

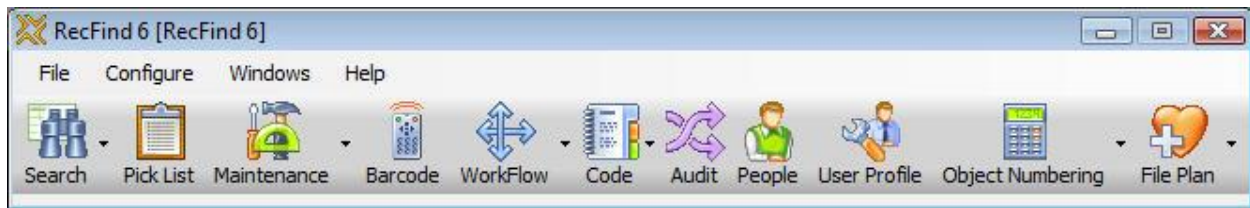


Using RecFind 6 to implement HR Functionality

Table of Contents

| | |
|---|-----------|
| <i>What does a Human Resources (HR) system do?</i> | <i>3</i> |
| Payroll | 3 |
| <i>What already exists in RecFind 6?</i> | <i>4</i> |
| Tables you will probably use | 4 |
| How do you look at these tables? | 4 |
| Can you modify these tables? | 7 |
| How many tables do you have to work with on a daily basis? | 7 |
| <i>Reporting</i> | <i>7</i> |
| <i>Searching.....</i> | <i>7</i> |
| <i>What 'commands' or 'methods' do you have when managing employee records?</i> | <i>7</i> |
| <i>Importing Information</i> | <i>9</i> |
| <i>Exporting information</i> | <i>9</i> |
| <i>Adding the Leave Form.....</i> | <i>9</i> |
| <i>Adding the Leave Form Logic to RecFind 6</i> | <i>10</i> |
| Trigger..... | 10 |
| Scheduled Task/Stored Procedure | 11 |

| | |
|--|----|
| Using two Triggers for the Annual Leave form | 11 |
| <i>To Run your HR Application</i> | 11 |
| Reporting..... | 15 |
| <i>Step by Step</i> | 18 |
| DRM..... | 18 |
| RecFind 6..... | 18 |
| <i>Appendix A</i> | 19 |
| Sample Annual Leave form Triggers | 19 |



What does a Human Resources (HR) system do?

AN HR system allows an organization to enter, access and update all the information required to maintain an up-to-date and searchable database of employee information including name, home address, resume, test results, qualifications, salary reviews, performance, etc.

It should also facilitate the normal business processes involved in HR including the ability to search the employee database based on a set of criterion such as age, experience, marital status, qualifications, experience, availability for transfer and job reviews, etc.

For example, if an organization has a requirement for an experienced project manager for a mining consulting job in Mozambique for six months but living out of a male-only mining compound it would want to search its employee database for a male employee who has the prerequisite mining degree and experience and who was prepared to travel on extended assignments and was available during the projected timeframe.

Similarly, the HR system should allow the organization to implement an employee review system based on a selection of criteria, (e.g., next review date), such that the HR manager would be automatically reminded of each employee's coming review in a stated time, for example, one month prior to the next review date.

AN HR system should also include a host of 'normal' business processes such as allowing the employee to request leave and then automatically notifying a manager so the leave request can be both considered and approved.

For most HR managers, the four major system concerns would be that the HR system facilitates:

1. Hiring and selection (recruitment)
2. Performance management
3. Leave management
4. Learning and development

These are of course just the major objectives; an HR system would also be used to facilitate mentoring, coaching, job rotation, succession planning, fast track selection and a host of other HR related activities common to any organization.

Payroll

HR systems may also include payroll components. We would never recommend trying to implement a payroll component with RecFind 6. This is because payroll is a specialized application that requires constant updating by specialists with new rate and awards information. Payroll systems should be left to payroll experts. Making mistakes with employees' pays is something you really, really want to avoid. Please link (integrate) to your payroll application, do not try to replace it with RecFind 6.

What already exists in RecFind 6?

When we configured the data model and business processes of RecFind 6 we assumed customers would want to use it as an HR solution so we configured in most of the tables, fields and business processes you would need.

This means that whenever you think you need to create a new table or a new field or a new process please first check to make sure that it doesn't already exist.

Tables you will probably use

The pre-configured tables we thought you may want to use for HR management are as follows:

