Data Analysis for Auditors

(Practical Case Studies on Using CAAT's)

(Revised and updated Edition)

(Software IDEA CD with 1000 records restriction enclosed with this book for practice)

The Institute of Chartered Accountants of India
(*Set up by an Act of Parliament*)
New Delhi

Revised Edition	:	January, 2016
Committee/Department	:	Committee on Information Technology
Email	:	cit@icai.in
Website	:	www.icai.org/http://cit.icai.org
Price	:	₹ 400/- (including CD containing Demo copy of CAAT Software)
ISBN	:	978-81-8441-172-0
Published by	:	The Publication Department on behalf of Committee on Information Technology, The Institute of Chartered Accountants of India, Post Box No. 7100, Indraprastha Marg, New Delhi-110 002.
Printed by	:	Sahitya Bhawan Publications, Hospital Road, Agra 282 003. January/2016/P1856(Revised)

FOREWORD

It is a matter of immense pleasure that the Committee on Information Technology of the Institute has come out with this comprehensively revised publication on "*Data Analysis for Auditors – Practical Case Studies on Using CAAT's*". The publication aims to provide practical insight into use of auditing tools and techniques by Internal Auditors, Statutory Auditors, Business Control Analysts, Fraud Investigators for their data analysis requirements using Computer Assisted Audit Techniques (CAATs)/ General Audit Software (GAS).

Earnest endeavor has been made to share practical experience in using CAAT tools for professional assignments and research in wide range of industries and experiences gained from interactions with fellow professionals. This compilation of practical case studies is expected to provide the requisite exposure to enable Chartered Accountants to see how CAAT Tools can be used to analyze the operations of an enterprise and recommend steps to ensure better control and efficient functioning of the business, apart from detecting fraud/mistakes.

I am sure that this compilation would be immensely useful to the Chartered Accountants both in practice and Industry. They would be able to further enhance the effectiveness and efficiency of their services to have better insight through better data analysis and reporting using CAAT/ GAS Tools. Using the book the members will be able to easily accomplish global standards in terms of skills, knowledge and competencies as is now being demanded/ expected from the profession.

I compliment CA. Rajkumar S. Adukia, Chairman and other members of the Committee on Information Technology for bringing out this useful publication. I am sure that this initiative would empower the members to enhance their professional offerings.

CA. Manoj Fadnis
President, ICAI

Place: New Delhi
Date: January 13, 2016

PREFACE

The Committee on Information Technology has great pleasure in releasing the revised publication on “Data Analysis for Auditors- Practical Case Studies on Using CAAT’s as a capacity building initiative for members to enhance the bouquet of services that they can offer, apart from enhancing the efficiency and effectiveness of their operations.

Using of Computer Assisted Audit Techniques (CAAT) is gaining popularity within audit departments in enterprises and servicing of clients by Chartered Accountants as they enable us to conveniently undertake better analysis of data and report thereof in a structured manner, apart from keeping an audit trail of the analysis and intermediate results.

The CAATs/ GAS enable Professional Accountants to better analyze the data, detect fraud/ mistakes, and recommend steps to ensure better control and efficient functioning of the business, apart from offering compliance services. This initiative enables professional to address the practical audit requirements and statutory compliances for businesses using CAAT Tools.

We are grateful to CA. Deepjee Singhal and CA. Manish Pipalia for contributing the basic content of this publication.

I extend my heartiest congratulations to the IT Committee and wish the publication a grand success.

CA. Rajkumar S. Adukia
Chairman
Committee on Information Technology

Place: New Delhi
Date: January13, 2016

CONTENTS

S. No	Contents	Page No.

Chapter 1
Introduction

Chartered Accountancy (CA) firms operate within an environment of high expectations from both its clients, the external public and regulators. Regardless of size, CA firms whether Statutory Auditors or Internal Auditors are always looking for new and innovative ways to deliver value to their existing customer base while attracting new clientele. Despite changing regulatory environments and fluctuating economies, one constant to a more profitable practise is lowering costs and time. CA firms require technologies that can add as much value as possible to its clients, in the most cost-effective method available. With audit enabling technologies CA firms can harness the power of data analytics to ensure a structured framework for performing organized, efficient and reliable audits that meet or exceed professional standards.

Audit, financial control and compliance professionals are faced with the challenge of reliably achieving objectives while addressing uncertainty and acting with integrity. Whether the objective is to provide business assurance, improve controls, or ensure compliance, it will be difficult to draw any credible conclusions to help the business without analyzing the underlying data in detail.

However, data analysis specifically for audit, financial control and compliance is easier said than done. The key goal behind the analysis is to identify data discrepancies that need more investigation – this requires more sophisticated capabilities than just summarization and aggregation. Add to this the importance of easily sharing, presenting and distributing analysis results to stakeholders and the search for an effective analysis solution becomes even more challenging.

This book grew out of our desire to provide to internal auditors and other professionals like statutory auditors, business control analysts, fraud investigators, students and educators of this subject the right kind of help and guidance to understand and learn the application of CAATs. We have been using CAATs in our professional practice for the last fifteen years. This book is based on the material researched and experiences gained by us by. We have also incorporated new insights gained by us by talking to fellow professionals on the subject.

Throughout this book our use of the word 'internal auditor' should be read to include other professionals like external auditors, fraud investigators, students, educators, and professionals who would like to work with or are already working with CAATs.

This book uses narrative descriptions, illustrations and case studies to explain CAATs. It includes information on pre-requisites for the application of CAATs, introduction to a data analysis tool, stages in the use of the tool, techniques to examine data and isolate specific items, identifying exceptions and irregularities, profiling data, and various case studies like travel claims review and others, use of the tool in specific industries like Insurance, banking, and in forensic review and interrogation (fraud investigation).

Audit tools may be classified thus:

- Electronic Working Papers
- Information Extraction and Analysis
- Fraud Detection
- Network Security
- Electronic Commerce and Internet Security
- Continuous Monitoring
- Audit Reporting
- Other specialized software tools like Risk Management etc.

Among all the available tools, the ones for data extraction and analysis including fraud detection and continuous monitoring are the most widely used. These have also higher payback in terms of analyzing vast volumes of data for identifying deviations. Because of this the book focuses on only Data Extraction and Analysis tool.

Though there are numerous tools available for extracting and analyzing data, we have used "IDEA" CAAT tool for case studies. The audit objectives and the analysis process covered in these illustrations and case studies are generic in nature. These can therefore be used as a learning base for its application in a specific work environment. We would like to thank Caseware Idea Inc., Canada for giving us permission to use their deeply researched material, including the training material on IDEA software tool in this book, especially screenshots which help us to demonstrate the use of the tool in various live situations.

As audit tools grow more efficient and sophisticated, they also become easier to learn and use. At the same time, they must also fit into a complex and ever changing environment. The features of audit software can easily conflict with the features of other software on the computer or network, and must be carefully managed.

Chapter 2

Pre-requisites for the Application of CAATs

2.0 Tool Selection

The Global Technology Audit Guide (GTAG) 16: Data Analysis Technologies released by the Institute of Internal Auditors (IIA) lays down detailed information which will help potential users pick the right Data Analytical Tool.

Attributes of Data Analytical Solutions for Audit

Data analysis technology for Audit use needs to have the features and functionality that auditors require to do their job effectively. Not only should it deal with the data access challenges, but it also needs to support the way in which auditors work and the types of analytics that are appropriate to the audit task on hand

Mentioned below are some of the key attributes of Data Analytic Tools -

- Able to analyse entire data populations covering the scope of the audit engagement.
- Makes data imports easy to accomplish and preserves data integrity.
- Allows for accessing, joining, relating and comparing data from multiple sources.
- Provides commands and functions that support the scope and type of analysis needed in audit procedures.
- Generates an audit trail of analysis conducted that is maintained to facilitate peer review and the context of the audit findings.
- Supports centralized access, processing and management of data analysis.
- Requires minimum IT support for data access or analysis to ensure auditor independence.
- Provides the ability to automate audit tasks to increase audit efficiency, repeatability and support for continuous auditing.

Ranking Card to choose the right Data Analytical Tool for Audit

Auditors may employ the comprehensive ranking card (below) while evaluating suitable Data Analytical Tools for his/her engagements:

Need: 0=Needless: 1=Nice to Have: 2=Desirable: 4=Mandatory		Need
Internal Audit Strategic Objectives		
1	Software is easy to learn and use	
2	Competitive advantage	
3	Minimize reliance on IT professionals	
4	Improve work accountability, responsibility and supervision	
5	Enforces production program change controls	
6	Reliability: bug free, speed, work like a professional	
7	Portability: runs on a laptop	
8	Scalable: grow from desktop to server without learning new software	
9	Data integrity and security: client data is protected from auditor change	
10	Collaborative features	
11	Supports development of automated and continuous programs	
12	Compatible with electronic work papers	
13	Improves documentation of audit work completed.	
Provider and Implementer Support		
14	Global presence	
15	Years in business	
16	Multiple languages	
17	Help desk available	
18	Ease of doing business: knowledgeable in auditing needs	

Need: 0=Needless: 1=Nice to Have: 2=Desirable: 4=Mandatory		Need
19	Regular software upgrades	
20	Training readily available	
21	User group program for networking with other users of the Tool	
22	Knowledgeable consultants independent of the provider available	
23	Getting started programs available	
Technical Features and Functionality		
24	Import all file types used by the organization	
25	Handles large file record sizes	
26	Handles large data volumes	
27	Ease in validating and reconciling data import	
28	Modify imported data field properties	
29	Support search for text, numbers and time.	
30	Project visual chart or mapping of data actions performed	
31	File join/merge/compare	
32	File append	
33	Visual Connector	
34	Sorts, indexing, filtering and fuzzy logic	
35	Summarization	
36	Extraction	
37	Pivot Table	
38	Stratification	
39	Gap Detection	
40	Aging	
41	Compare data to predicted data – Benford's Law	
42	Advanced statistical analysis: correlation, trend analysis, time series	

Need: 0=Needless: 1=Nice to Have: 2=Desirable: 4=Mandatory		Need
43	Sampling	
44	Statistical analysis	
45	Export to typical office applications	
46	Create custom reports and graphics	
47	Create simple and complex calculated fields	
48	Data cleansing solutions – character @functions available	
Cost		
49	Software purchase	
50	Implementation costs – scripting and components	
51	Upgrade fees	
52	Annual help desk support	

The IIA's analysis made several key recommendations for internal auditors to consider before using of CAATs:

- Determine the organization's audit mission, objectives and priorities: Auditors must consult with the management of organizations regarding their audit functions of the highest priority and where exactly CAATs could be used to meet those priorities.
- Determine the types and scope of audits: To determine the specific objectives of each audit and its terms of reference. Information on this is crucial for making an appropriate software decision.
- Consider the organization's technology environment:To make sure that selected CATTs will have to operate with the software, hardware and network systems of an organization.
- Be aware of the risks – Applying software to any mission-critical function carries some risks, and CAATs is no exception to this. As the IIA article suggests, CAATs can prompt auditors to jump to premature, faulty and hasty assumptions and conclusions Data should not be taken for granted and every finding revealed by the CAAT should be thoroughly validated for its accuracy and relevance.

In short good CAATs should possess:

- Simple and intuitive interface
- Menu driven functions rather than command driven functions
- Ease of data extraction
- The ability to access a wide variety of data files from different platforms
- Ease of data analysis
- Simplified query building
- Comprehensive logging features with ‘read’ rights only - unalterable
- Easy template creation
- Convenient template recall and use
- Ease in automation of tasks
- Simplicity in scheduling of automated tasks for continuous monitoring”

Appendix I gives an illustrative list of CAATs.

2.1 Tool Application

2.1.1 Management support for the optimum use of CAATs

- The CAATs should be made a vital part of the Internal Audit Charter and should have ready support of the Audit Committee as a key audit resource.
- The management of an organization should perceive the real value of CAATs and should encourage the Internal Auditor to apply it optimally, and the Information Systems (IS) department should to support the data flow/ availability for the use of a CAATs.

  At one location, where the experiences with CAATs were highly successful, the senior management of the organization encouraged the Internal Auditor to make optimum use of the CAAT and MS-Excel in day-to-day data analysis. Moreover, based on the successful experiences of the CAAT within Internal Audit, the management decided to roll-out the CAAT to process divisions like Sales and Purchase. Eventually the CAAT was made an important Management Tool.

2.1.2 Data availability for the optimum use of CAATs

- Access to open data is a must for the use of CAATs. Encrypted data, like the data in 'Tally' Accounting Software, cannot be accessed directly. (Tally is a widely used accounting software in the South Asian region). The 'Tally Vault' prohibits direct link to the data tables in 'Tally'.
- Information Systems (IS) Administrator should appreciate the functionality and application of the tool. Considering the confidentiality and integrity of the data desired, an elaborate plan of data access, data storage and data use should be identified. Support is necessary in the form of organizing data availability for specific audit objectives in formats desired by the Auditor. A working procedure should be established in which the roles of IS officials and Auditors are clearly defined for ensuring the easy availability of data.

2.1.3 Internal Auditor skill-sets for the optimum use of CAATs

- The user of CAATs should be innovative and imaginative and capable of thinking out of the box.
- The user should be computer literate and possess a working knowledge of Windows and any Office Suite.
- He/She should have adequate audit knowledge and experience to define
  - ✓ Audit Objective/s
  - ✓ Data required for achieving audit objective
  - ✓ Steps to be employed within the CAAT to achieve the intended results in minimum time.
  - ✓ Likely benchmarks for validating results – adequate industry domain knowledge

Our work has led us to the conclusion that a CAATs Auditor does not need to be a programmer with years of intensive programming experience.

2.2 Good processes for the implementation of CAATs

- Selecting a user champion/s for the CAATs, preferably within the existing audit group who has spent relatively long years in the function/organization/industry.
- Organising a structured training programme covering product functionality with case studies and including organization specific cases in the training programme. Involving all users in the training programme at all levels. If the audit group is large, the training programme could be a "Train the Trainer" type.
- Identifying a plan for introducing the tool by linking it to the existing audit programmes. Follow the KISS--'Keep It Short and Simple--'concept in its introduction and follow it up by using advanced functionality. The period of evolution would depend on the competence of the team members, and could range from six months to two years.
- Reviewing utilization every six months to ensure its continuing use.
- Availing continuous support of the vendor. This way the tool is updated at all times and is free from known/unknown bugs.

We have noticed in countless instances that a nominated CAATs Auditor is overburdened with day-to-day audit responsibilities, which prevent him from devoting his physical and mental energies to effective implementation of the CAATs. A nominated CAAT Auditor should be given free space and time to explore the CAATs fully. At the outset it may seem a bit costly for the organization, but it will start paying back within three to six months.

Chapter 3

Regulatory Drivers for the use of CAATs Worldwide

Pronouncements by Academic and Audit Regulators on the use of CAATs

3.0 Introduction

Numerous regulatory bodies from all parts of the world have backed the use of CAATs: from Accounting and Audit Regulators and Watchdogs.

Some of their pronouncements are reproduced below, -

3.1 The Institute of Chartered Accountants of India (ICAI)

Recognising developments in the field of technology and their impact on the accounting profession in India, Auditing and Assurance Standards Board issued Auditing and Assurance Standard (AAS) 29, Audit in Computer Information Systems Environment in September 2003. This Guidance Note on Computer Assisted Audit Techniques is a sequel to that AAS. The Guidance Note deals extensively with the concept of CAATs, with focus on the areas where it may be used, considerations for this use, controlling the application of CAATs, documentation required for its use, and its use in small entities. The Guidance Note also contains a comprehensive appendix containing examples of CAATs, their description and comparable advantages and disadvantages of each of the CAATs.

From the *Handbook of Auditing Pronouncements*, Vol. II (Edn. 2005)'

An excerpt from the Guidance Note on Computer Assisted Audit Tools (CAATs)

"The application of auditing procedures may, however, require the auditor to consider techniques known as Computer Assisted Audit Techniques (CAATs) that use the computer as an audit tool for enhancing the effectiveness and efficiency of audit procedures."

3.2 The American Institute of Certified Public Accountants (AICPA)

An Excerpt from the Statement on Auditing Standards No. 94, The Effect of Information Technology on the Auditor's Consideration of Internal Control in a Financial Statement Audit, AICPA, New York, USA, May 2001. (Amends Statement on Auditing Standards No. 55, Consideration of Internal Control in A Financial Statement Audit, AICPA, New York, USA, April 1988.)

“The auditing standard states that computer-assisted auditing techniques (CAATs) are needed to test automated controls in certain types of IT environments. This paper revisits auditing-through-the-computer techniques, which should become more widely used with the issuance of SAS No. 94, and focuses on the test data technique, which can be applied in almost any audit to test automated programmed controls.”

3.3 International Audit Practice Committee (IAPC) of IFAC

In March 2000, the International Audit Practice Committee (IAPC) of IFAC released an (exposure draft?) on four topics, which form a supplement to ISA (International Standard on Auditing) 401 "Auditing in a Computer Information Systems Environment (CIS)."

“According to the exposure draft, the purpose of the statement on CAATs ”*is to provide guidance in the use of Computer Assisted Audit Techniques (CAATs), which are techniques that use the computer as an audit tool."* The exposure draft "*applies to all uses of CAATs involving a computer of any type or size."*

3.4 Information Systems Audit and Control Association (ISACA)

Excerpts from the Document G3

1.2 <u>Need for Guideline</u>

1.2.1 Computer Assisted Audit Techniques (CAATs) are important tools for the IS auditor for performing audit.

1.2.2 CAATs include many types of tools and techniques, such as generalised audit software, utility software, test data, application software tracing and mapping, and audit expert systems.

3.5 Hong Kong Institute of Certified Public Accountants

Practice Note 1009 provides guidelines on the use of CAATs.

Excerpts from the Practice Note:

“The overall objectives and scope of an audit do not change when an audit is conducted in a computer information technology (IT) environment. The application of auditing procedures may, however, require the auditors to consider techniques known as Computer-Assisted Audit Techniques (CAATs) that use the computer as an audit tool.”

“CAATs may improve the effectiveness and efficiency of auditing procedures. They may also provide effective tests of control and substantive procedures where there are no input documents or a visible audit trail, or where population and sample sizes are very large.”

The full Practice Note 1009 is reproduced in Appendix II.

Chapter 4
Stages in the use of CAATs

CAATs are used in the following stages:

1. Identifying the audit objectives and areas for the use of CAATs
2. Determining the data required and arranging the download
3. Importing data
4. Reconciling data
5. Working with the data imported: Use of Functions and Scripts within the CAATs
6. Reviewing of results and housekeeping

4.0 Identifying the audit objectives and areas where the CAATs can be used

A list of audit objectives to be achieved should be identified by the internal auditor at the outset. The audit objectives may include financial and operational dimensions for various business processes. They may also include special investigations and/or reviews into alleged fraudulent activities or business analysis at the request of the top management.

The objectives should be clearly documented to cover all the desired aspects of audit. The Internal Auditor needs to make a check on the feasibility and cost/benefit aspect of the application of CAATs for each of the objectives. This is a one-time exercise which will yield good results in the longrun.

By feasibility is meant the easy availability of data in an electronic format for each objective. For example, a r*eview of payments to contract labour at a manufacturing site cannot be checked through CAATs for validity and accuracy of payments if the attendance records are logged manually.*

By cost/benefit analysis is meant the cost and effort to be invested in obtaining the data vis-à-vis the benefits in money terms/in kind which can be reaped. For example*, instances where a borrower has overdrawn the lower of his sanctioned limit or drawing power in a bank are normally available in the 'End of Day Reports' of the Core Banking System. It will be uneconomical to verify this objective by using CAATs.*

4.1 Determining the data required and arranging for the download

The audit objectives should be mapped against the data that should be available in very clear and accurate terms. The internal auditor should seek the assistance of the data owner and IS department in this regard. This can be done as follows:

4.1.1 Request IS for Data

4.2.2 Request the Data Owner for Data

4.2.3 Access to the Database Server

4.1.1 Request IS for Data

The Internal Auditor should send an email to the IS Manager. It should contain

- A clear and full description of the audit objective
- The period for which the data is required
- The data fields essential to the test.
- The format in which the data is required
- The medium through which the data is to be conveyed to the internal auditor: email, CD, network LAN access, etc.

This approach is explained with the help of an example from the banking industry

Electronic 'Know Your Customer (KYC)' Audit

The team selected an objective to test KYC norms on current account customer master data.

So the team issued a Data Request to IS in the following format:

- Data required: Current account customer master information.
- Period:(As of the date of Audit: reword this because date and period are different things).
- Fields of reference: Branch ID, Customer ID, Account ID, First Holder & Joint Holder/s Name, Address, PAN No., Mobile No., Residence No., Office No., Mode of Operation and Clear Balance.

- Format of Data: Text form.

In response to this, the IS ran a SQL query on the production database and generated a text file dump which was saved in a secure folder with special access to the audit team only.

4.1.2 Request the Data Owner for Data

The internal auditor should send an email to the data owner, which should contain

- A clear and full description of the audit objective
- The period for which the data is required
- The data fields essential to the test
- The format in which the data is required
- The medium through which the data is to be conveyed to the Internal Auditor: email, CD, network LAN access, etc.

Here the data owner/process owner/department head provides data to the internal auditor either through the standard report suite of the application system or through some specialized Business Warehouse solution like Crystal Reports, BIW, COGNOS, and SAP-AIS.

This approach is explained with the help of an example from the banking industry

Audit of Loans and Advances MIS from Business Warehouse (BIW) for approval verification

The team decided to audit the MIS report on Loans and Advances generated from the Business Warehouse (BW) for approval verification. The MIS report is a comprehensive listing of Account ID, Type of Loan, Type of Security, Type of Industry, Sanction Limit, Drawing Power, Rate of Interest, Due Date of Loan, and Approval Officer.

To complete this test, the team sought assistance from the Merchant Banking Wing of the Bank. The Loans officer for Branch “X” generated the MIS report for the period--1st April 2014 to 31st March 2015. This report was saved as a Microsoft Excel file and provided to the team on a CD.

The team imported the Microsoft Excel file using the MS-Excel option for import within the Generalised Audit Software (GAS).

4.1.3 Access to the Database Server directly

There is a common belief in professional Information Technology (IT) circles that while using CAATs, connectivity to data servers should not be allowed. Normally, as the data server contains data in a raw tabular format by using CAATs on the server directly, it could compromise the integrity of the data and slow down performance. While performance may be a factor that can be considered in some circumstances, integrity is not a concern. In order to avoid controversy, production data servers need to be avoided. However, there are organizations where the IT officials are willing to provide access to a Flash Server, Disaster Recovery Site Server or a Training Server, which has a historical image of the production data server.

