using pre-defined business rules, based on customer account type.

Electronic File Transfer to DCA

As a result of each Debt Transfer process a file is produced showing changes in debts managed by the DCAs. This report is forwarded to the DCAs in an electronic file format. The DCAs will also receive a file listing payments received/ adjustments processed for debts managed by them.

Automatic Creation of Follow Up Task

An automatic follow up task will be created for internal collectors when they have been assigned a new item of debt as a result of this process. This task appears in the standard AR Scheduler window.

Conclusion

This paper has covered three key ways that you can extend Oracle Receivables (automatic credit and receipt allocation, and automatic refunds) to drive operational efficiencies into your receivables processing, thus freeing up time for people who can focus on collecting delinquent debt. It has also presented another three key methods to support your collection processes by allowing you to provide your customers with flexible repayment terms (Time Payment Arrangements), penalise late payers (Penalties) and transfer debt collection responsibility to internal and external parties.

If you would like to discuss this further please feel free to contact me by phone on 025 899 412 or e-mail Greg.Fookes@cgnz.co.