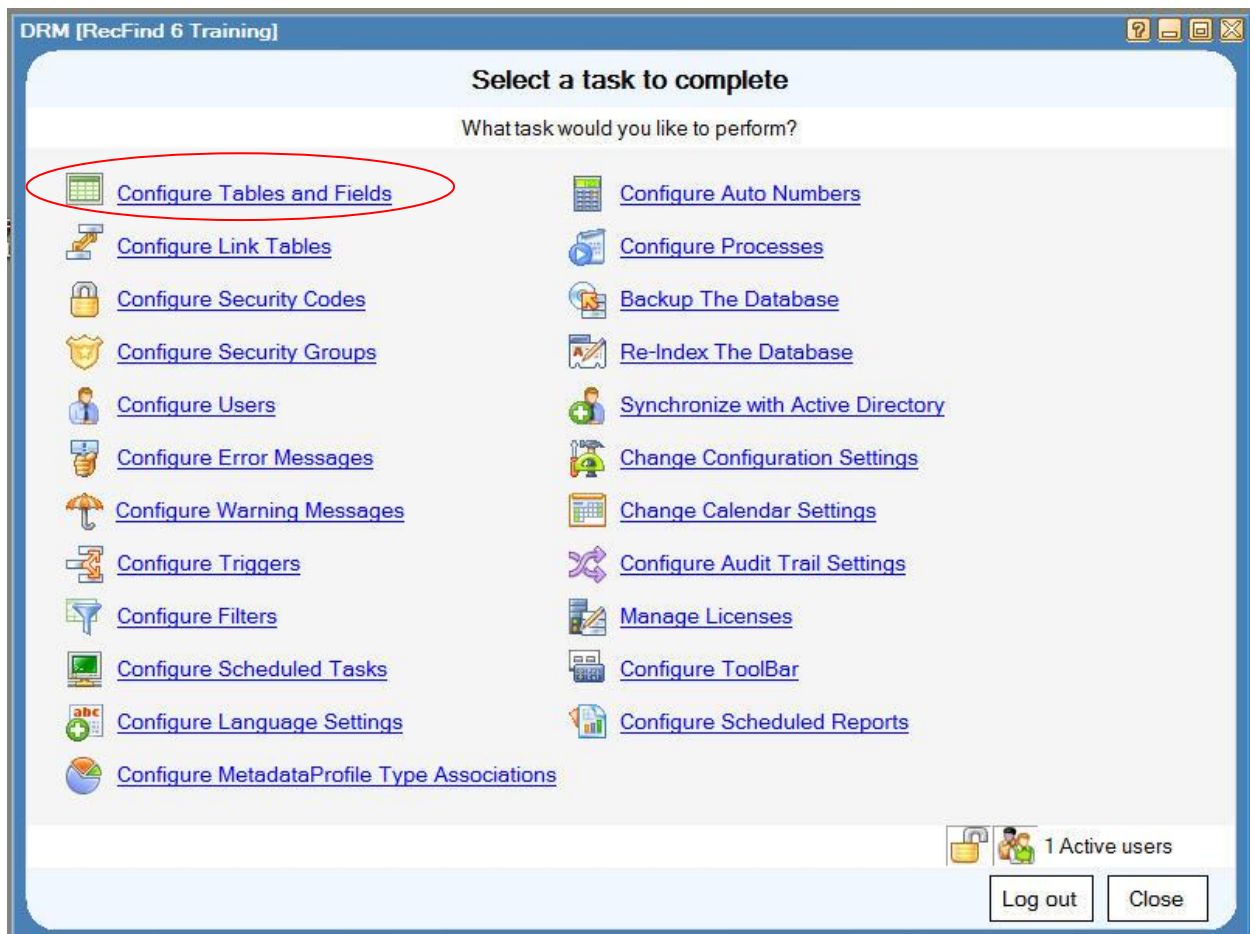
1. *Person
2. *User Profile
3. EDOC (this is where all electronic documents are kept including scanned pages, Word documents, pictures, videos, test results, etc. This is how you easily capture and link them to an employee record)
4. Benefit
5. Location
6. City
7. State
8. Country
9. Entity
10. Department Division
11. Citizenship
12. Holiday
13. Job History
14. Job History Benefit
15. Job Rating
16. Job Title
17. Language
18. Qualification
19. Reference
20. Resume
21. Resume Test History
22. Why Left

The two most important tables are the Person and User Profile tables. The Person table contains details of any person you deal with including employees, partners, suppliers, customers, etc. If a 'person' is an employee then the Person record will link to a User Profile record. Every employee will have a linked User Profile record. That is, the Person table contains information common to any person; the User Profile table contains information common to any employee.

All of the other HR tables will already be linked to either the Person or User Profile table.

How do you look at these tables?

You can run the DRM and select the first menu 'Configure Tables and Fields' (the recommended way) or you can view them in the RecFind 6 client by selecting 'Search any table'. You will need to be an Administrator to do either of these things.



Configure Table Schema

Configure Tables

| TableName | ClassName |
|-------------------|------------------|
| JobHistoryBenefit | Application |
| JobRating | Application |
| JobTitle | Application |
| K1Configuration | System Locked |
| K1StartPoint | System Essential |
| Language | System Essential |
| LanguageString | System Locked |
| Leave_Form | Application |
| LeaveType | Application |
| LineItem | Application |
| ListColumn | System Locked |
| Location | System Essential |
| MailList | System Essential |
| MarketSector | Application |
| MediaType | Application |
| MetadataProfile | System Essential |
| Method | System Locked |
| MethodOfPayment | Application |
| Movement | System Essential |
| Notes | Application |

Return

Configure Fields

Configure Fields

Leave_Form Type: [Default] ☒ Visible Only

| Caption | Name | DataType | Essential | Locked |
|--------------|----------------|----------|-----------|--------|
| ExternalID | ExternalID | NVarChar | Yes | No |
| TypeID | TypeID | Int | Yes | No |
| Employee | Employee | Int | No | No |
| Manager | Manager | Int | No | No |
| DateFrom | DateFrom | DateTime | No | No |
| DateTo | DateTo | DateTime | No | No |
| NumberOfDays | Number_of_Days | Decimal | No | No |
| Leave Type | LeaveType | Int | No | No |
| Comments | Comments | NVarChar | No | No |
| Approved | Approved | Bit | No | No |
| SecurityID | SecurityID | Int | Yes | No |

Note that you do not need to be an Administrator to actually use these tables inside RecFind 6 (i.e., when logged on as a normal user). You just need to have them added to your Security Group by your RecFind 6 administrator. You do need to be an administrator to modify (e.g., add fields) or add any new tables.