There is also a myth that because servers contain a million data tables, identifying the right data table for each objective is virtually impossible. This needs to be cleared at the outset. Take the SAP server, for instance. The SAP server has not more than a few thousand tables. Identification of the right table is difficult but not impossible. .

Access to any server requires the following to be in place:

- The workstation on which the CAATs is loaded, (and which attempts to access a specific server should contain the client end data server drivers this part of the sentence is not clear) For instance, Microsoft ODBC for Oracle, Oracle for Ora 92 drivers for Oracle, Microsoft SQL drivers for MS-SQL, etc.
- The CAATs should contain the ODBC facility. ODBC is a middleware which allows third party querying/reporting tools to connect to database servers and access server based table information. ODBC is a secure acceptable data conduit.
- Server reference and schema (logical modules (?) like MM, FI, SD in SAP) reference.
- User access 'READ' rights to the server.

Once the connectivity is in place, good CAATs allow the internal auditor to firstly write SQL queries on the raw data table at source itself to limit the data to be imported. Popular CAATs also allow the user to include or exclude fields of their choice at the data import stage itself.

This approach is explained with the help of an example from the banking industry.

With the help of the IS wing, the team set up connectivity between the CAATs and the FINACLE ORACLE DR site server. (FINACLE is a Core Banking Solution used in many banks in India and abroad).

The test was undertaken to identify Term Deposit accounts, where the deposit was accepted for more than 120 months.

As per the bank-rules for acceptance of Term Deposits and guidelines of the regulator, the bank cannot solicit Term Deposits in excess of 120 months. This test was undertaken to identify non-compliances with the rule.

The team connected the DR site server using the CAATs and identified the data table containing the Term Deposit account masters TAM. At the import stage, the team selected three fields – Account ID, Account Name and Deposit Period in Months from the table containing 15 different fields. The team also entered an SQL query in the GAS, filtering the data for Branch "X". The import of data of 0.1 million rows/lines was initiated and duly completed within 1 minute. This was faster in comparison to writing an SQL query which would take 3-4 hours to run in the past.

4.2 Import of data

In order to interrogate and analyse the data by using CAATs, the data must first be 'imported'. Good CAATs contain a rich, robust, easy-to-use, screen based import facility. Many a times the file to be imported is very complex (like a multi-line text report file). In such cases, CAATs should offer the facility of saving the import process in the form of a reusable template. The template can be used as long as the report format (fields and field information) does not change.

The import facility should read the following file types (extensions provided):

- Spreadsheet files – MS-Excel (xls), MS-Access (mdb), Dbase (dbf)
- Lotus Files (.wk1, .wk3, .wk4)
- Text files – Data Files (dat), Print Report Files (prn)
- Portable Document Format (PDF) Files. PDF files are gaining prominence on account of the security which comes with such files.
- Delimited Files – Comma Separated Value Files (csv)
- Flat Files – ASCII Files (asc), EBCDIC Files
- Extendible Markup Language (XML) Files commonly found in Internet

Based Web Applications like Online Air Ticket Reservation Systems – SABRE.

- Audit Information System (AIS) from SAP.

4.2.1 Easy Practical Tips on Technical Issues for Data Import

- Request File Layouts from the IS Department, which are normally called record definitions, data dictionaries, schemas or a host of allied names.
- Take IS help in identifying salient fields in file layouts.
- Review the data with some knowledgeable users on screen or through hard-copy review to identify important fields for selection and inclusion. For example, date of birth field may take a whole new meaning from a validation on child fares or child facilities. Otherwise this information may be normally excluded.
- Pay attention to type, ledger and product coding systems.
- If a report that has all the required data consistent with the audit objective can be located, then go for that file.
- Many IS support staff will automatically format a report to an 80 or 132 column width to fit standard paper sizes. As the report is to be printed to a file and not to paper it can be as wide as it needs to be. IDEA software can approximately accommodate 256 odd columns.
- Before transferring a large file over a network, carry out a test on a small file to determine transfer time and find out if resource will be inadequate for other system users. If yes, maybe the large file data import can be scheduled outside working hours or on holidays.
- Many files include “deleted” or old records that may not be included in control totals. For example, payroll in which ‘left employees’ (terminations) are often kept on the system for tax or social security reasons. Identify such items and exclude them.
- To import COBOL files, take assistance from the COBOL file definition Section. COBOL files can be imported by creating a record definition for the file and then using the very same Record Definition Editor (RDE) to import the file. Alternatively, it can be imported through the ASCII option.

- Printouts are generally formatted with page headers and footers as well as column headers. These should be excluded when specifying the data to import. However, reports often contain "header" information such as an account number, region or area code, etc., which needs to be included for data completeness. Mark off such headers and identify a consistent pattern.
- Data residing in columns can be identified as separate fields based on delimiters within the fields.
- Totals and sub-totals should be excluded from the selection process. Reconciliation is very critical to ensure that 'include' and 'exclude' features have been correctly dealt with while importing.
- In some cases the amount of data is too lengthy to be printed in a single line. In such cases, reports are designed to wrap onto two or more lines. Other reports lines may wrap unintentionally due to the printer set up. Such reports must be 'unstacked' to produce single-line records prior to specifying the columns of data to import.

While generating a report in print file form from an application, it is prudent for the auditor to be aware of report generation facilities and not depend fully on IS. This serves as a foolproof cross check on data integrity of files imported.

4.3 Backward Reconciliation of Data to the Host System

Once the data has been defined and read into the CAATs, it is imperative that before commencing audit testing, it is reconciled back to the "host" system. Given below is the list of errors that could occur during the process:

- Requesting wrong data;
- Being supplied with wrong data (and/or for the wrong period);
- Errors on the extraction of the data from the host system;
- Errors when importing the data like invalid or bad data in certain fields due to the inconsistent nature of transaction entry in that field.

The data can be reconciled back to the host system by reviewing:

- Numeric field totals;
- The record count;
- A sample of records compared to a print from the host system.

It is important that reconciliation reports and control totals are requested along with the data and filed with the documentation from the download/import process.

Data can be reconciled in different ways. CAATs contain Field Statistics and Control Total options to reconcile totals and to ensure that the dates match the data requested and do not include transactions from another period. This is the essence of cut-off checks.

4.4 Use of Functions within the CAATs

On the data analytical front CAATs can do the following –

- Isolate Specific Items
- Identify Irregularities
- Profile
- Categorize
- Field Manipulation
- Sampling
- File Look Ups
- Merging of Data Files
- Repetitive Analytics

4.5 Review of Results and Housekeeping

4.5.1 Review of results

Findings arising out of an analytical task are potential cases for reporting. They turn into actionable findings post validation only. The internal auditor should use CAATs as a means to an end and not as an end in itself. CAATs is simply a tool to examine and review data. The results need to be discussed with the process owner in the light of the objective being tested, data provided and circumstantial facts in the process. This also needs to be tested vis a vis the business trend, company performance trend, practical possibility of such results in the area of audit and other non- financial and operational areas.

CAATs can display instances where the net accounts receivable exceed the credit limit of a customer. This need not necessarily indicate deliberate attempts by ‘Sales’ to bypass the safe credit exposure limits of a customer. The need to ‘force-sales’ may be justified by the need to meet year-end sales targets. This can be supported by due pre-facto approvals and post balance sheet date verification of collections.

4.5.2 Housekeeping

Data extraction and analysis should be undertaken in distinct file-folders within the CAATs. Each assignment should bear separate reference of a file-folder which is properly identified and stored. The internal auditor should follow appropriate record/ folder backup and retention practices and ensure consistent implementation by his team. The file imported, its template (if applicable), the resultant output files, queries/equations, views, graphs and more should all be stored together for easy recall and use.

For example, in a cyclical Payroll Audit, the pay file, attendance file, and allied files for historical cycles should be regularly retained on a separate drive of the same workstation or on an off-site location, appropriately named and indexed to make way for the current files. The back-up should be easily identified by any member from the internal audit Team whenever required. This is essential for electronic evidence management and audit automation – macro (audit routine) creation, record, playback and scheduling.

Chapter 5

Examining Data and Isolating Specific Items

5.0 Introduction

Data once imported can be examined and reviewed in IDEA through

- Indexing
- Sorting
- Grouping of Records
- Go To
- Find, Find Next and Find Field

Isolating specific items can be accomplished in IDEA through

- Display of Data
- Extraction of Data

Examining data and Isolating Specific items allows the user

- ➢ The conveniences of speedy processing of tasks within IDEA, since databases are largely indexed prior to performance of any major function in IDEA.
- ➢ The benefit of efficient auditing, as more can be accomplished in a lesser time.
- ➢ The advantages of effective auditing, as the user can directly move-in on exceptions, irregularities, and anomalies thereby meeting his/her objectives fully.

Examination of Data and Isolating Specific items briefly includes–

Indexing and Sorting: Displaying the data in a specified format for easy visual interrogation; for instance, visual examination of top 10 non-performing assets for a bank by product category.

Grouping of Records: Allowing the user to view multi-layer information in a tree-like structure; for instance, welfare spends by a state undertaking Zone City Taluka Village wise.

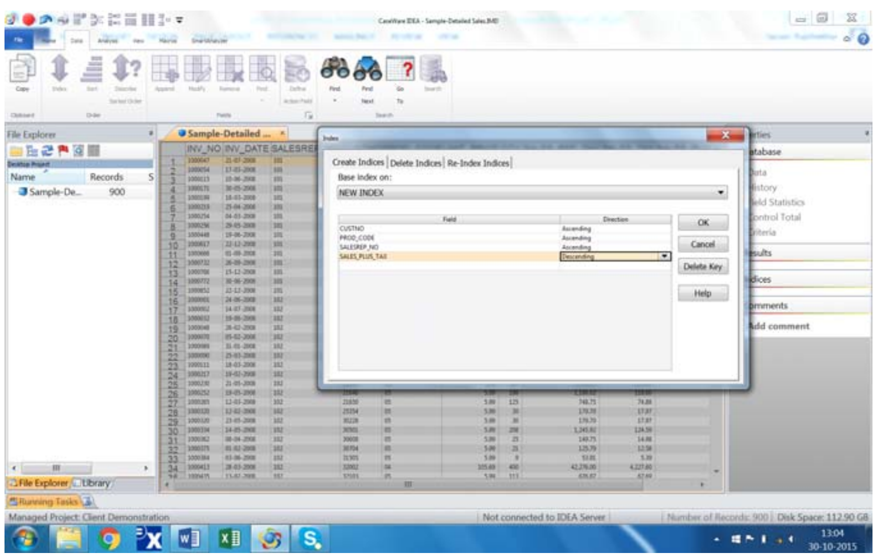
Go To and Find: Providing the interrogator with quick means of zeroing in on a specific record, value, string, or date. Find can be used for to identify all treasury transactions made on Sundays or public holidays for a foreign exchange center.

Display of Data and Extraction of Data: Facilitating identification of all records that meet specific criteria within the database. For instance, all claims admitted in a General Insurance industry wherein the Claim Loss Ratio is over 100%

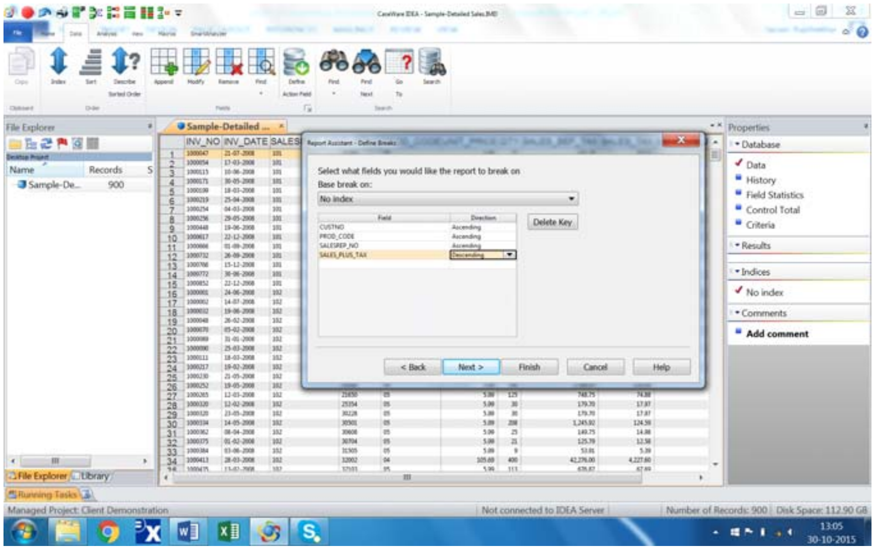
5.1 Indexing

The Indices task is used to specify the sequential order of the records – high to low, earlier to latest. The records are not physically sorted into a separate database; only an index is created to describe the logical sequence in which the records are to be read, displayed, printed, and processed.

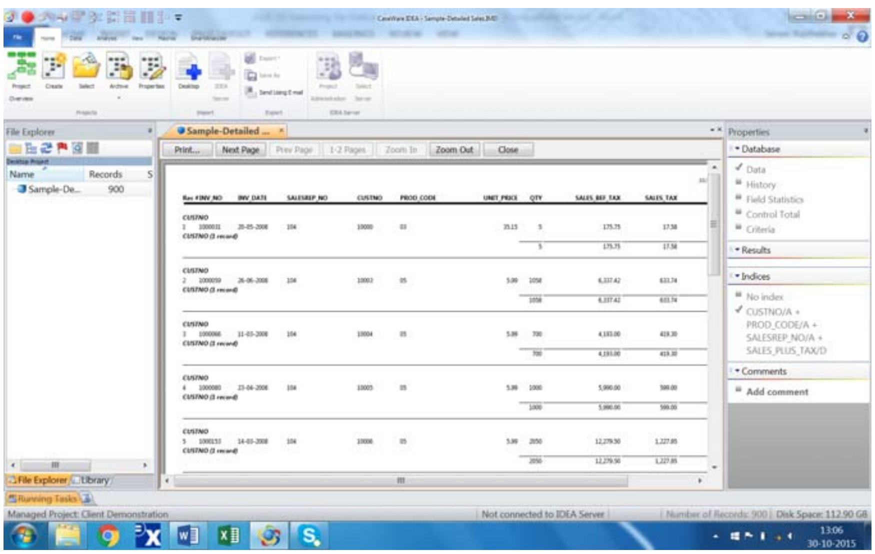
The field(s) you select to determine the order of the database is known as the key. You may select up to eight fields for the index key.



In addition to determining the sequence in which records should be displayed or printed, you can use indices to insert control breaks (or totals) in reports on change of key.



The report preview would appear like what is given below –



Many IDEA tasks will automatically sequence the database(s).

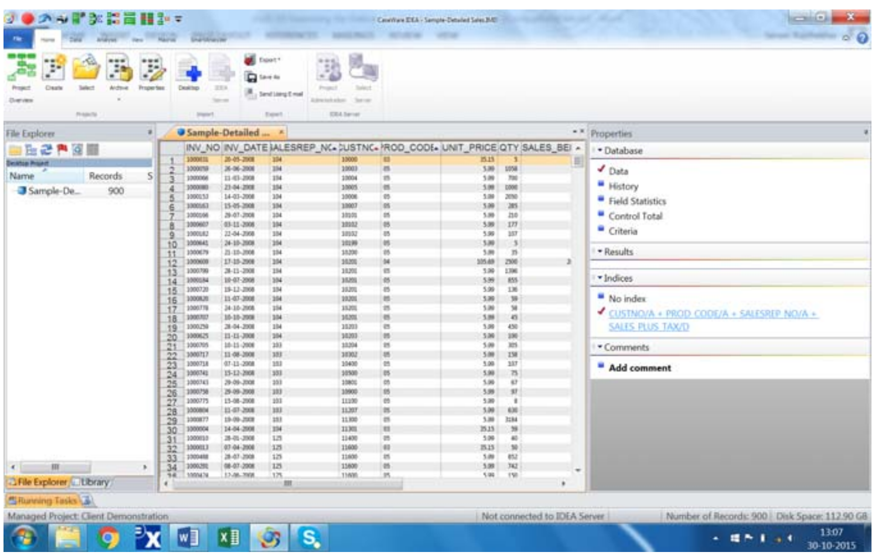
- Duplicate Key Detection
- Duplicate Key Exclusion
- Gap Detection
- Join Databases

- Summarization
- Compare Databases
- Stratification
- Aging

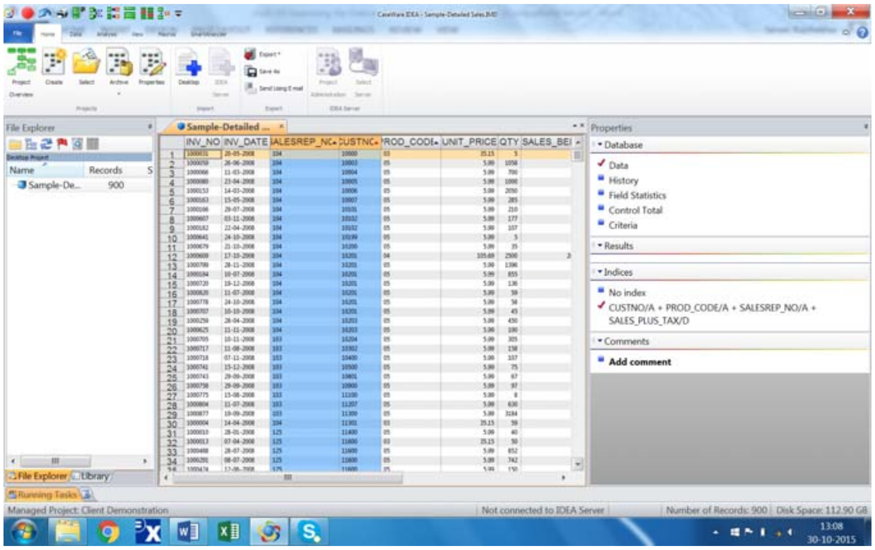
However, it is not necessary to sequence the database(s) before selecting these tasks, as the creation of the index is carried out within the task, if required. If the index or sort order has already been generated, then it is used; otherwise it is created once the test is initiated. The index order generated is then added to the list of available indices for that database.

When selecting an index order, the most significant field is selected first: this is followed by the next most significant field, and so on down to the least significant field. The length of time to index a database is proportional to the length or number of "characters" in the index key.

Once you have created an index, IDEA displays the records in the database in the index order. All generated indices are available in the Indices area of the Properties window. The current index is highlighted with a check mark.



IDEA also displays a directional arrow on the indexed field column header to allow you to easily identify the indexed field and its order (Ascending or Descending). The primary key index is indicated with a red arrow while secondary keys are indicated with a blue arrow.



The Indices dialog box provides three choices:

- Create Indices
- Delete Indices

  You can delete unused indices through the Delete Indices tab of the Index dialog box or through the right-click menu of the Indices area of the Properties window.

  Delete unwanted indices if you require additional disk space. However, as many IDEA tasks require an index and indices are often used when printing reports (to provide control breaks and totals), it is advisable to leave the indices until the end of the audit, unless additional disk space is required.

  If you create an index with the wrong key or by mistake, delete the index so that you do not use it inadvertently at a later date. If you are connected to IDEA Server and you delete an index in a Shared project, you are deleting the index for all users associated with that project.

- Re-generate Indices

  Occasionally, the index on a database may become invalid. This may happen if the index is based on an Editable field or a Virtual field where the equation has been changed.

Use the Re-Index Indices task to regenerate the index for any pre-defined index, thereby eliminating the need to delete and re-create the index. Old indices can also be re-indexed through the right-click menu in the indices area of the Properties window.

Once you have re-indexed a database, IDEA displays the records in the database in the index order and updates the list in the Indices area of the Properties window. The current index is highlighted with a check mark. IDEA also displays a directional arrow on the indexed field column header to allow you to easily identify the indexed fields and their order (Ascending or Descending). A red arrow indicates a primary key field, while a blue arrow indicates a secondary key field.

If you require only a single field for a key, perform a quick index.

When using the quick index method, you can only index the database by a single field.

<u>To index a field in ascending order</u>:

Double-click the field name.

<u>To index a field in descending order</u>:

Double-click the field name a second time.

<u>To return to the default order</u>:

Click the No Index link in the Indices area of the Properties window.

5.2 Sorting

The Sort task is used to create a new database that is physically sorted in a specified order. If you require a sequenced database and do not wish to create a new one, use the Indices task, which creates an index describing the logical sequence in which the records are to be read, displayed, printed, and processed.

Many IDEA tasks require the database to be sequenced. These tests run faster using a sorted database rather than an indexed database.

- Duplicate Key Detection
- Gap Detection
- Join Databases
- Summarization

- Compare Databases
- Stratification
- Aging

However, sorted databases require additional disk space. Once a database is sorted, you need to consider if the original unsorted database on which the sorting was performed should be deleted.

The field(s) used to determine the sort order of the database is known as the key. You may select up to eight fields for the key.

When selecting a sort order, the most significant field is selected first, followed by the next most significant field, and so on down to the least significant field. The length of time to index a database is proportional to the length or number of "characters" in the sort key.

Note

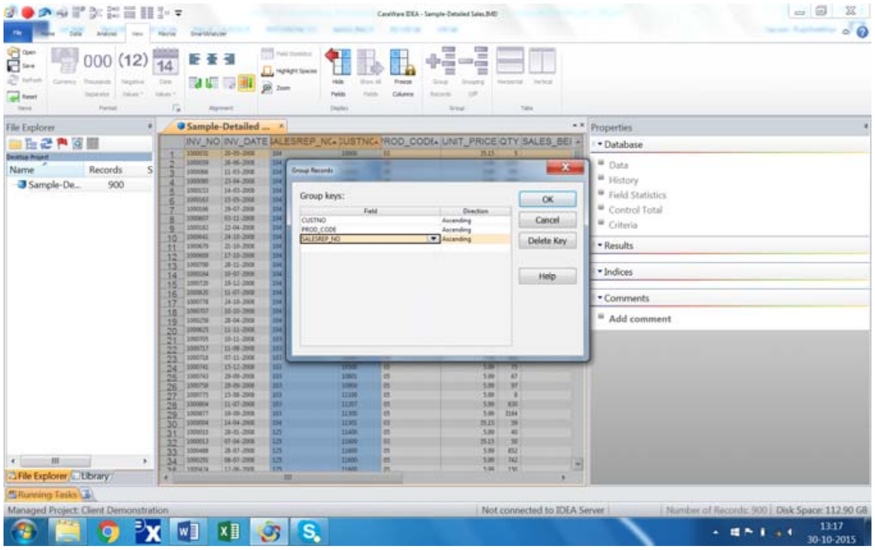
Sorting characters is based on the alphabet as well as the case. Characters are sorted by the alphabet first, and then the lowercase alphabet is followed by its uppercase. For instance, the characters C, c, D, and d are sorted in the following order: c, C, d, D.

With the Sort task, a new database is created with the records in the sequence of the key. This new database is a child database and appears in the File Explorer below the main database. You can view the child database by clicking on the expand button next to the main database.

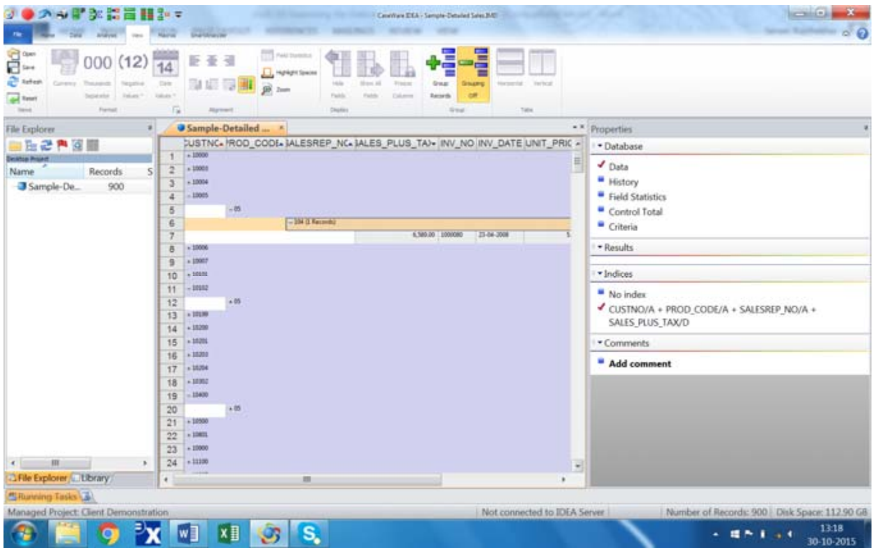
Once you have sorted a database, IDEA displays the records in the database in the sort order and updates the list in the Indices area of the Properties window. When an index is selected, a red directional arrow appears in the column header of the primary key field, while blue directional arrows indicate secondary key fields.

5.3 Group Records

Use the Group Records task to organize matching records in indexed fields (keys) into expandable/collapsible groups. This task will run on local databases and server databases (if you have an IDEA Server).



To view the records that comprise a group, click the nodes to the left of the records.



The Group Records task will run faster if the selected database has previously been indexed on the field(s) selected as the key(s). If not, IDEA will automatically index the database by the key selected before performing the Group Records task.

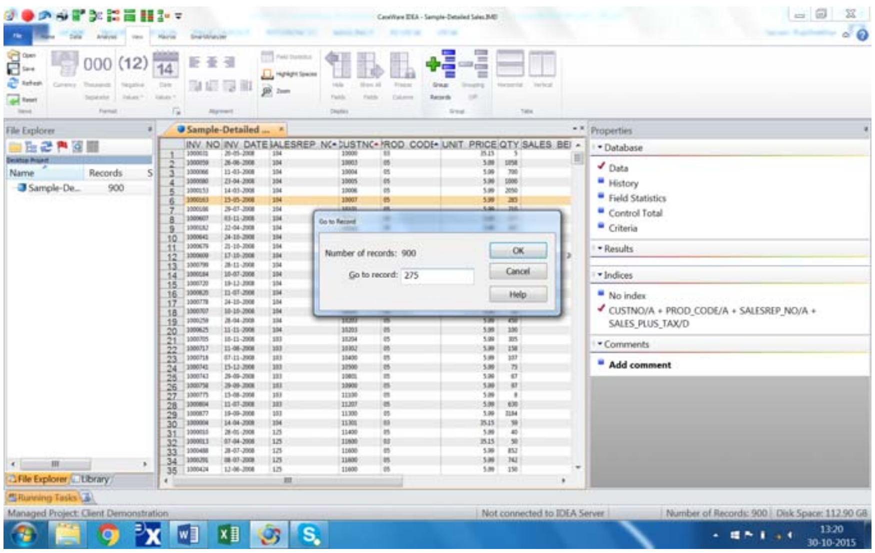
Upon completion of the Group Records task, IDEA moves the fields selected as keys to the left-most position of the database grid.

The results of the Group Records task are display-only. The Print, Print Preview, and History features are not available for this task.

Group Records mode can be turned off by clicking on Grouping Off under View in the IDEA menu tool-bar.

5.4 Go To

The Go To option is used to go to a specific record number within the active database, which is often a quicker method than paginating through the entire database.

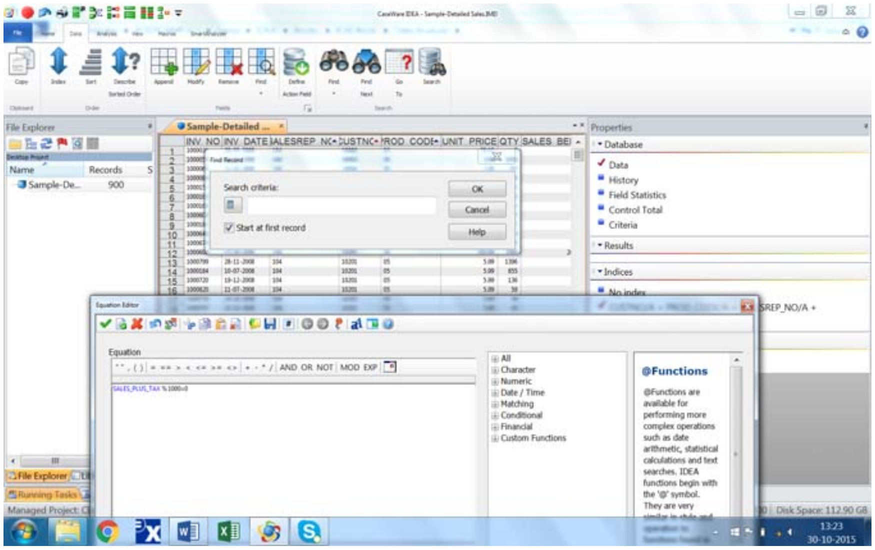


5.5 Find

Use Find to locate records within the active database that meet the specified criteria/parameter or condition.

For example, to find the first sale entry to a customer which is a round sum sale in multiples of 1000:

Select Data – Search category – Find – Find a Record



Click the Equation Editor Button.

Enter the expression SALES_PLUS_TAX % 1000 = 0.

Click OK. You may find further records that meet the same criteria by pressing F3.

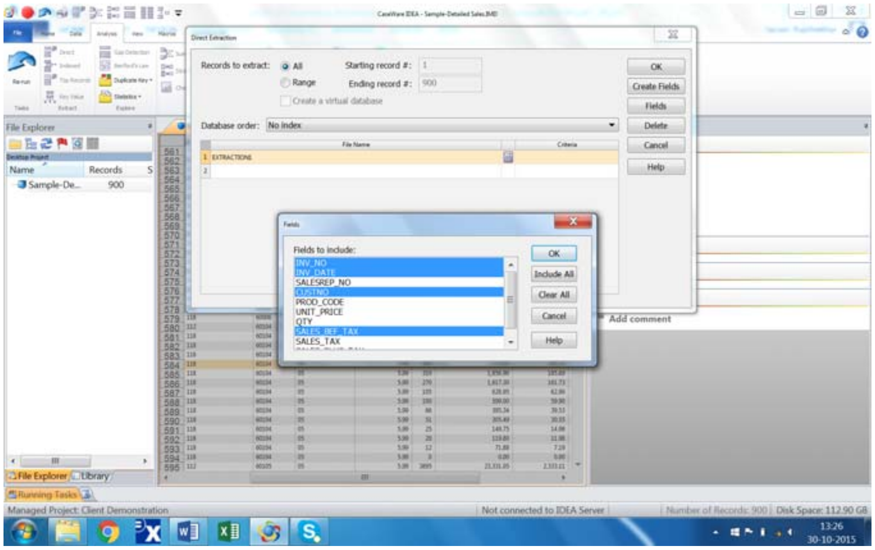
Find differs from Criteria and Direct Extraction by searching for the first occurrence of the specified criteria from the first record in the database, or optionally, the current cursor position. It does not find and list all occurrences.

Find also differs from Search because with Search you do not need to enter criteria. With Search, you can perform keyword searches for data within fields and across records in a single database or multiple databases located in the same Working Folder or project (if you have IDEA Server).

You can also use Find Next to scan through the database searching for further records satisfying the criteria.

5.6 Extraction of Data

Direct Extraction allows the user to write and run criteria (Equations and Functions) directly on the IDEA Database. This task generates an output database which can be further interrogated within IDEA or taken up for manual validation.



Nature of the Extraction

Direct Extraction can perform a single extraction on a database, or up to 50 separate extractions with a single pass through the database. This reduces the processing time of tasks within IDEA. For example - the user can look for high value, round sum, cash payments made to all Vendors containing a specific string like '& Assoc", through a single pass of the database. For each extraction separate databases will be created within IDEA.

Basis for the Extraction

Extraction can be performed on the entire database or on its sample through data range specifications. For example, extractions can be run on the sales of all retail stores in India or for a specific range of stores, if the store references are available in the database and uniformly formatted.

Display of Results in the Output Database

An option is available to select the order of the records appearing in the output database, such as oldest inventory items appearing first on the output list.

Extractions at times run on raw data imported into IDEA. This raw data (say from an Oracle Billing Table of a Telecom Utility) contains all the relevant and irrelevant data field schemas. IDEA offers the user the flexibility of choosing the fields relevant to the extraction test which need to feature in the output database.

Writing Equations and Functions within the Extraction

The Equation Editor can be used to enter the extraction criteria. The Equation Editor allows the user to enter arithmetical, logical, and conditional equations and functions in the equation box.

A <u>normal equation</u> consists of

- Parameter / field within IDEA.
- Arithmetical operator like +, -, *, and more.
- Logical operator like .AND., .OR., .NOT.
- Constant like a number or character.

Sample Equation – (DebitCredit = "CR" .AND. PAN = "" .AND. Amount > 50000)

Here the Parameter / field is DebitCredit, PAN and Amount. The arithmetical operator is = and >. The Logical operator is .AND. The constant/value is 50000

A <u>normal function</u> consists of –

- @ function
- Parameter / field within IDEA.
- Constant like a number or character.

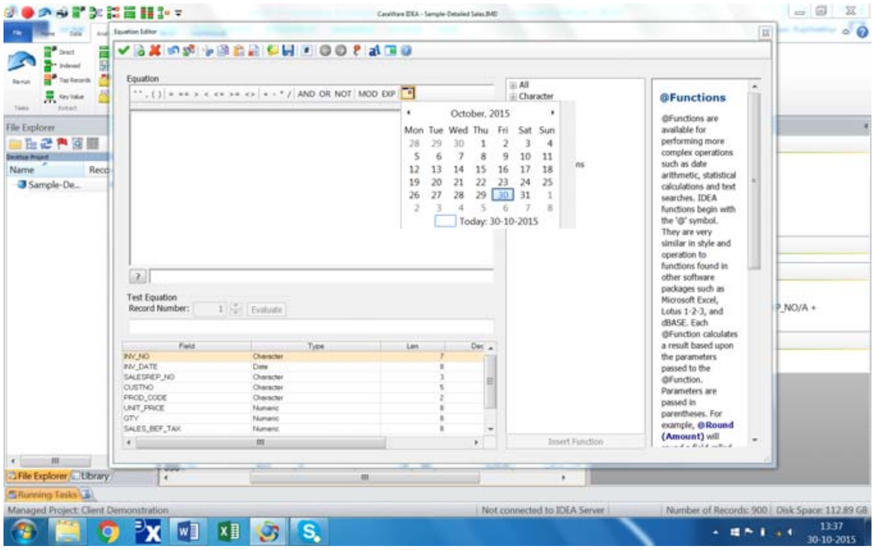
Sample Function - @len(PAN)<>10

Here the @ function is @len which displays the length of a character field. The Parameter / field is PAN. The constant is 10. The @function will allow the user to test the validity of the PAN for cases where length of PAN is not 10

The Equation Editor has a Validator which checks the equation or function for its structure and syntax. Equations which have an inherent error cannot be processed until the error is identified and duly de-bugged.

The Equation Editor in IDEA provides integrated online help for building equations and functions and provides easy-to-use practical tips and good practices while devising equations.

Character and Date fields in IDEA should always be in Parenthesis. While selecting Date constants within IDEA always use the Universal Calendar in the Equation Editor.

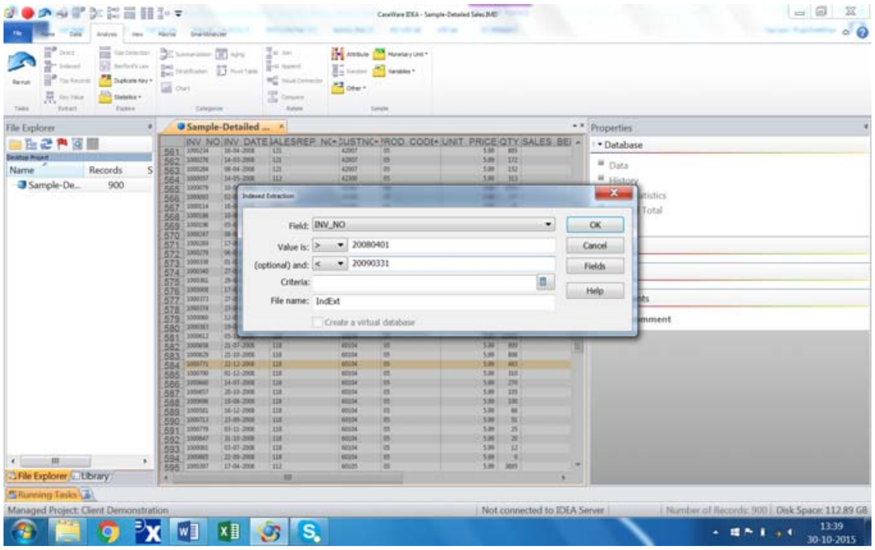


While using Character fields always sanitize the fields by using @ functions like @ strip to get rid off spaces and irrelevant punctuation marks (if the test so desires).

5.6.0 Types of Data Extractions within IDEA

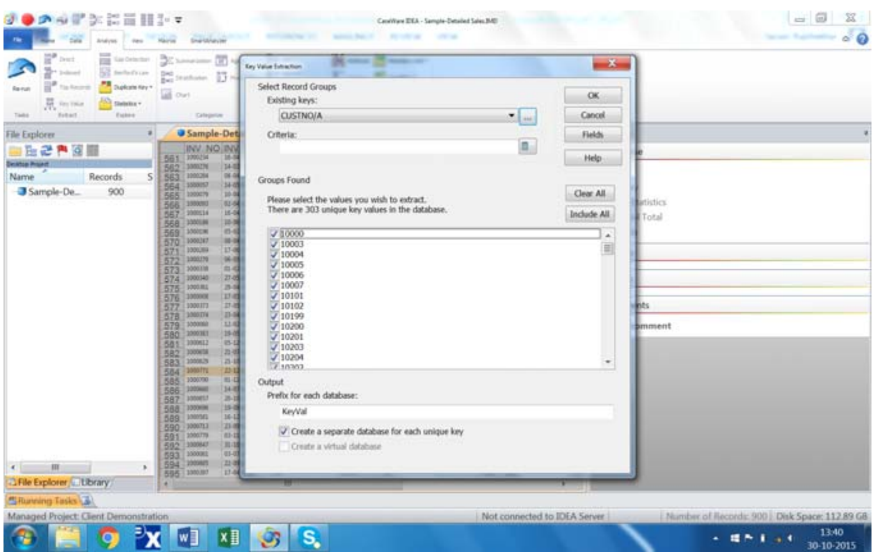
5.6.1 Indexed Extraction

Indexed Extraction allows you to limit the scope of data for which IDEA searches in the database. Indexed Extraction saves time when reviewing large databases. You can select an index for the search rather than have IDEA search through the entire database.



5.6.2 Key Value Extraction

Key Value Extraction allows you to generate a series of secondary databases quickly from common values found in the main database. Key Value Extraction does not require you to create equations to perform the extraction. A key is an index created for information in a database. A key value is a piece of information indexed by the key. You have the option of creating one output database or multiple output databases.



Key Value Extraction can be used to generate the bank branch wise report of high value cash transactions in newly opened current accounts. Here separate database will be generated for each branch to enable branch segmental reporting and oversight on potential suspicious money laundering indicators.

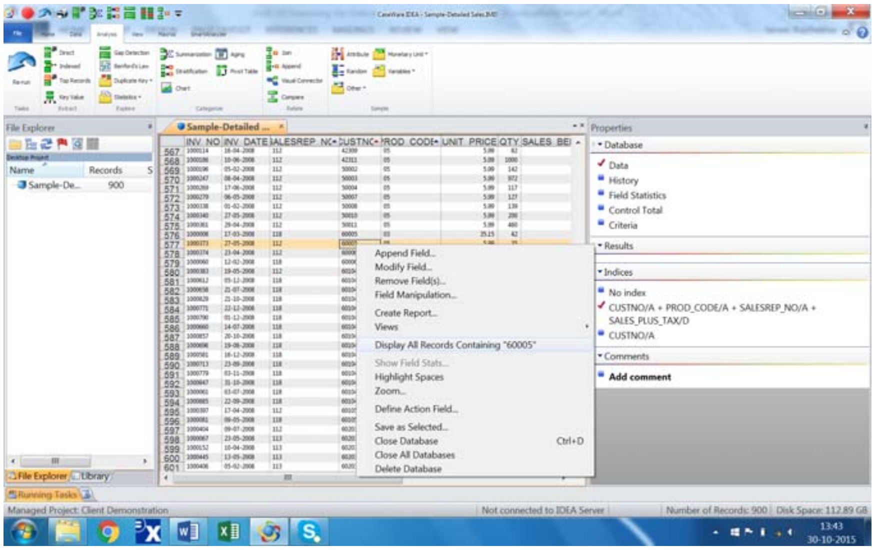
5.7 Display of Data

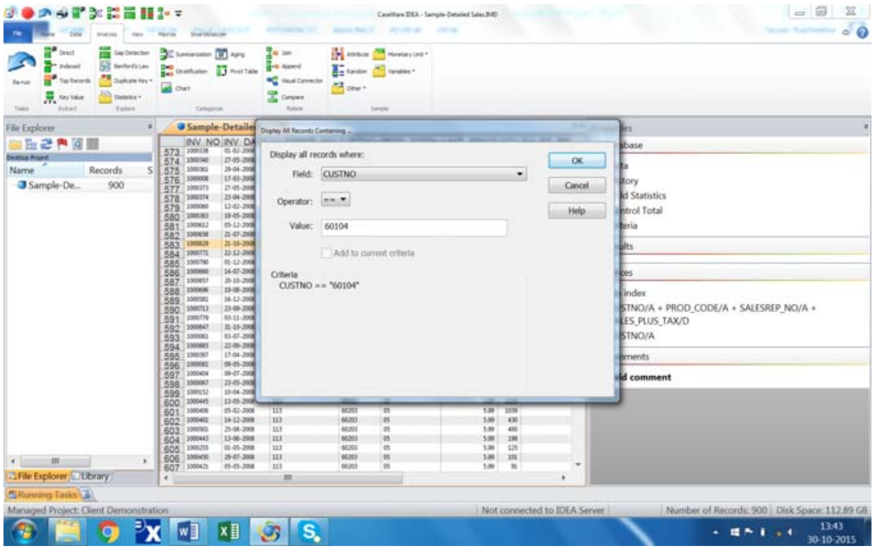
The Display of Data function assumes that the user is aware of the numeric value or character string to be interrogated/ isolated within IDEA. Here the user identifies the first occurrence of the value or string through the Find Function or through Visual Verification.

The user then goes on to display the data through the Display All Records Containing dialog box. (Seen Below)

Illustration: As an auditor you have received a tip-off from your internal source at the client's workplace that a given Employee (Say Emp. No. 287654) is indulging in excessive overtime claims. With IDEA you can display all overtime claims submitted by Emp No. 287654 and follow up the display with actual observation of his late-sittings and the productivity thereof.

The Display Data function can also be accomplished by writing an equation or function in the Criteria, seen on the right hand side of the screen under the Database Toolbar. Writing criteria is similar to writing criteria in the Direct Extraction dialog box seen in the section on Extraction of Data.





The Display Data function creates a temporary view of the database. This view appears as a result on the right hand side of the screen. The view can be saved by invoking the Save As option in File in the Menu Tool Bar.

Illustration: The Criteria can be used to identify all blank address fields in a customer master file to review the authenticity of the customer.

Chapter 6

Identifying Exceptions and Irregularities

6.0 Introduction

IDEA includes two key functions to identify exceptions, irregularities, anomalies and errors.

These are:

- Duplicate Detection
- Gap Detection

These functions assist the user to sift through large volumes of data and help pin-point specific duplicate entries or specific missing entries. These also help the user obtain an assurance on all the data reviewed by it. The duplicate or missing items identified can be taken up for tests of transactions and balances (substantive tests) after running the respective duplicate and gap tests within IDEA.

Duplicate Detection and Gap Detection tests are standard tests which are applied to every database to study the completeness and accuracy of its contents. The tests do not require much querying experience and resemble plug-n-play tests through standard Graphical User Interface (GUI). Both tests run largely on formatted sequential data fields like Invoice Number, Purchase Order Number and Indent Number as examples.

The user needs to evaluate the need for such tests, depending on the functional control being tested within the process. For example, if the user is reviewing financial controls within ERP-SAP R/3 for Procure to Pay, the chances of the system generating duplicate Purchase Order Numbers or missing numbers is quite improbable. But if the user would like to test for duplicate Vendor Bills submitted (an external document), which is a non-system generated field, the duplicate test could be employed.