Can you modify these tables?

You can modify any application table in RecFind6 and you can create new tables, if required, using the DRM. You can add new fields, you can change the field order, you can change field captions, you can change the headings when the table is displayed in RecFind 6 and you can add new links to other tables, etc, etc. You can do all of this without programming using the friendly interface of the DRM as long as you are an administrator in the RecFind 6 security system.

How many tables do you have to work with on a daily basis?

The design of RecFind 6 is what we call single-table-centric. This means most applications like HR have a 'main' table you work with and all other related tables are linked to this table. This means that you spend most of your time working with just a single table and most things you need to do can be done from this single table. For the HR application in RecFind 6, the main table for you to work with is the User Profile table. Because the RecFind 6 data model is a relational design, this means all 'related' information will be linked to this main table making it easy for you to add and modify all the information you need to for each employee.

Reporting

RecFind 6 includes a powerful report writer based on the Active Reports engine. You can make use of standard reports, you can modify and then save (as new) standard reports or you can create reports from scratch. You can create any report or form you require for your HR application.

Searching

RecFind 6 includes a powerful searching module that automatically indexes all text for both Metadata and the full text of any captured electronic documents like emails, PDF, Word, etc. You can combine Metadata and full text searches and you can use BOOLEAN operators and ranges (e.g., date ranges). You can search on any field and any combination of fields.











What 'commands' or 'methods' do you have when managing employee records?

This is configurable by you when setting up your application in the RecFind 6 DRM. You decide what tables and fields your staff see and you decide what they can do with this information. Everything is linked to Security Groups.

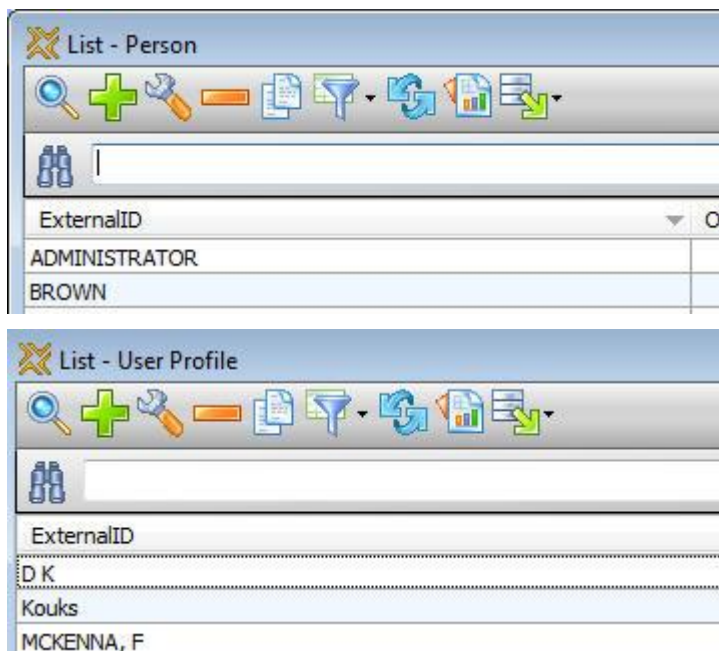
You decide for each Security Group just what the people in that group can see. That is, do you want all staff to see all the details of every employee including salary and the last review or do you wish to control access? In the normal world you would want to control access down to the field level; you would want to have several different Security Groups and restrict access to some information for some groups. As the HR manager you would give yourself access to all information but you would restrict most others from being able to see salaries and reviews because these are confidential.

Similarly, you would give yourself the right to add and modify and delete employee records but you would restrict these rights (particularly the delete method) for most other employees.

The list of available methods for the Person and UserProfile tables are:

-  ADD
-  MODIFY
-  DELETE
-  CLONE
-  EXPORT
-  SORT
-  PRINT
-  SELECT
-  VIEW (same as double click)
-  PROCESS (this is a clever way to run an external program from within RecFind 6. For example, you could use it to run a special label printing program to print employee badges from the records you have selected)

When you select methods they appear above your working list screen in RecFind 6. Though you need to remember that methods are configurable per Security Group and that the methods you see depend upon how the administrator has configured your Security Group.



Importing Information

RecFind 6 comes with a powerful tool called Xchange. This can be used to capture information from almost any source (e.g., Oracle, SQL Server, Access, DB/2, Excel, CSV, etc) and populate RecFind 6 tables. It can also be used to export any information in industry standard XML format.

You could use Xchange for instance to import information from your payroll system.

You can also import most employee data into RecFind 6 using the DRM feature called Active Directory Integration. This is probably the quickest and easiest way to import all the basic employee information you require from where it is already entered in your network security system. There is no need to re-key this information.

Exporting information

There is a standard Export command in RecFind 6 that allows you to select any employee data and then export it in either XML or CSV format (i.e., in a format that any other application can easily read). You therefore have two ways to easily export information, one using Xchange and another using the Export method within RecFind 6.