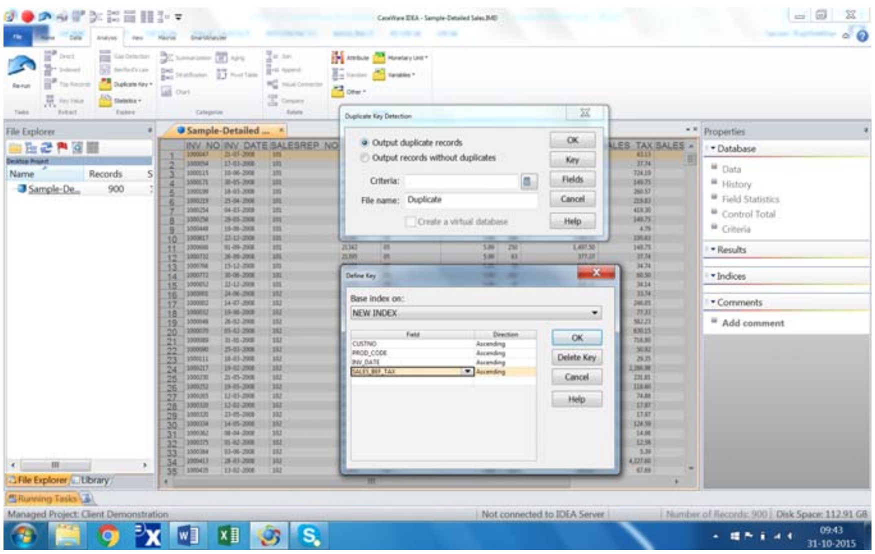
6.1 Duplicate Detection Test

Duplicate Tests within IDEA include Detection tests and Exclusion tests.

Duplicate Key Detection includes identifying Duplicates and identifying Non-Duplicates items within the active database.

6.1.1 Duplicate Key Detection – Output Duplicate Records can run on any normal database which is active within IDEA. The test allows IDEA to look for duplicates in and up to 8 fields. The fields may be numerical, date or character based.

The fields on which duplicate tests have to be run are entered as 'Key' fields within the Duplicate Key Detection dialog box. (See Below)



IDEA will look for duplicate cell-items in a combination of the Key fields selected.

Illustration: If a user is looking for duplicate sales invoices raised on customers, and if he enters the Invoice No., Customer No., and Invoice Amount as the Key fields, IDEA will look for duplicate sales made to the same customer for the same amount against the same invoice number.

The Duplicate Key Detection dialog box also offers the flexibility of Criteria.

The Criteria allow the user to look for duplicates on a filtered part of the active database and not on the whole active database.

Illustration: If a user is looking for duplicate sales invoices raised on customers, and if she enters the Invoice No., Customer No., and Invoice Amount as the Key fields, IDEA will look for duplicate sales made to the same customer for the same amount against the same invoice number. In *this case, the user can enter criteria wherein IDEA will look for duplicates only for a particular product code or a particular business division of the group. Here the search is more specific and meaningful.*

The user can also choose IDEA to display certain fields in the exception output database through the option Fields. This is relevant when the database being investigated has a large group of fields like master tables. The user may like to specifically report on duplicate items in a few relevant fields like Bank Account Number, PAN Number. Here she can use the Fields option.

6.1.2 Duplicate Key Detection – Output Records without Duplicates follows the same mechanism as the Duplicate Key Detection – Output Duplicate Records test in terms of Key fields, Criteria and Fields. The only difference in the two tests is that in the Duplicate Key Detection – Output Records without Duplicates looks for unique items in the database i.e. non-duplicates. This test will exclude all the duplicate items.

The Duplicate Test within IDEA has been improvised significantly over the years in terms of speed and performance. The test generates exceptions on large databases in record time. The test is also user-friendly as the user does not have to separately sort the database or write a vertical cell-difference function (like in MS-Excel) prior to test-run. One click of a button in the Duplicate Key Detection dialog box takes care of the sort and vertical cell-difference function automatically.

6.1.3 Duplicate Key Exclusion – Output Duplicate Records where one Field is Different

The duplicate key exclusion function is similar to the duplicate key detection function in the sense that the test allows IDEA to look for duplicates in and up to 8 fields. The fields may be numerical, date or character based. In addition, Duplicate Key Exclusion offers the framework where one additional field other than the 8 matching fields may be different.

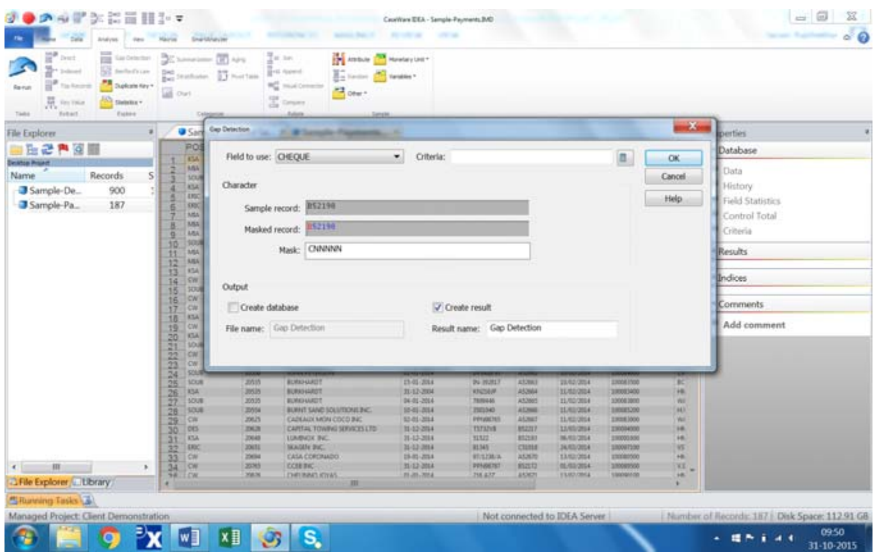
Illustration: In concurrent audits of Registrar and Transfer Agents (R&T), the AMC (Mutual Fund) pays the R&T, brokerage as entry and exit loads where the units have been purchased or sold through a SEBI registered broker. Here specific brokerage rates are fixed for specific mutual fund schemes. Moreover, brokerage is not paid where the transaction is done through a direct dealing popularly called 'ARN-DIRECT" in the Broker Code or where transactions are over Rs. 5 crores. Here, through Duplicate Key Exclusion, the auditor can identify incorrect / varying brokerage rates paid for the same mutual fund scheme for the same broker code.

Duplicate Key Exclusion is popularly called De-Dup and is used extensively for effective Know Your Customer (KYC) checks in the Banking and Financial (BFSI) sector and Commodity and Instrument Exchanges on large customer master files.

6.2 Gap Detection Test

Gap Detection within IDEA can be deployed on numeric, character and date fields.

Gap Detection can run effectively on numeric fields provided the fields are uniquely / sequentially formatted and represent the same logical field. For example, if gap detection is to be run on a Cheque Number field, cheque Number being numerically formatted, the test can be run for the entire cheque series or a cheque range. In most cases, multiple cheque series of varying ranges are used by organizations. In such instances the user can look for gaps in a specific cheque series range.



The user also has the express option of specifying a Gap Increment. This is true of a scenario where the numeric field number does not increment itself (through system or user design) by one. The increment is some number other than one like a Product Code which is in multiples of 1000, say 1000-1050, 2000-2050, etc. Here the Increment Range can be specified as 1000, so that IDEA looks for gaps within the actual number range (1000-1050) and not between various number range series (1050-.2000).

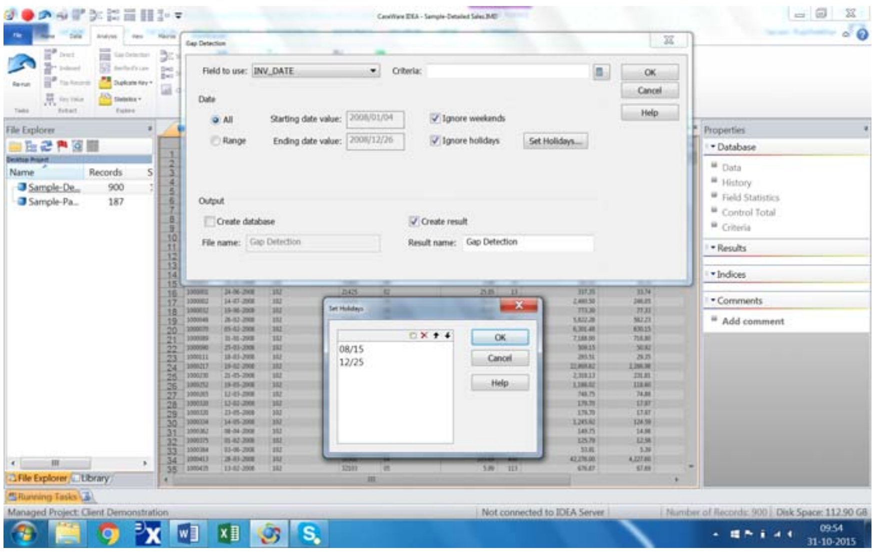
The Gap Detection dialog box also offers the flexibility of Criteria.

The Criteria allow the user to look for duplicates on a filtered section of the active database and not on the whole active database.

Illustration: If the user is looking for missing collection receipts issued to borrowers of a bank, and if she enters the Receipt No., as the Fields to Use, IDEA will look for missing receipts against instalment collections. In this case, the user can enter criteria wherein IDEA will look for gaps only for a particular loan product code or for a particular zone or collection agent. Here the search is more specific and meaningful.

Gap Detection can run efficiently on date fields also.

Gap Detection on date fields follows the same mechanism as numeric field gaps, in terms of date ranges and Fields to use.



IDEA offers the user the flexibility of excluding Holidays and Weekends from the gap testing for dates to identify genuine date gaps and not gaps attributed to holidays and weekends. Holidays can be set by enabling the option Set Holidays. Public holidays in a financial year may be entered herein. Weekends by default follow the universal weekend route of Saturday and Sunday.

The default universal weekends can be changed by altering the Weekend Settings under Date Options under File – Options – System Tab. This is relevant for organizations operating in geographical locations like the Middle

Eastern and Gulf States where weekends are normally Friday and Saturday.

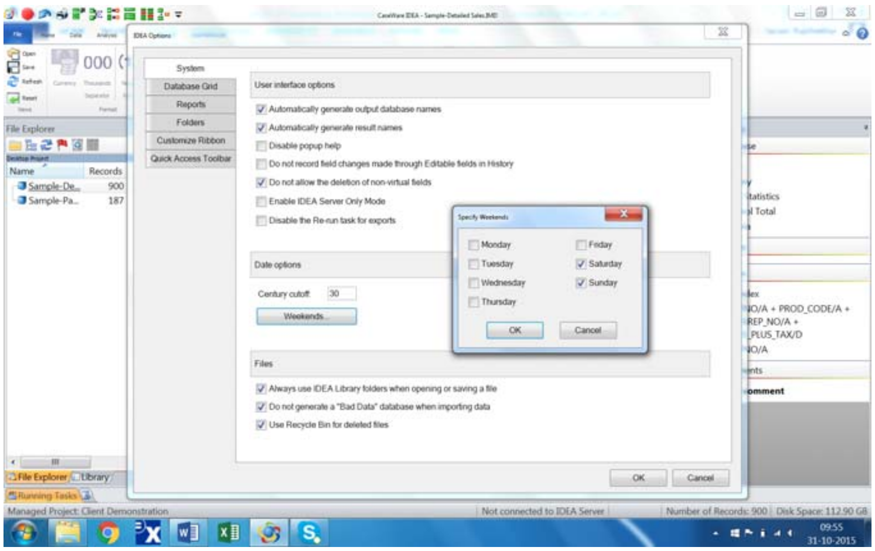
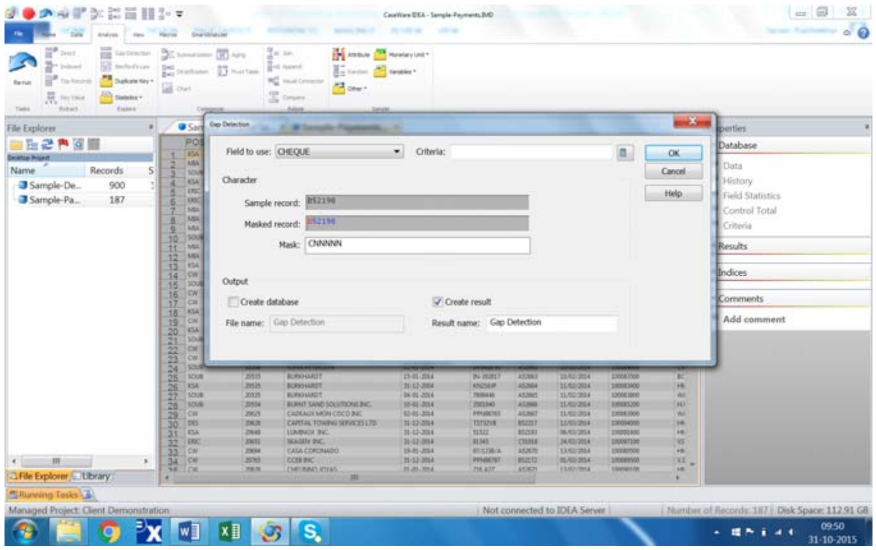


Illustration: the production server backup for an online e-tailer has to be taken every week-day on to a backup server as part of a Business Continuity Plan for the e-tailer. The backup history log can be imported into IDEA and Date Gap Detections can be run on the log to identify missing back-up dates (excluding Holidays and Weekends). This exception report will help to support the IS Auditors' assertion that controls are not being implemented regularly with regard to off-site data backups.

Finally **Gap Detections run well on Character fields** too.

The character fields in such cases need to be of the same length and must follow the same syntax. Character Gap Detection cannot run on Names or Addresses where the length of the field and Syntax does not follow any fixed pattern.

The perfect fields for for such tests are fixed length alpha-numeric fields like Employee ID's in a Payroll Data File. If the Employee ID is always formatted as A123456 for instance, where the syntax rule CNNNNNN works well, the Gap Detection detects missing employees within this combination of digits. This combination of digits is often called a MASK in IDEA.



The Gap Test within IDEA has been improvised significantly over the years in terms of speed and performance. The test generates exceptions on large databases in record time. The test is also user-friendly as the user does not have to separately sort the database or write a vertical cell-difference function (like in MS-Excel) prior to test-run. One click of a button in the Gap Detection dialog box takes care of the sort and vertical cell-difference function automatically.

Chapter 7

Profiling Data

7.0 Introduction

CAAT - IDEA can be used effectively as a Management Tool. It can play a vital role in assisting Management Teams to make key decisions by providing it with valuable, insightful and reliable analytical information on a near real time basis.

The Tool offers three key functionalities for Business Analysts:

- Summarization
- Stratification
- Pivot Table

Summarization - Allows the user to prepare and view a consolidated overview, i.e., a summarized position for specific fields as defined by the user. For instance, a Stock Broking firm dealing in the Options market can generate a Day-wise summary for Total Options Bought and Sold along with the Total Net Exposure for the day. The Firm can then go on to detect Net Exposures in excess of say Rs. 25 Lacs per day.

Stratification: Facilitates a range of band wise or interval wise based consolidation of data. For instance – sales made by an entity to its Customers through various Channel Agents can be stratified into value bands with respect to Channel Agents. Stratification will reveal sales made below the minimum floor price and above the ceiling price for different Channel Agents.

Pivot Table: it enables a cross-tab analysis of data. Here two data fields or more can be analysed against each other with respect to specific numeric fields. For instance, cross tab analysis can be performed on travel claims and travel policies of a company. Air travel rationalization can be performed to identify the most cost effective airline on fare basis to specific destinations, for a specific travel class and travel date too.

7.1 Summarization

Summarization accumulates the values of Numeric fields for each unique key. For example, summarizing an Accounts Payable database by supplier account number (the key) and totaling invoice amounts produces a database or Results output of outstanding liabilities by a supplier. The Summarization task provides:

- A list of unique items (keys) in the database
- The number of records/count for each key
- Totals of one or more numeric fields for each key

The user may select up to eight fields to summarize or if using one field can use the Quick Summarization option. By using multiple fields, the Summarization task works faster and is more efficient on a database that has been sorted.

Select Quick Summarization only if you are summarizing by a single field and if there are fewer than 32,000 unique values in the field.

The user may select any number of Numeric fields to total. The resultant database provides a list of unique key items and the number of records (NO_OF_RECS) for each key from where one can drill down to display the individual records for the key.

The user can select additional field data from fields from the first occurrence or the last occurrence.

The summarization result grid is limited to 4,000 rows and does not display results beyond that. If the result expected is more than 4,000 rows, then there is a need to create a database.

Summarization Examples

Example 1

The following database of cheque payments is to be summarized by Supplier with the payment amount being selected as the Numeric Fields to total and the Cheque Date selected as an additional field to include. The Statistics to include automatically appears as Sum.

Cheque Payments database to be summarized

SUPPLIER	CHQNO	AMOUNT	DATE
A100	348603	100	1995/12/31
A100	412591	500	1996/02/01
A100	311192	200	1995/09/04
B200	326319	300	1995/11/03
B200	521361	100	1996/01/04

The output database of summarized cheque payments shows three records for SUPPLIER A100 with a total amount of 800 (100, 500, 200). Although the date column displays 1995/12/31, it does not mean that there were three payments on 31/12/1995, but that this was the first entry in the cheque date field for the first record for SUPPLIER A100.

SUPPLIER	NO_OF_RECS	AMOUNT	DATE
A100	3	800	1995/12/31
B200	2	400	1995/11/03

Example 2

Stock items to be summarized

WH	STOCK_NO	QTY	PRICE	VALUE	DATE
NW	X-345-886	25	10.00	250.00	1996/06/30
NW	X-345-886	2	10.00	20.00	1996/09/15
NW	X-345-886	10	11.50	115.00	1996/11/03
NP	P-666-188	2	4.90	9.80	1996/12/02
NP	P-666-188	1	5.65	5.65	1996/10/08

To summarize the above database of stock items by stock number (STOCK_NO) within each warehouse (WH), we would select:

Fields to summarize: WH then STOCK_NO

Numeric fields to total: QTY, VALUE

Other fields to include: PRICE, DATE

The two tables below show the different resultant databases created, depending on whether we have selected the (other) fields data from the first or last occurrence for each key.

First Occurrence

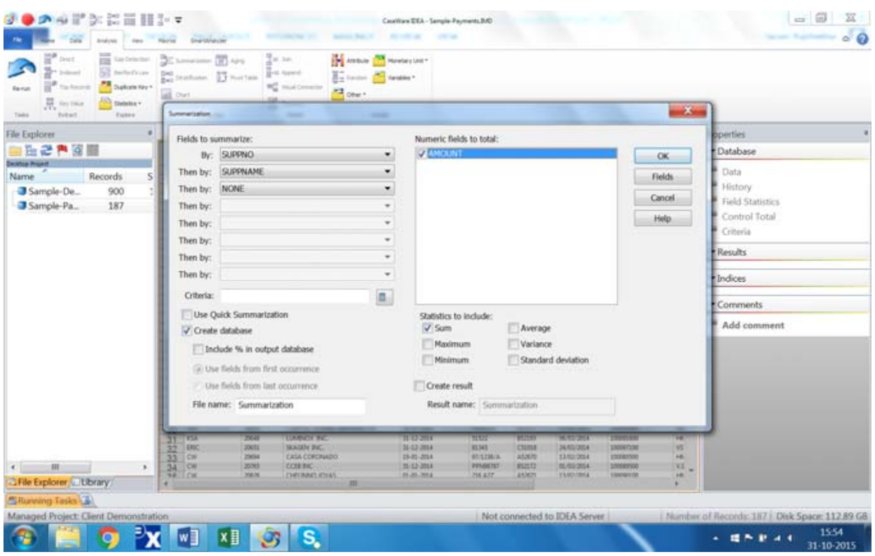
WH	STK_NO	NO_OF_RECS	QTY	PRICE	VALUE	DATE
NP	P-666-188	2	3	4.90	15.45	1996/12/02
NW	X-345-886	3	37	10.00	385.00	1996/06/03

Last Occurrence

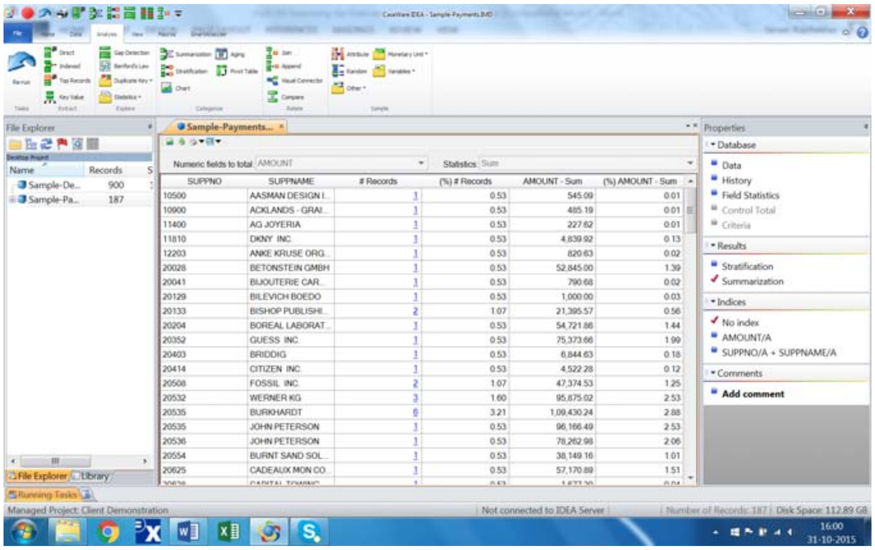
WH	STK_NO	NO_OF_RECS	QTY	PRICE	VALUE	DATE
NP	P-666-188	2	3	5.65	15.45	1996/10/08
NW	X-345-886	3	37	11.50	385.00	1996/11/03

The choice of first or last occurrence of (other) fields makes a difference to the PRICE and DATE field values. In this example, this information could be misleading. So one has to take care while selecting (other) fields. The (other) fields selected, in this case PRICE and DATE, do not affect the calculations, they are simply either the first or last occurrence of data for that field associated with the key. Therefore, in the examples given above, QTY + PRICE do not decide the value and are only meant for display and reference.

Typical view of the Summarization dialog box in IDEA is –



Typical view of the Summarization result in IDEA is –



7.2 Stratification

The process of stratification involves creating bands based on ranges of values (normally from the minimum to the maximum values of one or more fields) and grouping the records from the database into the appropriate bands. By totaling the number of records and value of each band, one can gain insights into the profile of the values in the database. The stratification analysis is also useful for determining high and low cut-off values for testing exceptional items - outliers.