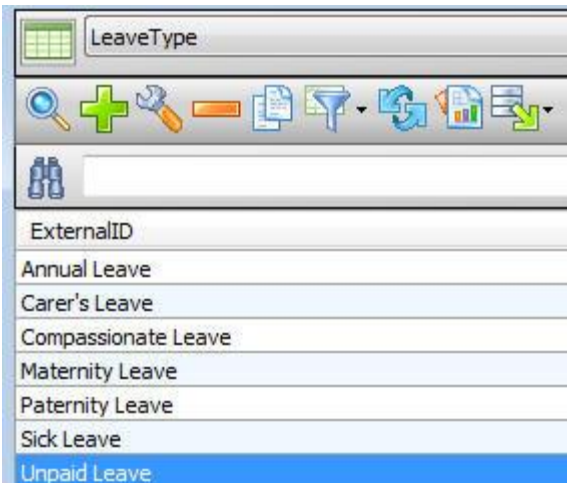
Adding the Leave Form

I have configured a simple HR application using RecFind 6 and the DRM as an example of how it can be done. About the only tables I have had to add (using the DRM) are the Leave Form and Leave Type tables as follows:

Leave Record

In the Leave Type table (below) I added a number of records to accommodate the types of leave we allow. The most appropriate is selected when the leave form is filled out. We also make this a mandatory field so it has to be filled out for each leave application.

Leave Type table, typical values



| ExternalID |
|---------------------|
| Annual Leave |
| Carer's Leave |
| Compassionate Leave |
| Maternity Leave |
| Paternity Leave |
| Sick Leave |
| Unpaid Leave |

I link the Leave Form table to the UserProfile record so I can create or view all leave applications from this single table. I can also provide access to this table and function in RecFind 6 for employees so they can add their own leave requests. As soon as the leave request is processed the manager is immediately notified by email so he/she can approve or disprove the leave request.

I 'automate' some of the business processes by using the DRM again and by adding a couple of 'Triggers'. Now it is probably appropriate to walk slowly through this Leave Form process as it is a good example of how you can make any changes to RecFind 6.

Adding the Leave Form Logic to RecFind 6

There are a number of ways to add logic and business processes to RecFind 6 but the most powerful are the Trigger and Scheduled Task options in the DRM. These are also the only places in the DRM where you will be expected to actually 'code' something. However, the good news is we aren't talking about a whole program and tens of thousands of lines of code; we are talking about a few lines or maybe a page of relatively 'simple' code.

RecFind 6 is a very unusual system in that it 'exposes' this application code (in the DRM) so you can view it, change it, copy it and re-use it or just add new code. And this is the really good news; using these two features you can add logic to perform absolutely any business process you can think of. There isn't any process that can't be added using the DRM's Triggers or Scheduled Tasks options. All you need is a little expertise (and we can certainly provide that if you can't). First however, some basic definitions:

Trigger

In simple terms a Trigger is a piece of code that executes (runs) every time a record from a particular table (e.g., the Leave Form table) is added (inserted), modified (updated) or deleted. You determine which action(s) 'trigger' the code to run, hence the name. For example, "Please run this piece of code every time a new Leave Form record is created (added) by an employee" and "Please run this piece of code every time a Leave Form record is updated by the manager".

This method is ideal for performing any business process that is event driven, i.e., every time an employee makes an annual leave request.

Scheduled Task/Stored Procedure

The DRM Scheduled Task function runs exactly the same kind of code as the Triggers function but it runs it under different conditions to do different things. It runs code on a timed basis; every second, every minute, every hour, every day, once a month, once a year, etc. You choose the table and the time interval.

This method is ideal for performing any business process that needs to happen as a timed event; for example, recalculating the value of all assets at the end of each financial year.

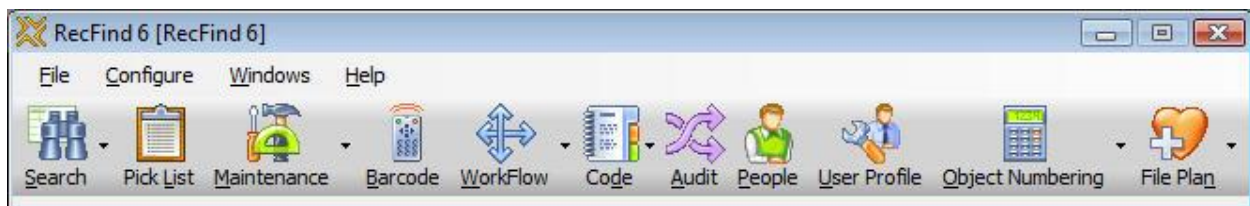
Using two Triggers for the Annual Leave form

For our Annual Leave form for we use two Triggers (see sample code as Appendix A).

- The first Trigger (on insert) generates an email to the employee's manager when the employee submits a new request for leave asking the manager to review and approve the request.
- The second Trigger (on update) checks to see if the approved flag has been set to yes and if so, sends an email to both the employee and accounts.

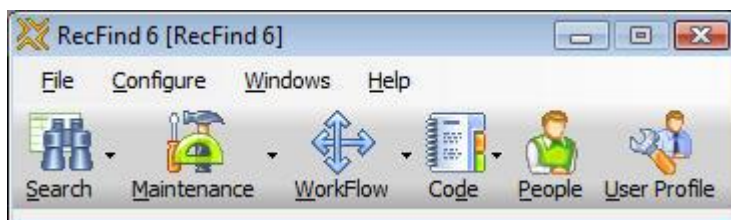
To Run your HR Application

The Security Group determines what you see when you run RecFind 6. So for example, if I log on as the RecFind 6 administrator I see the following RecFind 6 toolbar:



I now have access to all tables, all methods and all information stored in RecFind 6 regardless of which application it is because my Security Group allows me this level of access.