With the Stratification task, one has the option to create both a database and a result with the option to include details of the band range within which each record falls. One can then use this database for further analysis.

Stratification produces a numeric stratification analysis for each key in the database. For example, one could produce a profile of sales for each salesperson. This task can potentially create an extremely large volume of output. Therefore, there is an option to specify low and high cut-off values to restrict output to only the keys required, the keys whose total value of transactions is between the specified range are output.

One needs to remember that IDEA is case sensitive when processing data, therefore, one has to consider the case when specifying stratification bands. (If data has been input inconsistently?), such as in both upper and lowercase, we have to create a Virtual field with @upper or @lower to convert all text to the same case, and then stratify the Virtual field.

To include all items in the stratification analysis, the bands should start less than the minimum value and the upper band greater than the maximum value of all fields.

When stratifying a Numeric field, before entering the stratification bands we have to use the Field Statistics task to know the minimum and maximum values of the field(s) being stratified as well as the skewness of the values. These values help to determine the interval start and end values as well as the increment size.

Date and Character stratification are different than Numeric Stratification in the sense that different fields are totaled to the one used for banding.

Stratification Examples

Stratifying a Numeric field

When auditing a payments system, we can identify invoice splitting just below authorization by stratifying the payment amounts. We have to make sure to specify a narrow band below each limit.

A higher count and value of total payments just below the authorization limits would indicate that invoice splitting has potentially taken place.

Stratifying a Character field

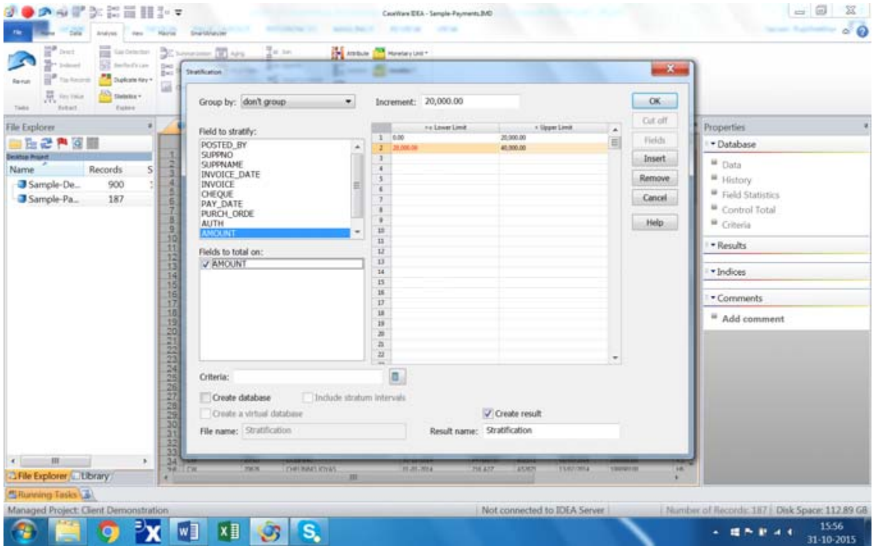
Sales data can be analyzed by Business code area. We have to specify the Business Code field as the Character field to stratify and enter the band using as many characters of the code as are required to identify the area.

Stratifying a Date field

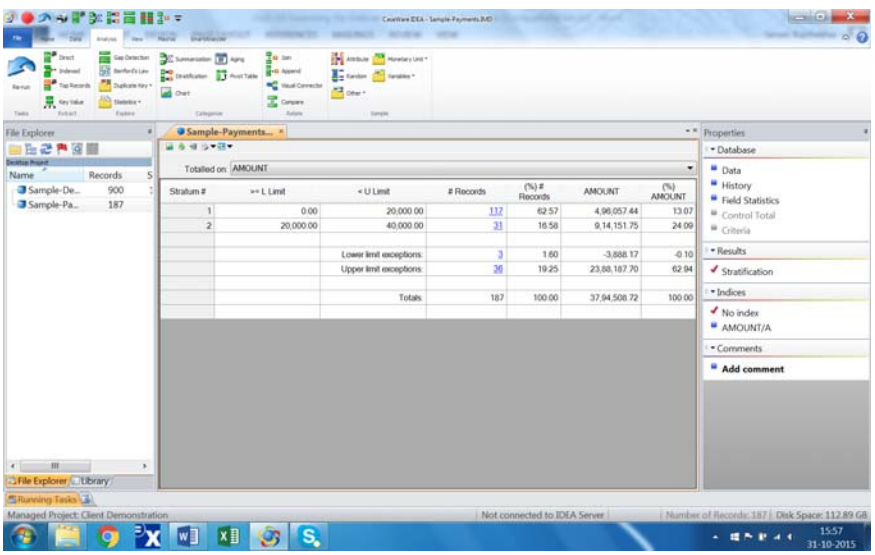
We can analyze employees by their date of birth to get an age increment profile, or we can analyze fixed assets by date of acquisition.

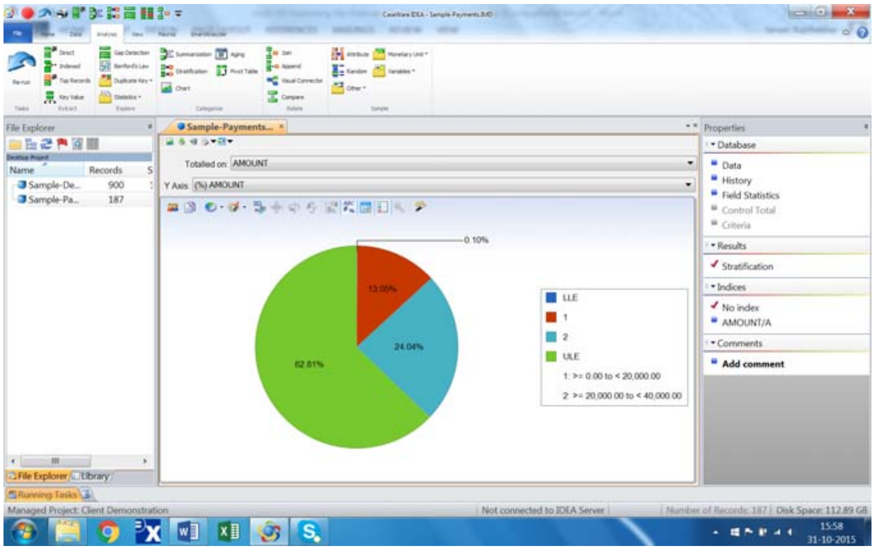
Stratification is the first stage process in identifying the classification of values in any population from High (A Class) to Low (C Class). This functionality within IDEA when used in conjunction with "Field Statistics" gives the user a true sense of the dispersion of values in a normal database. The user is able to easily appreciate the famed "80-20 Rule of Business" i.e. 80% value accounted for by 20% records and 20% value accounted for by 80% records". Once the stratum are created, the user can then employ "Stratified Random Sampling" where complete assurance can be sought on the confidence interval within the population.

Typical view of the Stratification dialog box in IDEA is –



Typical view of the Stratification result in IDEA is –





7.3 Pivot Table

The pivot table is an interface where the user has the freedom to define how the data is to be displayed and organized.

When creating a pivot table in IDEA, we use the Pivot Table Field List dialog box to choose the fields to be included in the pivot table. The position on the pivot table defines how the data is organized and summarized.

Once the fields are in position, the user can move them about easily by dragging and dropping. One may even drag a field right off the window to remove it.

The user may summarize the numeric data in many ways, by selecting from the Pivot Table Field List dialog box. One can configure the pivot table to show record counts, totals, averages, or minimum/maximum values. Color code summary fields can be assigned.

The records represented in any cell are visible by selecting the Drilldown Data button on the Pivot Table toolbar to display an Extraction Preview window from which the data may be saved to another database or printed.

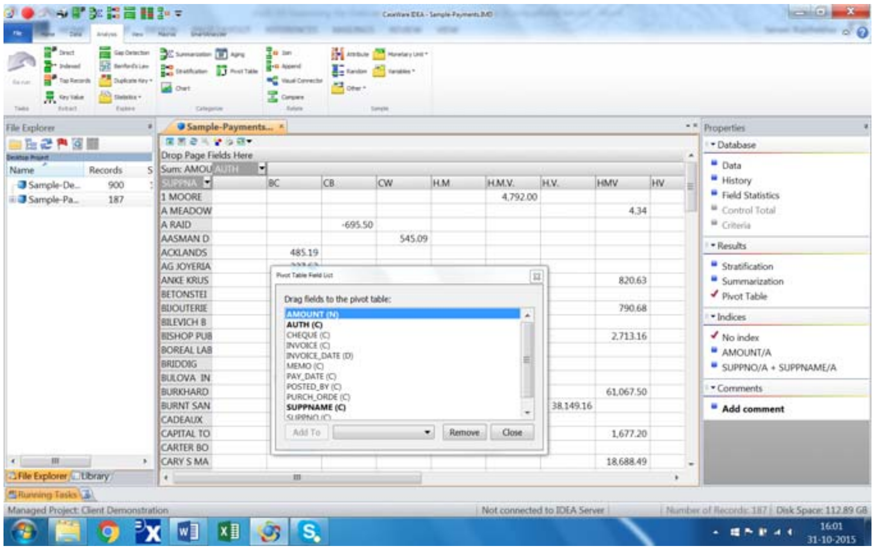
The user can create, record and save an IDEAScript relevant to the pivot table by using the IDEAScript button on the toolbar in the Pivot Table result.

The pivot table is capable of holding 50,000 rows x 2,000 columns within IDEA.

After creating the Pivot table the following options are available:

- Change the summary statistic for data fields.
- Export the displayed portion of pivot table to Excel and / or an IDEA database too.
- Change the sort order.
- Show/hide footer.
- Change the background colour of cells in the Total rows and columns.
- Show Top Records.
- Create IDEA Script code based on the pivot table.
- Add a new field to a pivot table.
- Delete a field from a pivot table.
- Filter displayed field items.
- Expand or collapse fields.
- Drill down field data.
- Print the pivot table.

Typical view of the Pivot Table dialog box in IDEA is –



Pivot Table Examples

In the General Insurance industry, motor premiums and claims form a crucial part of the underwriting business of a company. Historically, motor claims have always proved to be a loss making unit of a company due to the sheer volume of claims with premiums not following pace with the escalation in claims. Recently a leading General Insurance company undertook a novel exercise within IDEA where the motor claims admitted and paid for the last 5 years were analysed within IDEA's Pivot Table. The vehicle number of the insured was captured in the Pivot Row and the Claim Admitted for the last 5 years was captured in the column. Vehicles making repetitive claims were identified through this activity.

Chapter 8

Working with Multiple Files

8.0 Introduction

Just as the extraction of data into IDEA from a client's application system is imperative for the use of IDEA in the same manner tasks within IDEA can be optimally performed only with the help of multiple file operations and activities.

Even though data is getting centralized all over the world, it is distributed in data tables and schemas. A single buy in ERP-SAP for instance will affect data tables in FI and MM modules.

The CAATs auditor, with the assistance of IT Support, can bring data from different data sources into IDEA. The independent data sources can be merged, compared, reconciled to prepare a single base file for testing in IDEA.

IDEA offers the following techniques for working with multiple files:

- Visual Connect
- Join
- Append
- Compare

These techniques allow the user to combine multiple files conveniently for analysis and review.

8.1 Visual Connector

Many databases created by large systems, such as SAP and Oracle, are composed of several tables. When importing those tables (using Open Database Connectivity - ODBC), they appear in IDEA as separate files or databases. In order to rebuild the relationships, one can use the Visual Connector.

Visual Connector allows the user to generate a single database from several other databases that share common or "key" fields. To create a visual connection between databases, we need to select a primary database, and then connect databases that contain matching records.

The relationship that the Visual Connector creates between the databases is one to many, meaning that the primary database can have several matches in connected databases. All records in the connected databases that match records in the primary database are included in the output database. For example:

Primary Database

Name
Mary
Bill

Database A

Name	Value
Mary	A
	B
	C
Bill	M
	N
Fred	G
	L

Output Database

Name	Value
Mary	A
	B
	C
Bill	M
	N

Database A contains several records that match records in the primary database. All matching records are included in the output database. Records in Database A that do not have matches in the primary database Fred - G, and Fred - L are not included in the output database.

Illustration of the Visual Connector

Let us select the following three databases:

- Database 1 contains product codes, product numbers, and prices.
- Database 2 contains product codes and descriptions.
- Database 3 contains product numbers and regional sales.

We select Database 1 as the primary database, because the other two databases have at least one key common to Database 1. After we define visual connections between the three databases, the output database contains all records relating to the common keys in the three databases.

Database 1

PRODUCT_CODE	PRODUCT_NUMBER	PRICE
47	8000	6.29
48	6722	7.80
49	4467	3.29

Database 2

PRODUCT_CODE	DESCRIPTION
47	Cup
48	Glass
49	Bottle

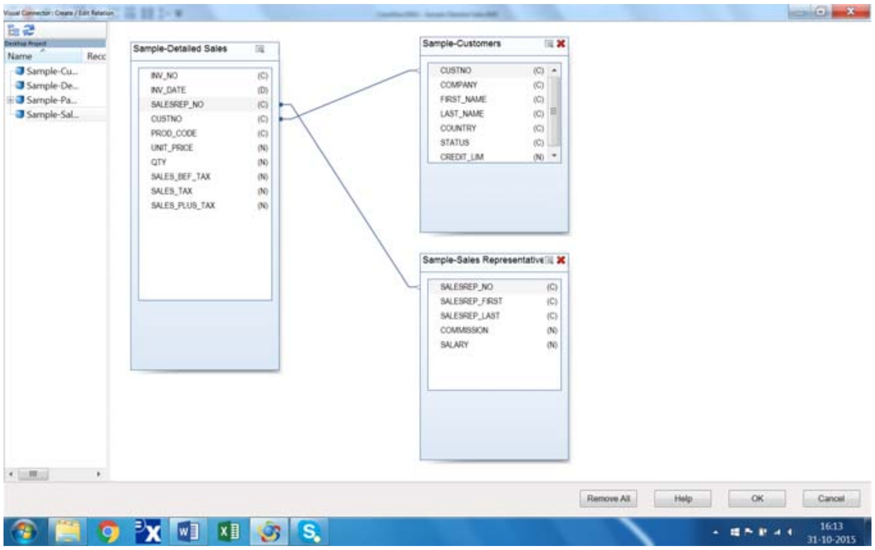
Database 3

PRODUCT_NUMBER	REGIONAL_SALES
8000	Toronto: 3700
8000	London: 5800
4467	Toronto: 6800
4467	London: 4400

Output Database

Information for PRODUCT_NUMBER 6722 is absent from the output database because Database 3 contains no information for PRODUCT_NUMBER 6722.

Typical view of the Visual Connector dialog box in IDEA is –



8.2 Join Databases

We can use the Join Databases to:

- Combine fields from two databases into a single database for testing.
- Test for data which matches or does not match across systems – looking for unreconciled items.

Tip

The Join Database runs considerably faster on sorted rather than indexed databases.

We can join or match databases only if they contain a common field (referred to as the "key"). The common field/s (up to eight in number) do not need to have the same name or length but they must be of identical field type. So while joining a bank book with the bank statement if bank book contains the cheque number as numeric and the bank statement contains the instrument number as character, the bank reconciliation join cannot take place till one of the fields are changed to unify the same.

We can join only two databases at a time.

- IDEA joins the secondary database to the primary database. We have to make sure to join the databases in the correct order.
- The databases to be joined must be in the same project folder in IDEA.

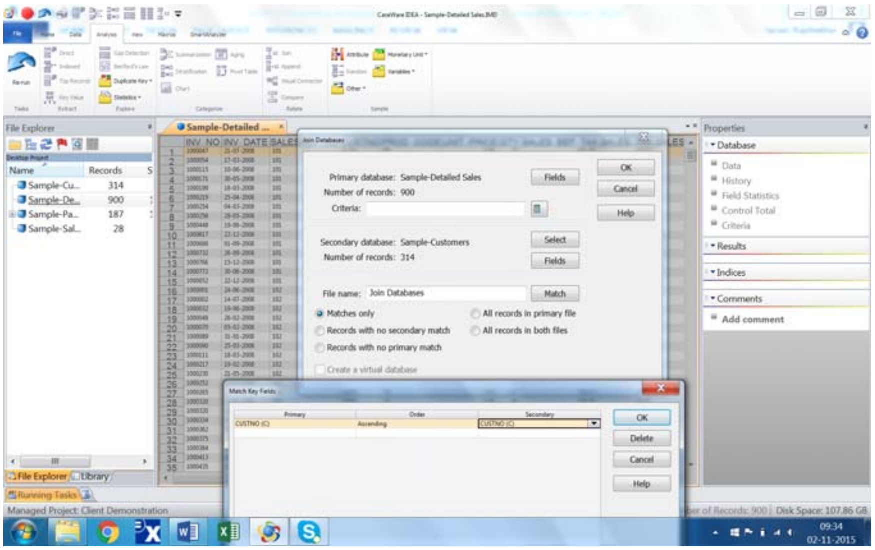
IDEA provides five join options. We have to ensure that we choose the correct option. .

- 'Matches only' provides the first match between two files like citizens having a PAN number who have filed their income tax returns for the last financial year.
- 'Records with no secondary match' provide records which exist in the primary file but are absent/missing in the secondary file, like citizens having a PAN number but who have not filed their income tax returns for the last financial year.
- 'Records with no primary match' provide records which exist in the secondary file but are absent/missing in the primary file, like citizens who have filed their income tax returns but do not exist in the PAN database for the country.
- 'All records in primary file' provides a representation of all records from the primary file with due matches from the secondary file and even unmatched primary file records, like sales to customers with matching orders or without matching orders.
- 'All records in both files' provide a combined representation of all records from the primary file with and without matches and all records from the secondary file with or without primary matches. This step is different from Matches only in that the Matches only looks for the first match, whereas all records in both files follow the one-way match technique.

The Join Database takes each record in the primary database in turns and looks for a matching key in the secondary database. It does not support many-to-many relationships. To support many-to-many relationships, we have to use Visual Connector.

When we join two Character fields with different lengths, IDEA pads the shorter field contents with trailing spaces.

Typical view of the Join dialog box in IDEA is –



8.3 Append Databases

We use the Append Databases to append or concatenate IDEA databases into a single database for audit testing. We can append or concatenate up to 32,768 databases, in the order selected, into a single database.

For example, by using Append Databases, we can append 12 monthly payroll databases to produce a database of all payroll transactions for a year. We can then summarize the database by employee to ensure that every employee has a count of 12 payroll payments in the year.

It is to be noted that when using Append Databases, we cannot append or concatenate:

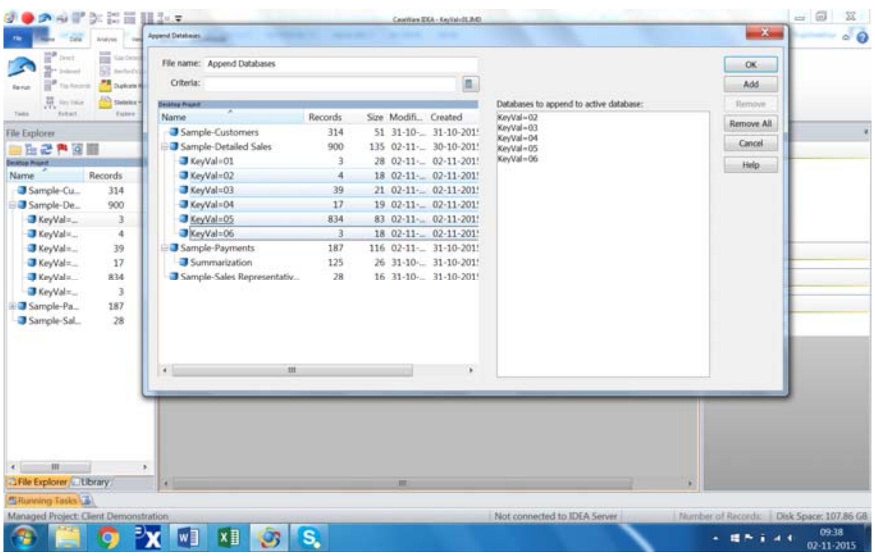
- Databases that contain one or more fields with the same name but different field types. So if the primary file has a field Transaction Number with character field type and one of the secondary files has the same field with numeric field type then the append would fail. Both the fields have to be of character or numeric field types.
- The same database more than once. So File A cannot be appended once again to File A.
- Databases from different locations: ensure all the files to be appended are in the same project folder.

Note

We have to retain each of the monthly archive databases and append them at the year end to produce a database of transactions for a year. Once databases have been appended or concatenated, we have to consider whether the individual databases are required, in addition to the appended or concatenated database. If they are not required, these databases can be deleted.

If we import the opening and closing balances databases into IDEA, we can reproduce the closing balances from the opening balances and transactions databases and compare them with the closing balances database. This test is the central theme of Financial Audits.

Typical view of the Append Databases dialog box in IDEA is –



Note

Append appears both under Analysis and Data in the IDEA menu tool-bar. To clear the use of the Functions Append under Analysis and Relate stands for Append Databases and Append under Data and Fields stands for Field Manipulation and Append.

8.4 Database Compare

We use the Database Compare to identify differences in a single Numeric field within two databases (referred to as the primary and secondary databases), for a specified common key.

The databases being compared need to be located within the same project folder.

Generally we compare the same database at two points in time, such as the Trial Balance at the beginning and end of the year to identify significant and material changes in each of the Heads of Accounts. We can also use it to compare a Numeric field from different systems (providing the key is the same), such as the quantity of inventory on the inventory master database versus the quantity of inventory on the stock take database (where the key is the item number) INVENTORY RECONCILIATION. Neither the key fields, nor the fields for comparison need to have the same names in the two databases, but the key fields must be of identical type. If necessary, we can change field types to allow the IDEA to match fields.

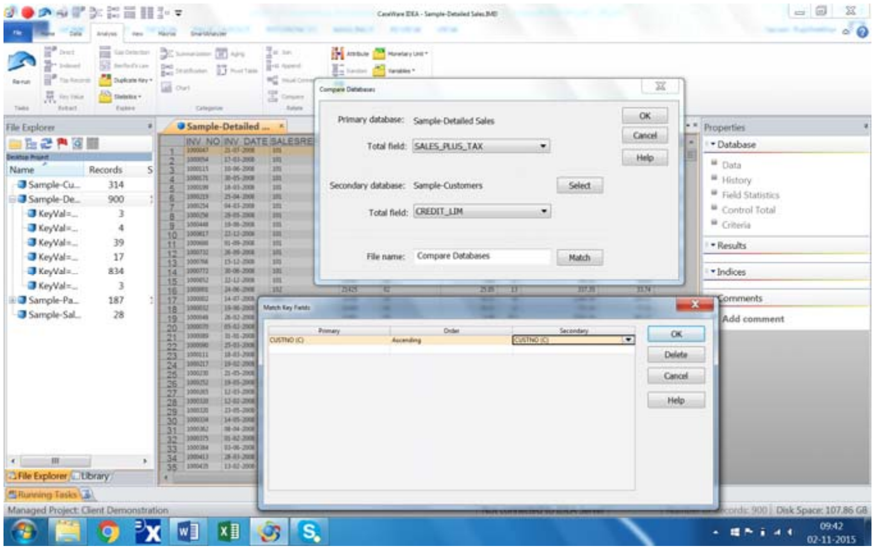
In statutory audits, File Compare can be used to view deviations in specific trial balance account heads between quarterly figures and annual unaudited figures. Where the deviations exceed 25%, the account heads under question can be taken up for detailed substantive testing. In fact, this procedure is at the center of the Audit Standard on Materiality.

As there may be more than one record for each key in either or both the databases, the Compare Databases task compares the totals for the Numeric field in each database (it essentially summarizes than compares the databases).

In addition to determining the differences in a Numeric field in a key, it also identifies the number of records there are in each database for each key.

If we need to compare more than one Numeric field, we can repeat the process for each field in turn.

Typical view of the Compare Databases dialog box in IDEA is –


Primary database: Sample-Detailed Sales
Total field: SALES_PLUS_TAX
Secondary database: Sample-Customers
Select
Total field: CREDIT_LIM
File name: Compare Databases
Match
OK
Cancel
Help
Primary
Order
Secondary
CUSTNO (C)
Ascending
CUSTNO (C)
OK
Delete
Cancel
Help
File Explorer
Name
Records
Sample-Cu...
314
Sample-De...
900
Sample-Pa...
187
Sample-Sal...
28
Properties
Database
Data
History
Field Statistics
Control Total
Criteria
Results
Indices
Comments
Add comment
Managed Project: Client Demonstration

Chapter 9

Sampling

9.0 Introduction

As a tool, IDEA has robust and stable capabilities to electronically audit 100% data in any application and database.

But CAATs do not replace traditional audit checks in terms of observation, inquiry, walk-through, substantive testing, and corroboration.

As an example – one can user IDEA to check manual adjustments to inventory balances for shortages and excesses. Armed with the list of such manual adjustments the auditor may want to check only a sample of such cases to ensure proper management authorization for the adjustments. This is where sampling comes into play.

Audit firms of international and national repute use IDEA primarily for sampling in their Financial Audits.

Sampling in IDEA is broadly statistical and probability-based.

The probability-based sampling techniques are:–

- Systematic
- Random
- Stratified Random

The statistical sampling techniques are:-

- Attribute
- Classical Variable
- Monetary Unit

9.1 Systematic Record Sampling

Systematic Record Sampling is a method to extract a number of records from a database at equal intervals to a separate database. It is often referred to as interval sampling.

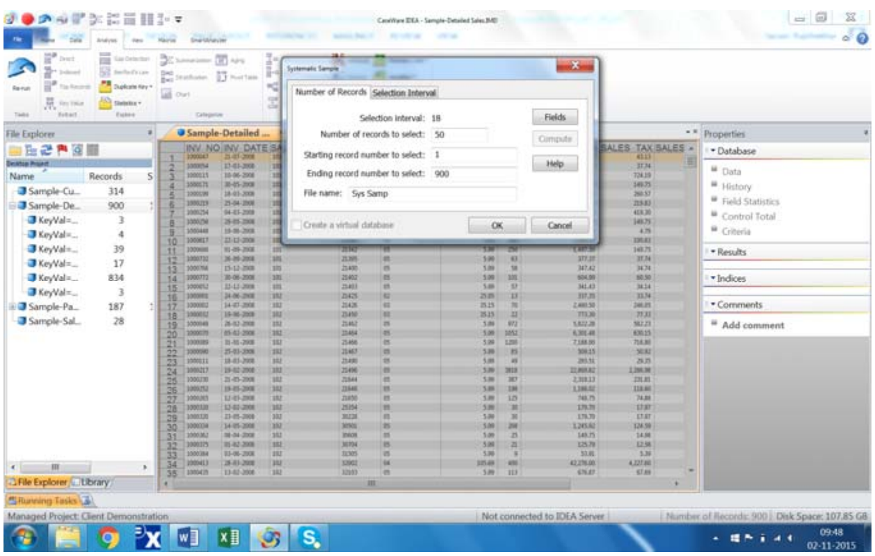
There are two methods of determining the sample:

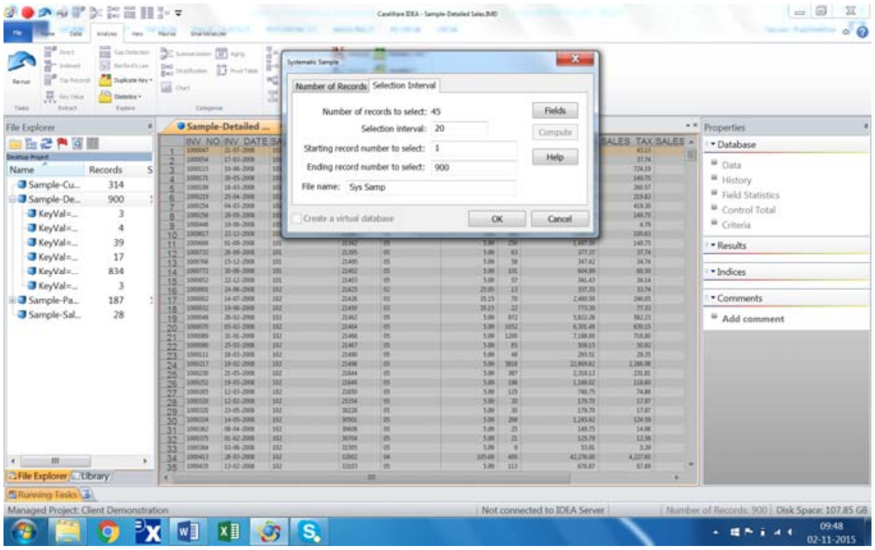
- Entering the number of records, in which case IDEA computes the interval size.
- Entering the selection interval, in which case IDEA computes the number of records.

IDEA calculates the above parameters on the number of records in the database and defaults to the first to last records. However, we can extract the sample from a range of records, if required.

Tip – Systematic sampling can be used where there is a need to cover samples evenly across the whole population, like in Payroll Audits. In Payroll activities the number of employees stays fairly uniform and even over a financial year. Here systematic sampling can be used for each quarter to cover different lots and intervals of Employees. Herein opinion can be expressed on a fairly representative lot of employees from the total employee base.

Typical view of the Systematic Sampling dialog box in IDEA is –



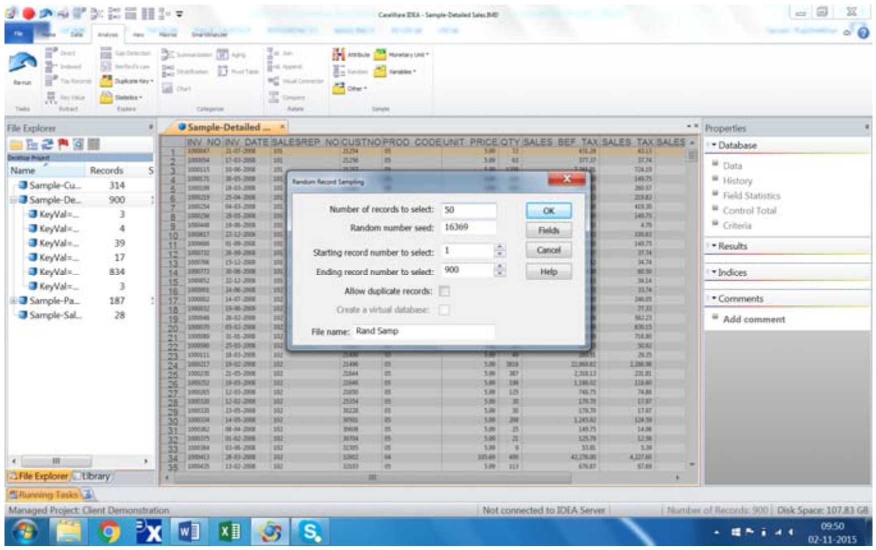


9.2 Random Record Sampling

Random Record Sampling is a commonly used method of sampling. With it we enter the sample size as well as the range of records from which the sample is to be extracted to a separate database. Then, using a random number seed, IDEA generates a list of random numbers and selects the appropriate records associated with these numbers.

> Random sampling is normally applied in every regular audit project regardless of a preliminary look at the nature of the data being audited. It should practically be applied where the data is normally distributed, in the sense that there are no artificial limits within the data or major troughs or spikes.

Typical view of the Random Sampling dialog box in IDEA is –



9.3 Stratified Random Record Sampling

Stratified Random Record Sampling is based on the materiality of transactions. Here due emphasis is given to the 80-20 rule in business, wherein 80% of value resides in 20% quantum and 20% value in 80% quantum. The 80% value representing the Class A is taken up for detailed scrutiny. The 20% value representing the Class C is taken up for limited random selection and scrutiny based on either rule of thumb or existing techniques based on business and data conditions.

Stratified Random Record Sampling may be used to extract a random sample with a specified number of records from each of a series of bands. This task requires then the database be first stratified into a series of numeric, character or date bands. IDEA then presents us with a table displaying the number of records within each band. We then enter the number of sample records we want IDEA to extract at random from each band.

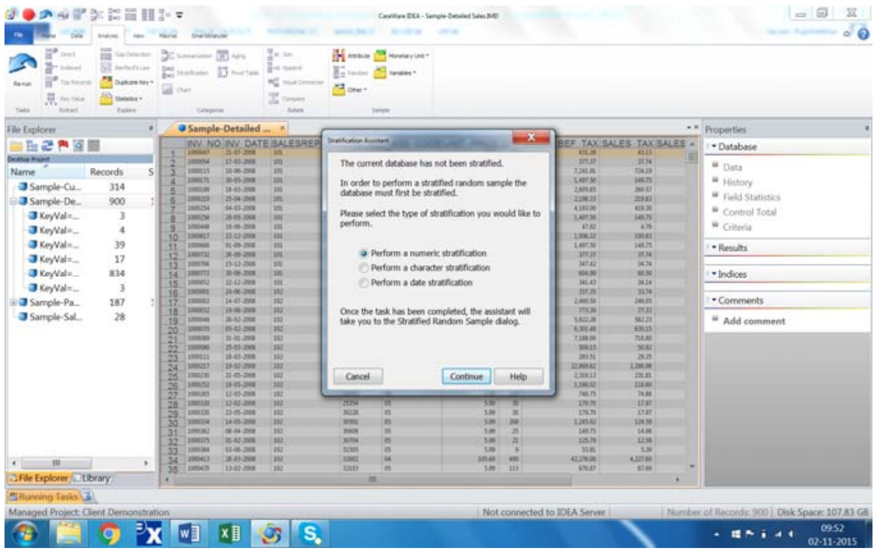
IDEA provides a Stratification Assistant to guide us through the steps for stratifying and generating the sample.

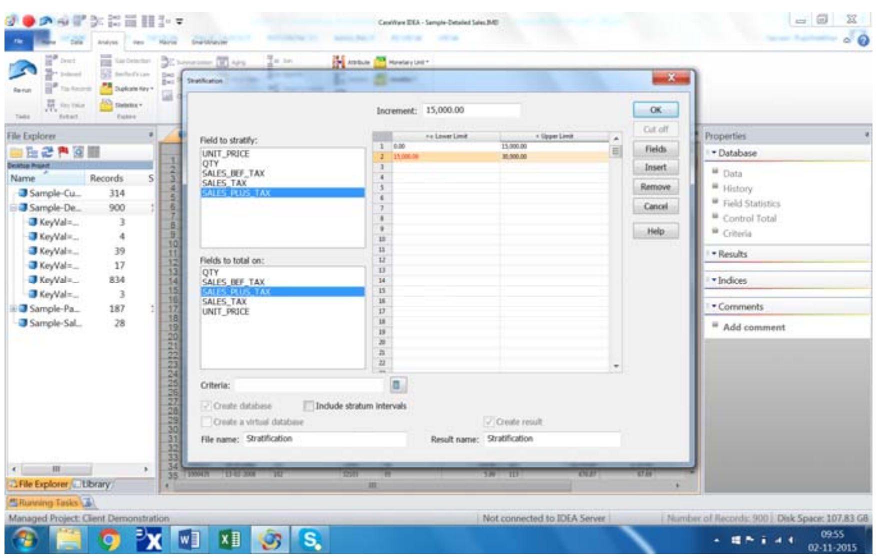
It is not necessary to stratify the file first, although if this has been done and a stratified database used, then the Assistant skips the initial stages and goes directly to the sample selection stage.

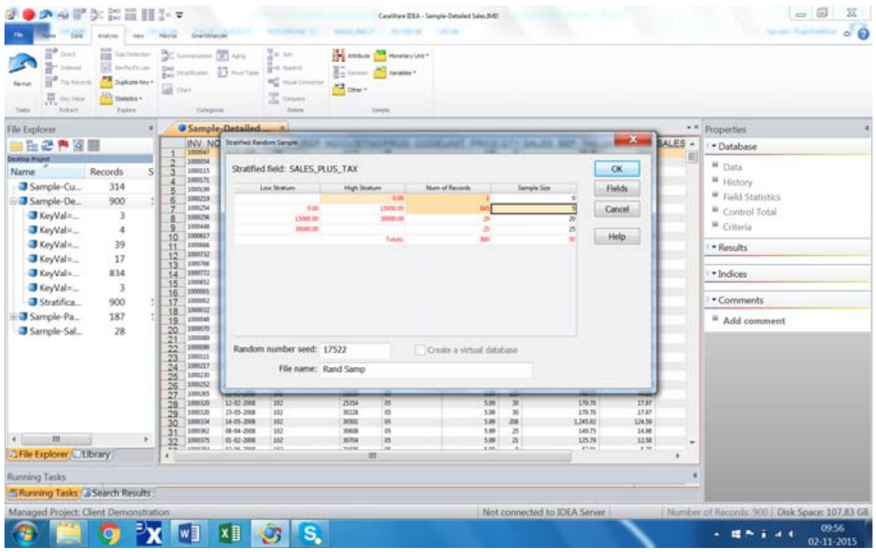
To ensure that we extract at least one transaction of each type in a random sample, we have to use Stratified Random Sampling: to stratify the database by the type field and then select at least one record from each band.

Stratified Random Sampling can be used in General Ledger Scrutiny as different values and risks are associated with different ledger items.

Typical view of the Random Sampling dialog box in IDEA is –





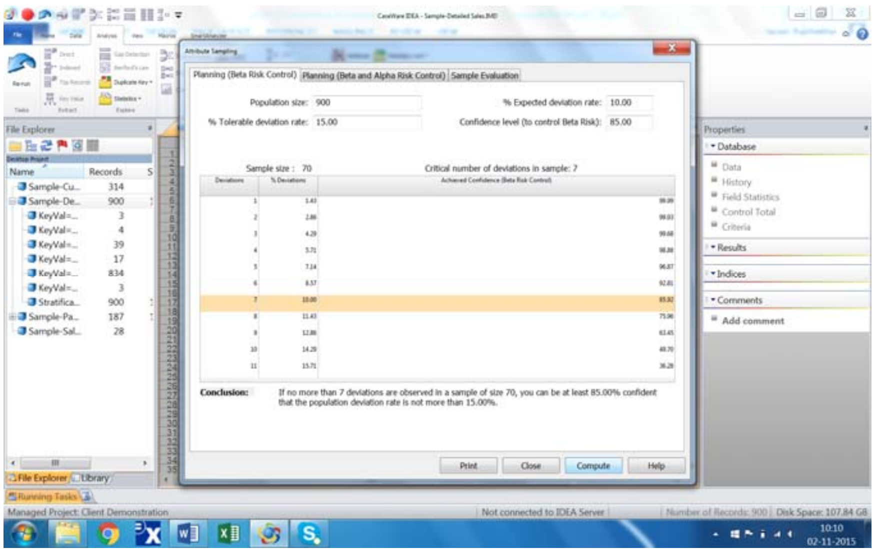


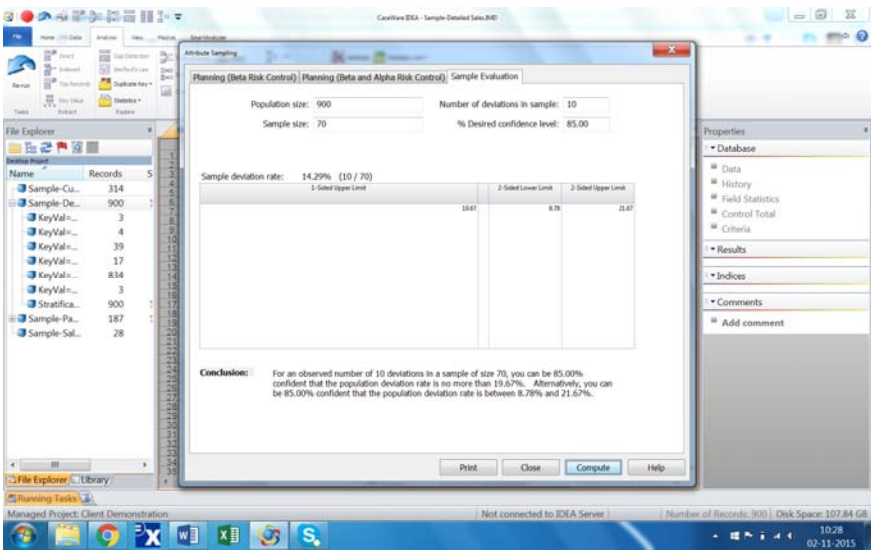
9.4 Attribute Sampling

Attribute Sampling refers to the examination of a subset of a population (a sample) in order to assess how frequently a particular event or attribute occurs in the population as a whole. An attribute has only two possible values: true or false. In auditing, typical attributes are whether an item is in error or not, whether a particular control has been exercised or not, or whether the entity complied with a particular law or not.

The Attribute Planning and Evaluation task is meant to plan attribute samples and evaluate results. The Planning dialog boxes are used to determine the minimum sample size required to meet specified audit objectives, as well as the critical or maximum allowable number of sample deviations in the sample.

The two Planning options, Beta Risk Control and Beta and Alpha Risk Control, allow us to (control or cover) the risk of one or two types of incorrect decisions. Auditors largely set only the risk of relying on controls when they are not effective (Beta Risk) and the tolerable deviation rate. To use this approach, we select the Planning (Beta Risk Control) tab in the Attribute Sampling dialog box.





Example - An internal control requires that all checks over $5,000 be signed by a manager. Whether the check associated with a transaction has been signed by a manager is an example of an attribute of the transaction, it has a value of true or false. Attribute Sampling can be used to determine whether the control is operating properly, to some specified level of confidence or reliability, without examining every $5,000 check in the population. The population in this example consists of all checks over $5,000. A subset of this population, called the sample, is selected for examination. Based on the number of control deviations (unsigned checks) found in the sample, inferences about the number of control deviations in the population as a whole can be made.

One of IDEA's two Sample Planning tasks can be used to determine the minimum sample size necessary to infer that the percentage of control deviations in the population is no more than a specified tolerable deviation rate, while controlling either the level of Beta Risk or the levels of both Beta and Alpha Risks. Roughly speaking, Beta Risk is the risk of incorrect acceptance, while Alpha Risk is the risk of incorrect rejection. The most crucial of these in audit situations is Beta Risk.

Here is an example of planning to control Beta Risk, since this is the most common situation. Suppose the auditor wishes to calculate the sample size using the following parameters:

- Population size (number of checks over $5,000): 10,000
- Expected deviation rate (expected percentage of checks over $5,000 that are not signed by a manager): 2%
- Confidence level (to control Beta Risk): 95%
- Tolerable deviation rate (maximum allowable percentage of checks over $5,000 that may be not signed by a manager while still allowing reasonable reliance on this control): 5%

Using these parameters, the required sample size is 180, with a critical number of sample errors equal to 4.

Planning to control both Beta and Alpha Risks would proceed in the same way, except that a second confidence level, equal to 100% - the maximum allowable Alpha Risk, would be specified as well. For example, with confidence to control Alpha Risk = 95% , lower threshold deviation rate = 2%, and all other parameters as above, the required sample size is 382, and the critical number of deviations is 12.

Suppose only Beta Risk is controlled, so the sample size of 180 is used. A sample of 180 items is selected from a population of checks over $5,000 (such as may be obtained by performing a Direct Extraction using Random Record Sampling, and the true number of errors or deviations in the sample is determined through audit procedures. Next, Attribute Sample Evaluation is used to determine the upper limit on the number of population deviations to a specified confidence level. Suppose the true number of deviations is equal to 4. Then, in the Sample Evaluation dialog box, enter:

- Population size: 10,000
- Sample size: 180
- Number of deviations in sample: 4
- Confidence level: 95%

The one-sided upper limit on the population deviation rate is calculated to be 4.98, which is just under the tolerable deviation rate specified when the sample was planned. This indicates that the sample size was efficient. If, on the other hand, five errors are found, the upper limit is 5.72, which exceeds the tolerable deviation rate. If only three errors are found, the upper limit is 4.22. IDEA also calculates a two-sided interval, delineated by a two-sided upper limit and a two-sided lower limit.

IDEA's Attribute Sampling programs are used to plan for and evaluate samples obtained by Random Record Sampling. If the sample was obtained by Stratified Random Sampling, then we do not use this evaluation program; instead, we seek the guidance of a statistician.

"A detailed Technical Specification on Attribute Sampling – CaseWare IDEA Research Department Document of release date June 17, 2003 has been reproduced in Appendix III for the benefit of the academically inclined readers. This document describes the mathematical procedures followed by IDEA in planning and evaluating Attribute samples."