But, if I log on as an HR user I see a very different picture:



I now only have access to the main functions, sub menus, tables and information relevant to the HR application. And, this would be an HR manager's view. I would restrict this view even further for people below the rank of HR manager. Remember that all of this is controlled by the user's Security Group and that the Security Group doesn't just control what tables you can see (e.g., the Person table) it controls what data (records) you can see in each table and what fields you can see in each record.

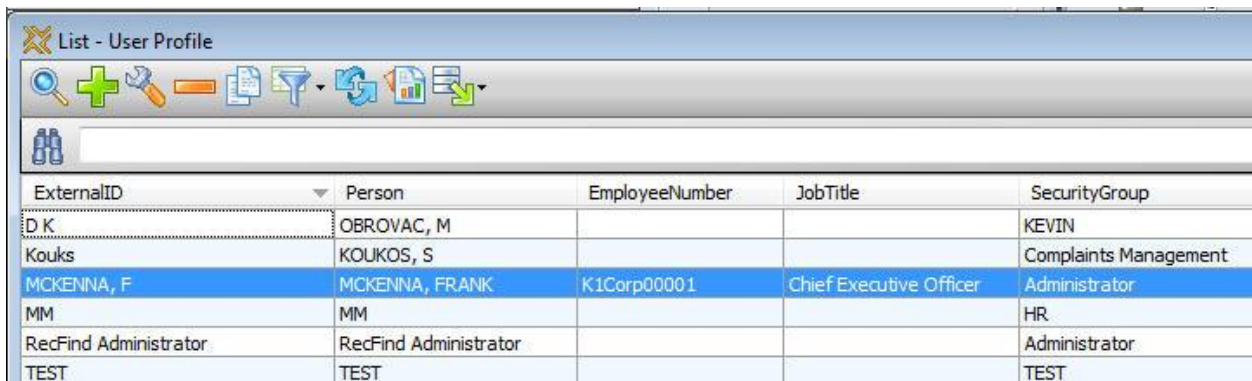
This means the manager for example can see the salary field when looking at an employee record but a clerk working in HR can't. It also means that when an HR clerk looks at the Person table he/she only sees Person records for employees, he/she does not see all the other Person records in that table for partners and suppliers.

This is the real beauty of the security system in RecFind 6; it is granular down to the exact level you require.

Now to begin processing HR transactions I click on the User Profile icon.




































Then I select the employee, e.g., me.

A screenshot of a software window titled 'List - User Profile'. The window has a toolbar with various icons including a magnifying glass, a plus sign, a wrench, a minus sign, a document, a funnel, a refresh arrow, a print icon, and a green arrow. Below the toolbar is a search bar with two small bottle icons. The main area contains a table with five columns: ExternalID, Person, EmployeeNumber, JobTitle, and SecurityGroup. The table has six rows of data, with the third row highlighted in blue.

| ExternalID | Person | EmployeeNumber | JobTitle | SecurityGroup |
|-----------------------|-----------------------|----------------|-------------------------|-----------------------|
| D K | OBROVAC, M | | | KEVIN |
| Kouks | KOUKOS, S | | | Complaints Management |
| MCKENNA, F | MCKENNA, FRANK | K1Corp00001 | Chief Executive Officer | Administrator |
| MM | MM | | | HR |
| RecFind Administrator | RecFind Administrator | | | Administrator |
| TEST | TEST | | | TEST |

Then I click on the Modify button 

This is where we clearly see how any application in RecFind 6 is what we call 'single-table-centric'.

| | |
|-----------------------|--|
| *ExternalID | MCKENNA, F |
| Type |   |
| *Person |   MCKENNA, FRANK |
| EmployeeNumber | K1Corp00001 |
| Barcode | |
| *UserID | FRANK |
| *Password | ***** |
| *SecurityGroup |   Administrator |
| *Language |   English |
| Citizenship |   |
| Qualifications |   Bachelor Commerce |
| Benefits |  |
| JobTitle |   Chief Executive Officer |
| JobRating |   3rd quartile |
| Leave_Forms |   1 related record(s) |
| JobHistories |   1 related record(s) |
| Resumes |   1 related record(s) |
| WhyLeft |   |
| Font |   Arial, White, 10, Bold |
| HeadingFont |   |
| ButtonFont |   |
| WorkingHoursStartTime |  <input checked="" type="checkbox"/> 08:00:00 AM |
| WorkingHoursEndTime |  <input checked="" type="checkbox"/> 06:00:00 PM |
| WorkingHoursPerWeek | 54.00 |

When working with the User Profile table (above), I have one click access to any other information in the database relevant to this employee. Everything is linked to the employee's User Profile record including leave forms.

For example, if I want to see Frank's resume I just click the resume link (see above).

| | | | |
|---------------------------------------|----------------------|-------------|---------------------|
| Resumes | | | 1 related record(s) |
| ExternalID | LastModifiedByPerson | UserProfile | |
| Frank McKenna Resume - Curriculum ... | MM | MCKENNA, F | |

I can now view this by double clicking it. I can also add a new resume at this point by scanning in a paper document or capturing an electronic document.

Adobe Acrobat Professional - [FMRESUME 15022007.pdf]

File Edit View Document Comments Tools Advanced Window Help

Create PDF Comment & Markup Send for Review Secure

95%

Bookmarks Signatures Pages Model Tree Attachments Comments

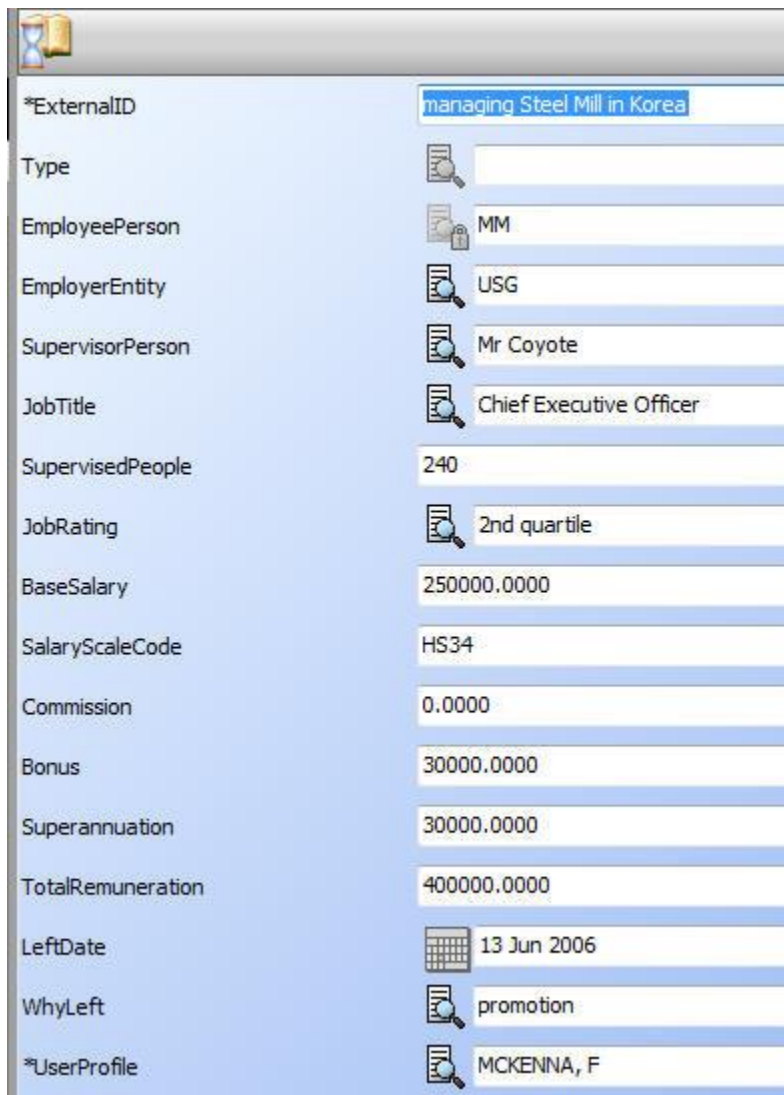
RESUME OF FRANK MCKENNA AS OF JUNE 2008 PAGE 1

Frank McKenna
f.mckenna@knowledgeonecorp.com

B I O G R A P H Y

Frank graduated from Dapto High with Honours in Economics in 1962 and then studied accounting at NSW tech and the University of NSW. Frank happily dropped accounting in 1965 after getting an A+ in the IBM programming aptitude test at Australian Iron & Steel and joining the computer division. He progressed through the traditional career path of; operator, programmer, analyst, project manager, IT director. He ended his multinational career as the Technical Director of Sperry Univac and then Burroughs in 1983. He is the founder and Managing Director of the GMB Group (now called the Knowledgeone Corporation) with offices in Sydney

I can look at Frank's complete job history (or update it) with this company by clicking on the Job History link (see below). This is where you maintain one to many records of the employee's previous jobs in your organization, it is the employment history.



| | |
|-------------------|------------------------------|
| *ExternalID | managing Steel Mill in Korea |
| Type | |
| EmployeePerson | MM |
| EmployerEntity | USG |
| SupervisorPerson | Mr Coyote |
| JobTitle | Chief Executive Officer |
| SupervisedPeople | 240 |
| JobRating | 2nd quartile |
| BaseSalary | 250000.0000 |
| SalaryScaleCode | HS34 |
| Commission | 0.0000 |
| Bonus | 30000.0000 |
| Superannuation | 30000.0000 |
| TotalRemuneration | 400000.0000 |
| LeftDate | 13 Jun 2006 |
| WhyLeft | promotion |
| *UserProfile | MCKENNA, F |

I can have as many links as I like to the User Profile table just as I can have as many business processes as I like.

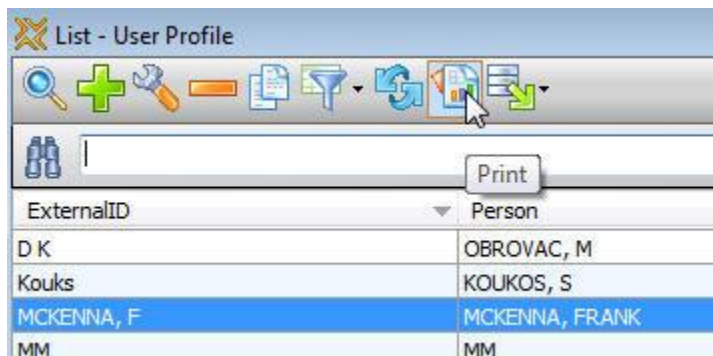
I don't have to have everything at the start; I can easily add functionality as the needs develop.