9.5 Monetary Unit Sampling

A numeric amount field is the basis of the sample in Monetary Unit Sampling, in which the monetary unit is sampled. Having chosen the monetary units to test on an interval (or random item within each interval) basis, the items which contain these units are identified. To do this, it is necessary to calculate the cumulative value of the relevant field for each item.

Record	Ref	Amount	CUMULATIVE
1	A123	50	50
2	A124	75	125
3	A125	40	165

With an interval of 100 starting at a random start point of 35, the first selection would be at 35, and the second at 135. It is the cumulative field that is used for the selection.

Any items greater than the sampling interval are to be chosen by the Auditor. However, these key or high value items have to be extracted from the population and subjected to separate testing, leaving the items below a specified high value amount (which can, if desired, be set to a different limit to the interval) to be evaluated statistically.

IDEA offers options for determining the sample size for both compliance and substantive tests, based on confidence levels, population size, materiality and certain options on the weighting statistics used.

Whether or not the planning option is used, samples can be selected using the Monetary Unit Sampling techniques.

The output file contains an Editable field called AUDIT_AMT. If desired, this can be updated from the results of audit testing by using the Field Manipulation task and then a detailed evaluation of the results can be performed against book value.

The Monetary Unit Sampling technique is used extensively by the Comptroller and Auditor General of India.

Monetary Unit Planning - Examples

You wish to sample from the salary data contained in the Sales Representatives file. You wish to sample from both positive values and negative value. You have the following information for your planning:

- Confidence level – 90 percent.
- Tolerable error - $200,000. In auditing, this is often known as the materiality amount.
- Expected error in the population - $40,000.

Open the dialog box for Monetary Unit Sampling - Plan. Enter the information as indicated below. IDEA will compute the sampling interval and sample size in the dialog box. If you are ready to select the items, you may do so directly from the dialog box using the Accept button.

Note that the sample size is only an approximate value since the effect of high value items on the sample size is not known.

- Use value from database field: SALARY
- Absolute values selected
- Confidence level : 90 percent

- Tolerable error: 200,000
- Expected error: 40,000
- No change to basic precision

Estimate

- Approximate sample size : 15
- Sampling Interval : 64,733.33
- Sum of tolerable sample taintings : 61.98 percent

You must accept the population at the 90 percent confidence level when no more than 0.6198 total taintings are observed in a sample size of 15.

"A detailed Technical Specification on Monetary Unit Sampling – CaseWare IDEA Research Department Document of release date May 7, 2003 has been reproduced in Appendix IV for the benefit of the academically inclined readers. This document describes the mathematical procedures followed by IDEA in planning and Monetary Unit samples."

Monetary Unit Sampling is most effective in the following areas -

- Accounts Receivable confirmation - testing for the existence of accounts receivable.
- Accounts Receivable - testing for unrecorded receivables by sampling on cash receipts subsequent to year end.
- Allowance for doubtful accounts - testing for the valuation of accounts receivable when the allowance is calculated item by item.
- Inventory costing where overstatement of inventory is a concern.
- Loans or mortgages receivable - confirmation of new issues.
- Loans and mortgages receivable - confirmation of existing loans. In this case, sampling efficiency may be obtained by an appropriate choice of basic precision pricing and the application of the small tainting assumption.
- Accounts Payable - tests of existence. Sampling from disbursements subsequent to the period's end.
- Commission income - where income is based on certain types of revenues such as revenues from insurance premiums. In this case,

sampling efficiency may be obtained by choosing the basis precision pricing to reflect he maximum commission rate.

- Payroll expenses - testing for the valuation and existence of recorded payroll transactions.
- Other expenses - testing for the valuation and existence of other expenses.

9.6 Classical Variables Sampling

We use Classical Variables Sampling to provide estimates of quantities that are not recorded in the main database (also known as the population), by projecting results obtained for a sample drawn from the population.

A common use of Classical Variables Sampling is to estimate the total error and the total corrected, or audited, amount in a population, using an audited sample from the population

Classical Variables Sampling is also useful for making accounting estimates. For example, we may estimate how many rupees of inventory are represented by obsolete inventory, based on a sample for which the portion of each entry that is obsolete has been determined from manual records.

The sampling unit used in Classical Variables Sampling is a single record.

The sample database contains the key field from the population database, that is, the Numeric field whose entries are being checked by the audit, as well as an Editable field in which corrected amounts, determined through audit, can be entered.

Classical Variables Sampling techniques for projecting errors are most useful for sample auditing of accounts in which there may be errors or misstatements. If we wish to determine limits on the error in the population though we expect no errors or very few errors in the population, and if the principal risk is overstatement, we should consider using IDEA's Monetary Unit Sampling instead of Classical Variables Sampling. If we do not wish to project errors between the total book value and audited amounts, and require only limits on the total audited amount based on a sample of audited amounts, the mean estimator supplied by Classical Variables Sampling may be a good choice. The mean estimator can be computed for samples having no errors or very few errors.

Classical Variables Sampling is performed in three steps: sample preparation, sample audit, and sample evaluation. In sample preparation, the population database is stratified, the required sample size is determined based on our audit objectives, and the sample is extracted. In auditing the sample, we must update the field containing audited amounts with correct values determined through the audit procedure. Finally, the audited sample is evaluated to obtain projections of the most likely total audited amount and error in the population, and precision limits on this quantity. IDEA automates the sample preparation and sample evaluation phases of the Classical Variables Sampling process.

Classical Variables Sampling may use stratified estimates, in which case the population and sample databases must be stratified, or un-stratified estimates, in which case they are not. Generally, stratification reduces the sample size needed to achieve a desired result, but requires a larger number of errors in the sample. Various estimation techniques exist in Classical Variables Sampling theory. The most efficient technique generally depends upon the characteristics of the population database. IDEA supports six Classical Variables Sampling estimation techniques:

- Mean
- Difference
- Combined Ratio
- Separate Ratio
- Combined Regression
- Separate Regression.

Mean estimation is the least efficient of all and generally not used. except for making comparisons, or estimating quantities for which no recorded values exist prior to the audit.

Chapter 10
Field Manipulation

10.0 Introduction

Field Manipulation can be used within IDEA to view field definitions, add or delete fields, change field properties such as field name and type, or add tags (field templates) for the Smart Analyzer Financial component within IDEA.

Field Manipulation within IDEA can be performed through Virtual Fields and Editable Fields.

- Virtual Fields, as the name suggests, are computed fields through specific user based equations and/or functions. For example, adding a virtual numeric field using the @Function, @justnumbers will remove all numbers from an alpha-numeric narration within IDEA. This new virtual numeric field will contain only numbers like Cheque Numbers, for instance.
- Virtual Fields are generally used when we need to do a recalculation, re-computation, or field –conversion.
- Editable Fields, as the name suggests, are fields that are updated by user entry. For example, creating an editable field for recording the inventory count numbers during a physical inventory verification.
- Editable Fields are generally used when a user-based instruction or entry is to be made within the database. This is the only occasion where a user can add values into the database which cannot otherwise be altered in any manner.

10.1 Editable Fields

Editable fields are modifiable fields that can be added through:

- Field Manipulation
- Direct Extraction
- Import Assistant (with Delimited, Fixed length, EBCIDIC, and dBASE files only)

Editable fields are valuable for:

- Entering comments/data
- Amending fields, such as when evaluating monetary unit samples
- Performing new calculations and ratios from fields in the database

To help us distinguish Editable fields from protected fields, by default, IDEA displays editable data in the blue font. One can change the display colour through Grid Settings.

By default, IDEA does not record the definitions of Editable fields and any entries/changes we make to the data in the History. To record these, we need to clear the Do not record field changes made through editable fields check box in the System options.

If we make any changes to the data in an Editable field, any index that is based on the Editable field becomes invalid. To validate the index, we have to re-index the database.

Editable Fields: Examples

Example 1: Add an Editable field to a database for comments

Select Data > Extractions > Direct Extraction....

In the first row of the grid, specify a file name for the resultant database.

Click Create Fields and append an Editable field with the following definition:

Name	COMMENTS
Type	Character
Length	100
Parameter	""

Click OK on the Create Fields dialog box.

Click OK on the Extract to File(s) dialog box. This creates a new database with an additional empty field, COMMENTS.

Select the field and enter appropriate comments. The History records any possible (?)changes.

Example 2: Add an Editable field to a database for recording an audited value

Select Data > Extractions > Direct Extraction....

In the first row of the grid, specify a file name for the resultant database.

Click Create Fields and append an Editable field with the following definition:

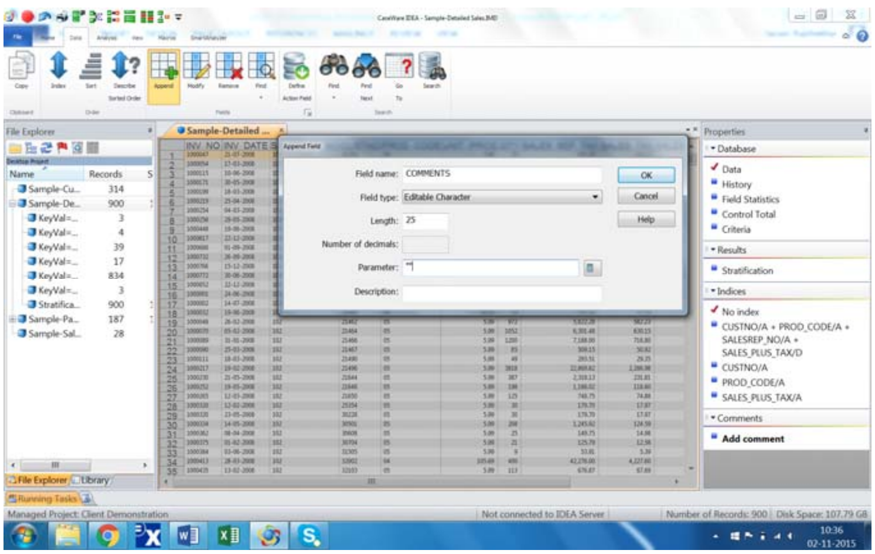
Name	AUDIT_VALUE
Type	Numeric
Dec	2
Parameter	VALUE (the field name of the items being audited)

Click OK on the Create Fields dialog box.

Click OK on the Extract to File(s) dialog box. This creates a new database with an additional field, AUDIT_VALUE. This field is the same as the field VALUE but it is editable.

When auditing the items, if a value is different to VALUE, change the entry in AUDIT_VALUE.

Extract where VALUE <> AUDIT_VALUE. If testing a sample of items, extrapolate the results for the whole population, if required. The History records any and all changes.



10.2 Virtual Fields

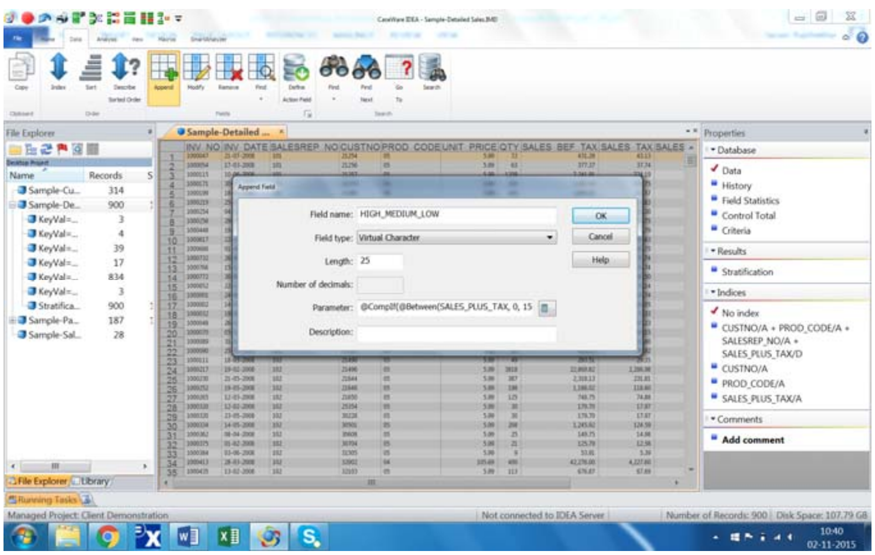
Appended (Virtual) fields are calculated fields that can be added and used for:

- Proving existing calculations in a database.
- Performing new calculations and ratios from fields within the database.

Converting data from one type into another, such as converting cheque numbers stored in a Character field into a Numeric field prior to testing for gaps in the cheque number sequence.

As virtual fields are calculated fields, IDEA computes the results "on the fly" and the results are not physically stored in the database. However, any files we create from a database with a Virtual field then contains the virtual field as a non- calculated value.

Virtual fields may also be added through Field Manipulation or the Import Assistant. If we want to add an empty field for entering comments/data, we need to add an Editable field.



Virtual Field: Examples

A virtual field can be used to

Convert a cheque number field (CHQNO) stored as a character field into a numeric field for testing for gaps by adding a Virtual Numeric field with the expression: @VAL(CHQNO)

Convert a date field (TRANS_DATE) stored as a character field with the format DD-MMM-YYYY (03-NOV-2015) into a date field by adding a date virtual field with the expression: @CTOD(TRANS_DATE, "DD-MMM-YYYY"). The date will then be displayed in the IDEA date format YY/MM/DD.

Caution:

Making changes to the database through Field Manipulation may show anything based on that field (results, drill-downs, indices, views, etc) as incorrect or invalid. Results can be made valid again by returning the settings to what they were when the result was created.

Field Manipulation and Commonly used @Functions in IDEA Software with Examples

S No	@Function	Purpose	Example
1	@age	For creating a new virtual numeric field with difference in days.	To arrive at the difference in days between 'Token Date' and 'Cheque Date' in ANY BILL PAY FILE use @age(Cheque Date, Token Date) through Field Manipulation.
2	@dow	Identify transactions made on any day of the week.	To capture cheques issued on Sunday use @dow(Cheque Date)=1 through a Direct Extraction. (1 is Sunday in IDEA)
3	@isblank	Filter out blank character fields	To present blank scheme agency name in GRANTS-IN-AID data use @isblank(Scheme Name) through Criteria.
4	@isini	Capture specific character strings from a character or alpha-numeric field where the <u>strings are not case sensitive</u>.	To display cheque in favour of field containing 'Shri' for cheques issued inter-se Government use @isini("shri", cheque in favour) through Direct Extraction.

S No	@Function	Purpose	Example
5	@compif	Conditional function which allows for preparing an expression with multiple conditions and consequent results. Useful for 'If then analysis'	In the Disaster Claim file, to create a new virtual numeric field titled Sanctioned Amount using @compif(Category="0400", 1000000, Category = "0104", 100000, "0103", 500000, 1, 0) through field manipulation. This means if the category is 0400 the result in the new virtual numeric field will be Rs. 1000000 and so on so forth.
6	@strip	Removes all special characters, spaces and punctuation marks from a character or alpha-numeric field.	To check if the same sanction number has been used more than once in the same controller, DDO, PAO, Function Head and Scheme by inserting special characters or spaces to bypass inbuilt system checks. Append a virtual character field with criteria @strip(Sanction Number). Then run a duplicate key detection test on the stripped sanction number field.
7	@left	Removes the left most occurrences of a character field or alpha-numeric field.	In the Disaster Claim file, every case claim is administered a Unique Case ID of 12 digits. The first 4 digits represent the category of claim i.e. death (0400) etc. can be captured separately by creating a virtual character field using @left(Unique Case ID, 4) through Field

S No	@Function	Purpose	Example
			Manipulation.
8	@right	Removes the right most occurrences of a character field or alpha-numeric field.	In the Disaster Claim file, every case claim is administered a Unique Case ID of 12 digits. The last 6 digits represent the claim serial number which can be captured separately by creating a virtual character field using @right(Unique Case ID, 6) through Field Manipulation.
9	@mid	Removes the central most occurrences of a character field or alpha-numeric field.	In the Disaster Claim, every case claim is administered a Unique Case ID of 12 digits. The 5th and 6th digit together represent the case tribunal number which can be captured separately by creating a virtual character field using @mid(Unique Case ID, 5,2) through Field Manipulation.
10	@year	Derives the year from any date field in IDEA	In Tax collection data via APPLICATION SYSTEM containing nodal scroll date, a new virtual numeric field can be created using @year(nodal scroll date) to derive the year of collection through Field Manipulation.
11	@month	Derives the month from any date field in IDEA	In Tax collection data via APPLICATION SYSTEM containing nodal scroll date, a new virtual numeric field can be created using

S No	@Function	Purpose	Example
			@month(nodal scroll date) to derive the month of collection through Field Manipulation.
12	@day	Derives the day from any date field in IDEA	In Tax collection data via APPLICATION SYSTEM containing nodal scroll date, a new virtual numeric field can be created using @day(nodal scroll date) to derive the day of collection through Field Manipulation.
13	@DtoDays	Computes the days corresponding to any date field in IDEA	In Gradation List to compute the days for the retirement date add a virtual numeric field with criteria @DtoDays(Retirement Date) through Field Manipulation
14	@DaystoD	Reverse compute Date from days.	In Gradation List to compute the date from the days add a **virtual date field** with criteria @DaystoD (Retirement Date Days+30) through Field Manipulation. The +30 is to suggest adding 30 grace-days as an example.
15	@match	Looks for multiple values in a field of choice through one single equation without the need of having to write multiple equations.	In Pension payments file to filter out basic pension equal to Rs. 3500 or Rs. 40000 (i.e. outliers) @match(Basic, 3500, 40000) can be run through a Direct Extraction. This avoids the need to run two equations like (Basic=3500) .OR. (Basic=40000)

Chapter 11

Importing various Data File Formats

IDEA and Data Files

11.0 Introduction and Stages in the Import of Data into IDEA

To analyse data to arrive at meaningful insights it is imperative to identify, request and download the required data onto the computer before starting IDEA. The "Guide to Downloading Data" lists several methods that can be used for downloading data to personal computers for audit purposes. It also provides pointers and tips on how to deal with a number of situations, on the basis of practical experience of innumerable downloads. The Guide is available on the CaseWare IDEA web site for registered CaseWare users.

11.0.1 Planning

The planning stage involves discussions with users and IT staff on what data is to be kept, how it is to be is stored, and how it can be transferred. Discussions on these issues can help us choose the best available method for our use.

11.0.2 Requesting the Data

Once the transfer has been planned, a formal request should be made to the relevant parties, asking permission where necessary, and describing what is required (format, method of storage, method of transfer, and time frame for the transfer of data). IDEA and other similar software rely on the transfer of data from the host computer system to the personal computer used by the auditor. The host computer may be a mainframe or a minicomputer, a personal computer, or possibly some form of distributed computing over a network. The data may be in a single file, a database, or spread across a number of systems. There are several means of transferring data, many conventions for the file types, and a myriad of structures for storing data.

11.0.3 Performing the Transfer

The transfer may be achieved through a direct connection from the host computer to the personal computer via a network that includes dedicated leased lines, or using media such as an external hard disk. The transfer may be achievable in a single action, or there may be an intermediate step, such as via a file server or a personal computer on a network and then to a laptop.

In order to interrogate data using IDEA, the data must first be imported. The **Import Assistant** guides the user through the process of import.

11.1 Data File Types compatible with IDEA Import

IDEA integrates and connects seamlessly with the following Data file formats:

1. Microsoft Excel
2. Microsoft Access
3. ASCII (Text) Fixed Length
4. ASCII (Text) Delimited (including Comma Separated Value)
5. EBCDIC Fixed Length
6. Print Report and Adobe PDF files
7. XML
8. SAP/AIS
9. dBASE III and IV
10. Lotus 1-2-3 (.wk1, .wk3, .wk4)
11. AS400
12. Proprietary Accounting Packages
13. ODBC

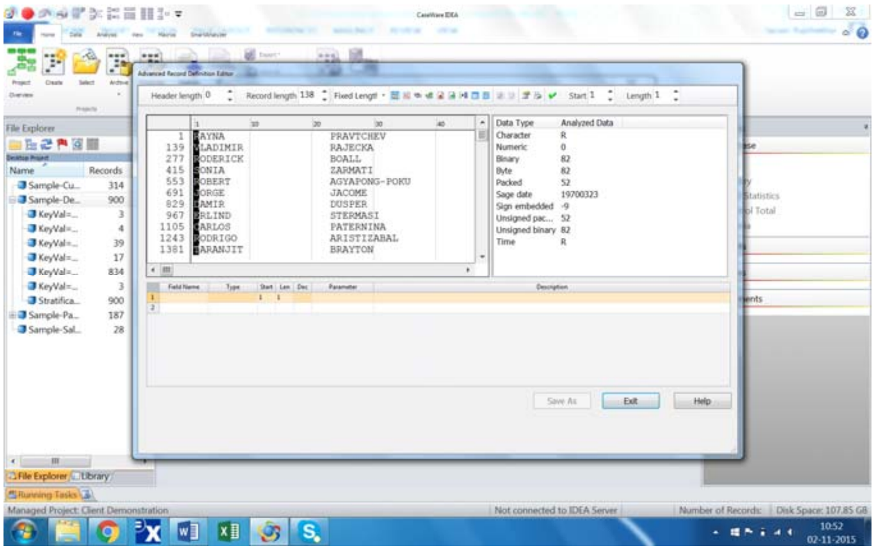
11.2 Data File Types and Common Application Systems

These data file formats are frequently used for varying mainstream commercial applications.

Report File Format	Illustrated Applications
Spreadsheet	Tally, SAP ERP, Oracle Apps.
ASCII Fixed Length	Tally
Comma Separated Value	BAAN ERP
Print Report	Home-grown ERP Systems
XML	Tally, Sabre (Airline Reservation System)
SAP / AIS	SAP ERP
Dbase	TBA, PBA Banking Applications.
Lotus	Lotus Workbook
ODBC	Direct interface to SQL, Oracle Server

For ASCII Fixed Length, EBCDIC Fixed Length, and ASCII Delimited, the data must be defined for IDEA in terms of size, type and number of fields (often called a record definition). Defining the data involves creating a record definition that describes the format of the file. The Import Assistant guides the user through the process of defining and importing the data.

Record definitions are file based and can be reused as required.



Other types of files, such as dBASE, Lotus 1-2-3, Microsoft Excel and Microsoft Access can be read directly. However, Lotus 1-2-3, SAP, and Microsoft Excel files must confirm to their respective database formats.

IDEA connects to a wide variety of databases, such as Oracle or Sybase, with Microsoft's ODBC (Open Database Connectivity) for data import.