The functionality of RecFind 6 allows me to View, Add, Modify or Delete any information. Whether or not I can actually do any of these things as a user (e.g., Delete) depends upon what rights I have been assigned in my Security Group.

Reporting

I have the choice of using a standard report, of customizing a standard report (and saving it as a new report) or I can just create a new report from scratch.

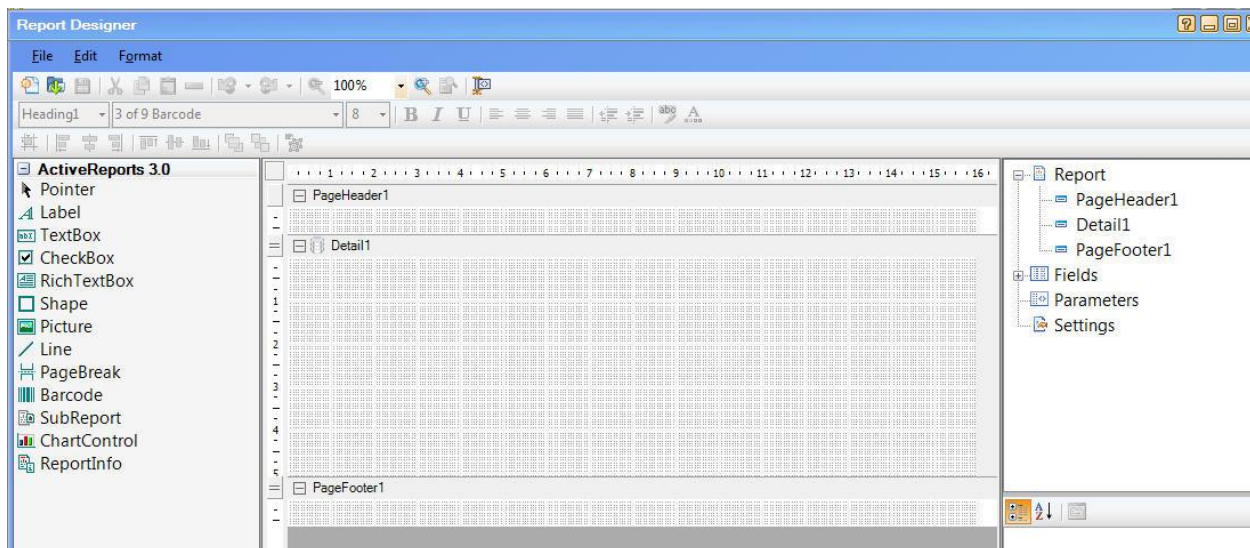
I just select the record(s) I want to produce a report for and click on the Print button as follows:



RecFind 6 then presents me with my three choices:

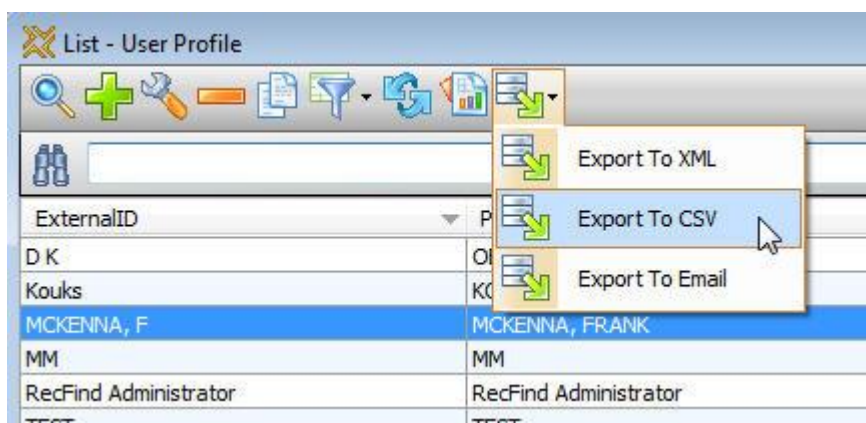


I choose (in this example) to build a new report (which I will save so I can use it again). The Report Designer appears:



This is a fully functioned report writer 'embedded' into RecFind 6 and callable whenever you click on the Print button. I can now design, test, save and run my HR report.

I could have also simply exported all the employee information in CSV format and used an external tool (e.g., Excel) to build my report. I do this by clicking on the Export button and selecting CSV as follows:



Or, I could have just emailed the information (e.g., to our US office) direct from this menu.

Step by Step

This paper can't duplicate everything you learn in a training course; rather it tries to explain how things are done in RecFind 6. It is a summary or guideline rather than a detailed instruction manual.

However, just as we have done with our other RecFind 6 papers (e.g., on asset management) we need to provide you with a step by step approach, listing all the things you need to do to achieve your HR application objectives.

DRM

1. Decide what HR processes you need to implement (e.g., Leave Forms)
2. Add a security code called 'HR' (or whatever you want to call it)
3. Add a Security Group called 'HR' (or whatever you want to call it)
 - a. Select the tables and fields, Methods and Security codes this Security Group has access to
4. Decide what other tables you will use and how you will link them to the Person or User Profile table
5. Make any changes you require to the Person table
 - a. Fields
 - b. Column headings
 - c. Field captions
 - d. Field order, etc
6. Make any changes you require to the User Profile table
7. Make any changes you require to other tables (e.g., Resume)
8. Add the Leave Type table (if you require this process)
9. Add the Leave Form table
10. Add any Filters you require for your new tables
11. Import all employee information using the Active Directory Integration feature

RecFind 6

1. Update employee information in the Person table (e.g., adding information that was not in Active Directory)
2. Update employee information in the User Profile table (e.g., assign Security Groups)
3. Design and save your HR reports
4. Design and save any complex but 'need-to-run-often' searches you require.
5. Train your staff

You now have an HR application running within RecFind 6, congratulations.

Appendix A

Sample Annual Leave form Triggers

*Note that you will need to 'adapt' this code for your use. This is actually the code from our production system that we use to manage leave requests at Knowledgeone Corporation (we call our production system K1-Live).

```
#####  
#                               #  
# trLeaveForm_Insert (AFTER / ON INSERT) #  
#                               #  
#####
```

< COPY FROM THIS LINE >

```
/* -----  
Step 1: Set security of record to manager only level  
Note: 17 applies to K1_LIVE only, update as appropriate  
----- */
```

```
declare @LF_id int  
SELECT @LF_id = [ID] from inserted  
update leaveform set securityid = 17 where [id] = @LF_id
```

```
/* -----  
Step 2: Generate e-mail and send to the email address for  
the selected manager. Note: Uses EmailNotification table  
which is processed every 5 minutes so there can be some  
delay before the e-mail is sent.  
----- */
```

```
DECLARE @toemail nvarchar(200)  
DECLARE @BodyText NVARCHAR(4000)
```

```
DECLARE @ApplicantID int  
DECLARE @ApplicantName nvarchar(200)
```

```

DECLARE @ManagerID int
DECLARE @ManagerEmail nvarchar(200)

SELECT @ApplicantID = UserProfileID from inserted

SELECT @ManagerID = ManagerID from inserted

SELECT @ApplicantName = Externalid from UserProfile where [id] = @ApplicantID
SELECT @ManagerEmail = workemail from person where [id] = @ManagerID

SELECT @BodyText = @ApplicantName + ' has entered an application for leave and it requires
your approval.'/*+CHAR(13) + CHAR(13) + 'To view record click the following link:' + CHAR(13)
+
'http://1.1.1.172/k1/ui/frmApiView.aspx?table=LeaveForm&recordid='+convert(nvarchar(10),@L
F_id)*/
INSERT INTO EmailNotification (RecipientID, Subject, BodyText)
VALUES (@ManagerID, 'Annual Leave Application (' + @ApplicantName + ')',
@BodyText)

```

< COPY TO THIS LINE >

```

#####
#                               #
# trLeaveForm_Update (AFTER / ON UPDATE) #
#                               #
#####

```

< COPY FROM THIS LINE >

```

/* -----
Checks to see if the 'Approved' flag has changed from
'No' to 'Yes', if so an e-mail is sent to applicant and
to accounts. Accounts ID is hard-coded so needs updating.
Note: Also uses EmailNotification table which is
processed every 5 minutes so there can be some delay

```


before the e-mail is sent.

----- */

DECLARE @OldApproved bit, @NewApproved bit

DECLARE @FromDate nvarchar(50), @ToDate nvarchar(50)

DECLARE @NumberOfDays nvarchar(5)

DECLARE @LeaveType nvarchar(50)

DECLARE @LeaveFormID int

DECLARE @Comments nvarchar(500)

DECLARE @ApplicantID int, @AccountsID int, @ApplicantPersonID int

DECLARE @ApplicantEmail nvarchar(200), @ApplicantName nvarchar(200)

DECLARE @BodyText nvarchar(4000)

SELECT @LeaveFormID = inserted.id, @ApplicantID = UserProfileID, @FromDate =
convert(char(11),(DateFrom),106), @ToDate = convert(char(11),(DateTo),106),
@NumberOfDays = NumberOfDays, @LeaveType = LeaveType.Externalid from inserted,
leavetype

where leavetypeid = leavetype.[id]

SELECT @ApplicantPersonID = Person.ID, @ApplicantName = Person.ExternalID,
@ApplicantEmail = WorkEmail from Person, UserProfile where Person.[ID] =
UserProfile.[PersonID] and UserProfile.[ID] = @ApplicantID

SELECT @OldApproved = Approved from Deleted

SELECT @NewApproved = Approved from Inserted

SELECT @Comments = Reason from leaveform where ID = @LeaveFormID

If NOT @OldApproved = @NewApproved

BEGIN

 If @NewApproved = 1

 SELECT @BodyText = 'Your ' + @LeaveType + ' leave was approved.'

 INSERT INTO EmailNotification (RecipientID, Subject, BodyText)

```
VALUES (@ApplicantPersonID, @ApplicantName + ' - ' + @LeaveType +  
' Leave Application (' + @FromDate + ' to ' + @ToDate + '), @BodyText)
```

/* Note: the following lines send a e-mail to accounts (hardcoded to Person ID 2548
for K1 LIVE) */

```
Select @AccountsID = '2548'
```

```
SELECT @BodyText = @LeaveType + ' Leave of ' + @NumberOfDays + ' day(s)  
for ' + @ApplicantName + ' was approved.' + CHAR(13) + + CHAR(13) + 'Comments:' +  
CHAR(13) + @Comments
```

```
INSERT INTO EmailNotification (RecipientID, Subject, BodyText)
```

```
VALUES (@AccountsID, @ApplicantName + ' - ' + @LeaveType + '  
Leave Application (' + @FromDate + ' to ' + @ToDate + '), @BodyText)
```

```
END
```

```
< COPY TO THIS LINE >
```