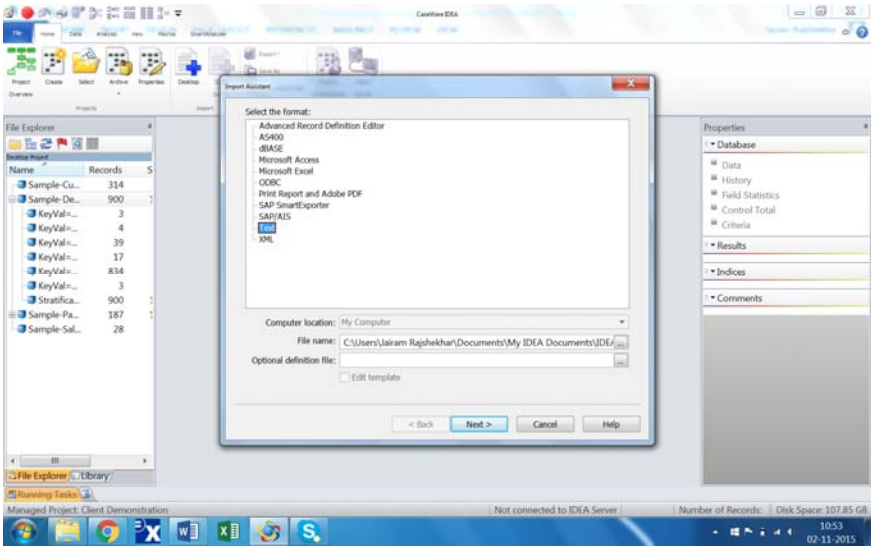
Where problems arise in obtaining the raw data, or an extract of the data, due to technical reasons, lack of IT assistance, or time constraints, a very successful and simple means of obtaining data is to import via Print Report files. Data formatted as a report is printed to a file and can be extracted using the Report Reader module in IDEA.

11.3 Importing a Microsoft Excel File

IDEA reads dBASE, Lotus 1-2-3, Microsoft Excel, and Microsoft Access files directly because the information about the data is detailed in the file header. These file types are the easiest to import, and are imported in generally the (same what same?) way.

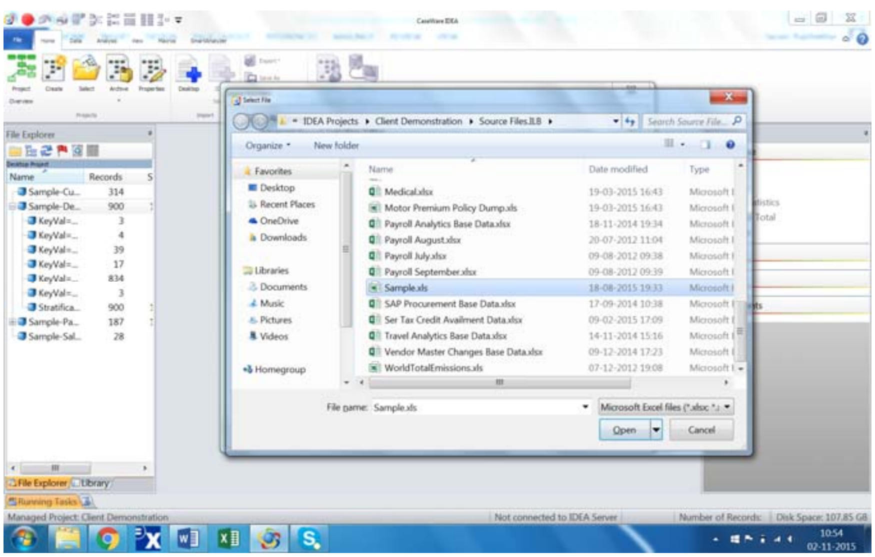
11.3.1 Click the **Desktop** button to display the **Import Assistant** dialog box.

11.3.2 Select **Microsoft Excel**



11.3.3 Click the **Browse** button adjacent to the **File name** box to select the Microsoft Excel file to be imported.

11.3.4 Navigate to and select C:\Users\PCNAME\Documents\My IDEA Documents\IDEA Projects\Samples\Source Files.ILB\Sample.xls



11.3.5 Click **Open**.

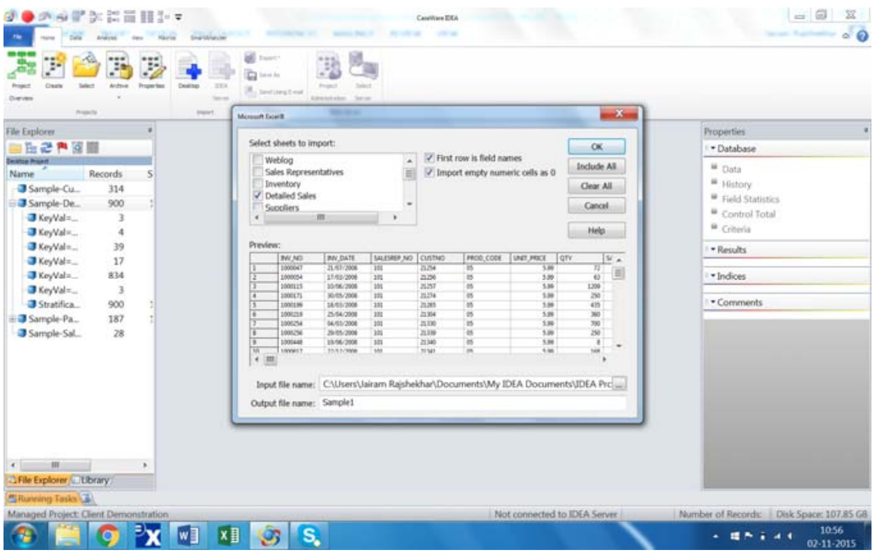
11.3.6 Click **Next**.

The **Excel** dialog box appears. The **Excel** dialog box displays a preview of the Microsoft Excel file.

IDEA determines the field types and widths by scanning all rows in the spreadsheet. During the scan, if any unusual non-numeric characters are found in Numeric or Date fields, the affected fields will be imported into IDEA as Character fields.

In the **Preview** area of the **Excel** dialog box, the Date and Character fields are left-justified, while Numeric fields are right-justified.

11.3.7 The first row of the spreadsheet contains column field names. We check the box **First Row is Field Names** to ensure that the field names from Excel and duly reflected in IDEA. This allows for meaningful analysis of data imported based on the field names captured.



11.3.8 If the First row in Excel is blank, default field names (Col1, Col2) will be provided during the import.

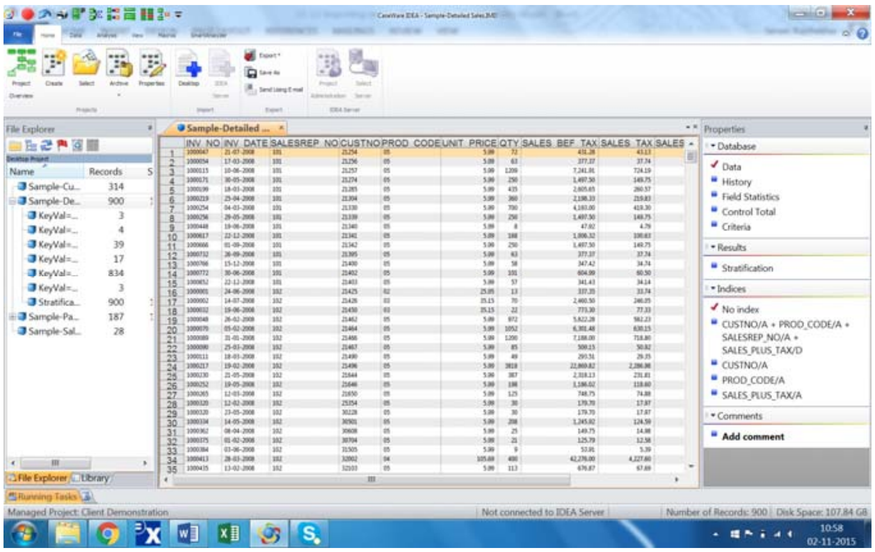
11.3.9 If the spreadsheet being imported has one or more empty cells in a column, the data in that column is imported into IDEA as a Character field. If the data is numeric, date or time and we want to import it into IDEA as a Numeric, Date or Time field, we have to select the **Import empty numeric cells as** 0 check box.

11.3.11 In the **Select sheets to import** box, select the worksheet Detailed Sales. If we select multiple worksheets to import, each worksheet will be imported as a separate database.

11.3.11 If the user does not enter a unique Output File Name, then the default name i.e. Excel file name will take over.

11.3.12 Click **OK**.

The file is imported into IDEA and displayed in the **Database** window.



Note

Worksheet/Spreadsheet Import Rules

It is strongly recommended that before importing data from Microsoft Excel or lotus 1-2-3 worksheets/spreadsheets, the format of the data is checked to ensure that it conforms to the following set of rules:

- The data to be imported is in table format, such as a block of rows and columns.
- The first row of the table should be the field name or contain the data to import.
- All columns containing both text and numeric data are formatted as text.

- All numeric columns contain a fixed number of decimal places.
- Blank lines, sub-total lines and total lines are removed from the data area.
- Merged cells are un-merged.
- The range of data to be imported is selected.
- The above steps are performed for all tables of data within the worksheet/spreadsheet.
- The worksheet/spreadsheet is closed during the import process.

11.4 Importing ASCII, EBCDIC or Delimited (Comma Separated Value) ASCII Files

11.4.1 Introduction

File types such as ASCII (American Standard Code for Information Interchange) and EBCDIC (Extended Binary Coded Decimal Interchange Code) are very common. Most mid-range computers and personal computers hold data in ASCII format, while the EBCDIC format is typically found on mainframe and mid-range computers.

Unlike Microsoft Excel, Microsoft Access, Lotus 1-2-3, and dBASE, ASCII and EBCDIC files do not contain standards or headers defining the structure of the data. Since no such header exists, we have to define the file structure in IDEA.

The following print-screen displays the ASCII file format:

```
99/08/01J1   244        2115    28.98    0.0099/08/01J1   244        5010   413.94
 0.0099/08/01J1   244        2300     0.00 442.9299/08/01J2   240        1100
1676.16    0.0099/08/01J2   240        4010     0.001676.1699/08/04J3   237
1100  470.00    0.0099/08/04J3   237        2116     0.00  30.0099/08/04J3   237
     4300     0.00 440.0099/08/04J4   238        1100  642.00    0.0099/08/04J4
238        2116     0.00  42.0099/08/04J4   238        4020     0.00
600.0099/08/04J4   238        2115     7.31    0.0099/08/04J4   238        5800
17.88    0.0099/08/04J4   238        5810  143.16    0.0099/08/04J4   238
2010     0.00 168.3599/08/07J6   13239      2115  126.49    0.0099/08/07J6
13239      5010 1807.00    0.0099/08/07J6   13239      2010
0.001933.4999/08/07J7   206        2010 1933.49    0.0099/08/07J7   206
1010     0.001933.4999/08/06J8   280358     2115   53.13    0.0099/08/06J8
280358     5010  750.00    0.0099/08/06J8   280358     5510     9.00
0.0099/08/06J8   280358     2300     0.00 812.1399/08/07J9   234        1100
2917.89    0.0099/08/07J9   234        2100     0.00 216.1499/08/07J9   234
2116     0.00 176.7599/08/07J9   234        4300     0.002525.0099/08/07J10 895
     1010 2917.89    0.0099/08/07J10 895        1100     0.002917.89
```

Technical Notes

Explanation of the ASCII Delimited File Format

ASCII Delimited files are a common format of variable length file where each field is long enough to contain only the data stored within it. This is an advantage with text fields, such as names, addresses, and narratives.

In order that software use such data files, there is a separator (a special character) at the end of each field within the record. Additionally, text fields may be enclosed (encapsulated) with another character, typically quotation marks (“ “).

The separator is often a comma (,) and such files are known as **comma separated value (CSV)**. However, the separator could be another ASCII character, for example a ; : ^ ~ These files are referred to as **ASCII Delimited** (comma separated files are also often referred to as **ASCII Delimited**).

There must also be a record delimiter at the end of each record, typically the carriage return (<CR>) and line feed (<LF>) characters, or sometimes only the line feed character.

When this type of file is imported, a fixed length database is created within IDEA.

Example of an ASCII Delimited – CSV:

“6776”,1865,1,190,”04-02-2006” <CR><LF>

“6714”,1865,1,97,”04-02-2006”<CR><LF>

“11905”,2366,11,9,”25-02-2006”<CR><LF>

“7555”,4352,1,79,”03-03-2006”<CR><LF>

“4547”,4815,5,17,”03-03-2006”<CR><LF>

“6344”,4815,1,7,”03-03-2006”<CR><LF>

“151”,2366,1,3,”29-03-2006”<CR><LF>

“145”,2366,1,1,”05-08-2006”<CR><LF>

“206”,2366,1,2,”29-03-2006”<CR><LF>

“207”,2366,1,3,”29-03-2006”<CR><LF>

The following specimen data request has been reproduced.

Request for Data Files for Analysis Purposes – DATA INDENT

Please supply the following data from the Inventory system in **ASCII Delimited** format:

FIELD NAME	TYPE	Dec	DESCRIPTION
PRODCODE	N	0	Product code
DEPOT	C		Depot name
QTY	N	0	Quantity on hand
AV_COST	N	3	Average cost for item
TOTALCOST	N	3	QTY * AV_COST
OBSOLETE	C		Obsolete = “Y”
MAX	N	0	Maximum inventory level on hand
MIN	N	0	Minimum inventory level on hand
DELQTY	N	0	Last delivery quantity
ORDERNO	N	0	Last delivery order number
DELDATED	D		Date of last delivery (YYYYMMDD)
CURSELLPRI	N	2	Current selling price
CUREFFDATE	D		Date selling price effective (YYYYMMDD)
PREVSELPRI	N	2	Previous selling price
USAGE	N	0	Sales quantity in current year
PREVCOST	N	3	Unit cost of last purchase

Data is required for the following period: December 31, 2014

Please also supply the following control totals for reconciliation purposes:

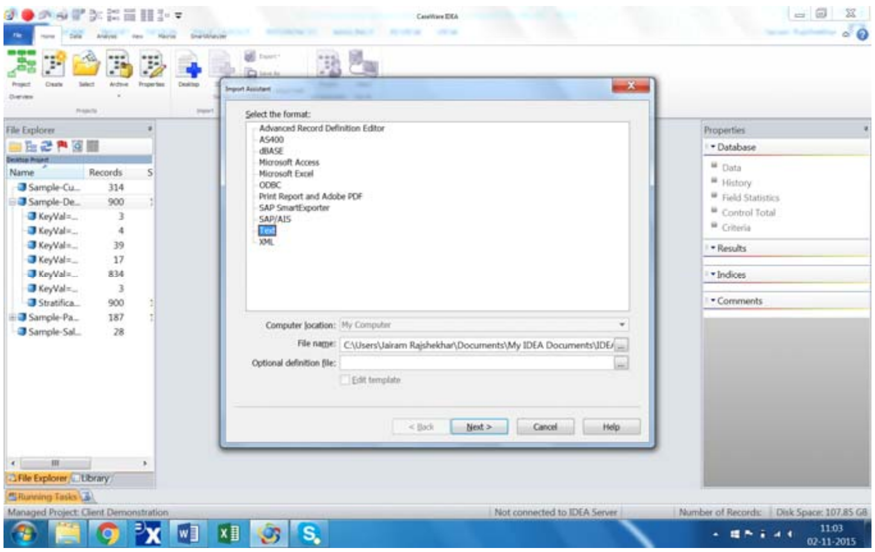
1. Total cost of inventory on hand at December 31, 2014.
2. Number of transactions supplied, i.e., product lines.
3. Record definition to support the data file provided.

Importing an ASCII Delimited File

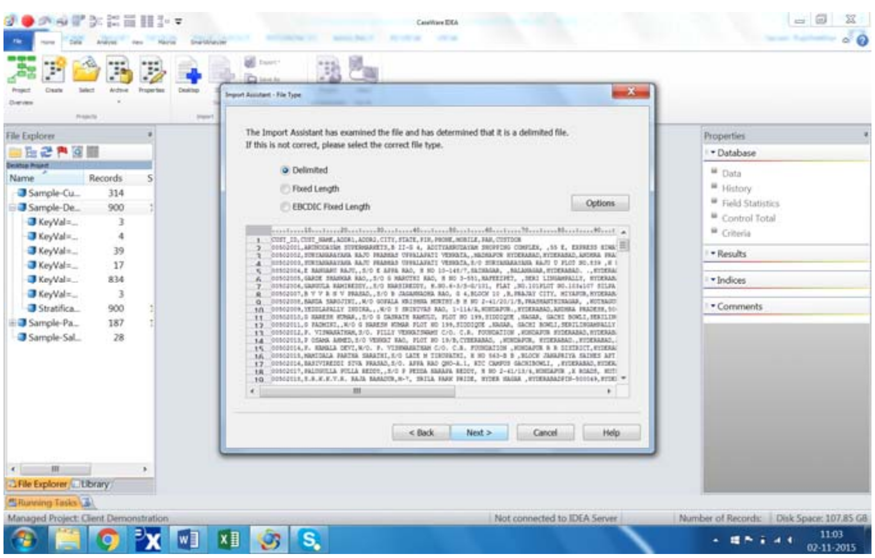
You have been provided with an **ASCII Delimited** file:

11.4.2 From IDEA select the **Import Assistant** by clicking the Desktop button under Home in the IDEA toolbar.

11.4.3 Select **Text** from the list. Click the **Browse** button and the **Select File** dialog box will be displayed. Navigate to and select the file from the **Working Folder**. Click **Open**, and then click **Next**.



11.4.4 The **Import Assistant - File Type** screen will be displayed. Ensure **Delimited** is selected as the correct file type.

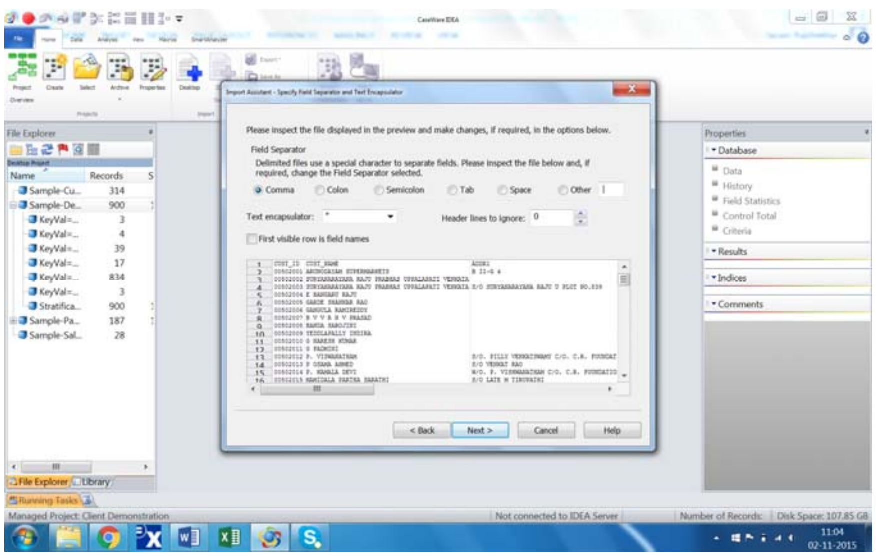


View the data and determine the field separator and the record delimiter.

Click **Next**

11.4.5 The **Import Assistant** will try to determine the field separators and text encapsulators (if any) for the file. Accept the defaults and click **Next** to proceed.

Notes	Some (but not this) **ASCII Delimited** files have field names as the first row of the file. To use these as field names, select the **First visible row is field names** check box.



11.4.6 The **Import Assistant - Field Details** step will be displayed. Click on each field heading in turn and using the record definition, define the field details.

FIELD NAME	TYPE	Dec	DESCRIPTION
PRODCODE	N	0	Product code
DEPOT	C		Depot name
QTY	N	0	Quantity on hand
AV_COST	N	3	Average cost for item
TOTALCOST	N	3	QTY * AV_COST
OBSOLETE	C		Obsolete = “Y”

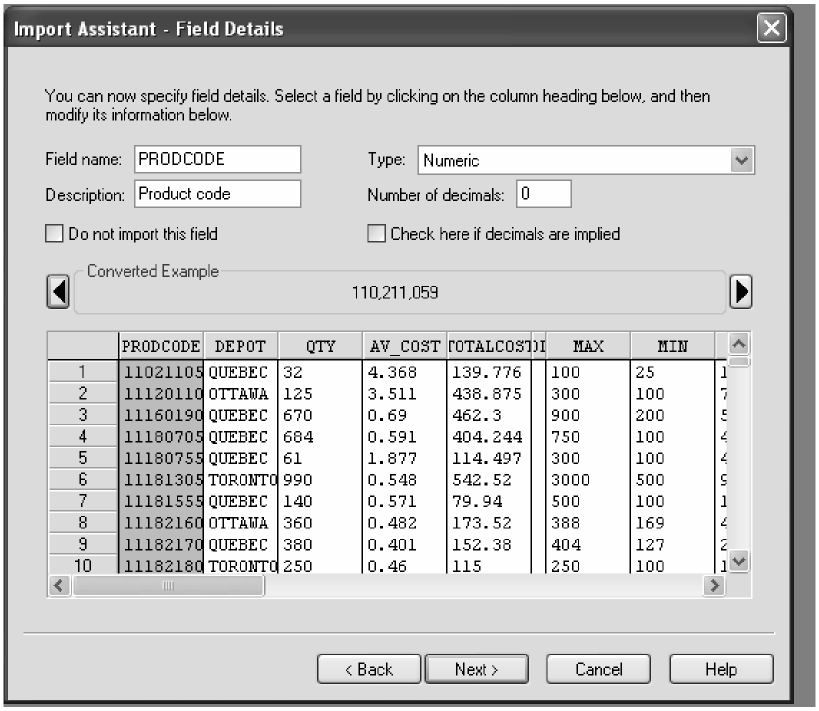
FIELD NAME	TYPE	Dec	DESCRIPTION
MAX	N	0	Maximum inventory level on hand
MIN	N	0	Minimum inventory level on hand
DELQTY	N	0	Last delivery quantity
ORDERNO	N	0	Last delivery order number
DELDATED	D		Date of last delivery YYYYMMDD
CURSELLPRI	N	2	Current selling price
CUREFFDATE	D		Date selling price effective YYYYMMDD
PREVSELLPRI	N	2	Previous selling price
USAGE	N	0	Sales quantity in current year
PREVCOST	N	3	Unit cost of last purchase

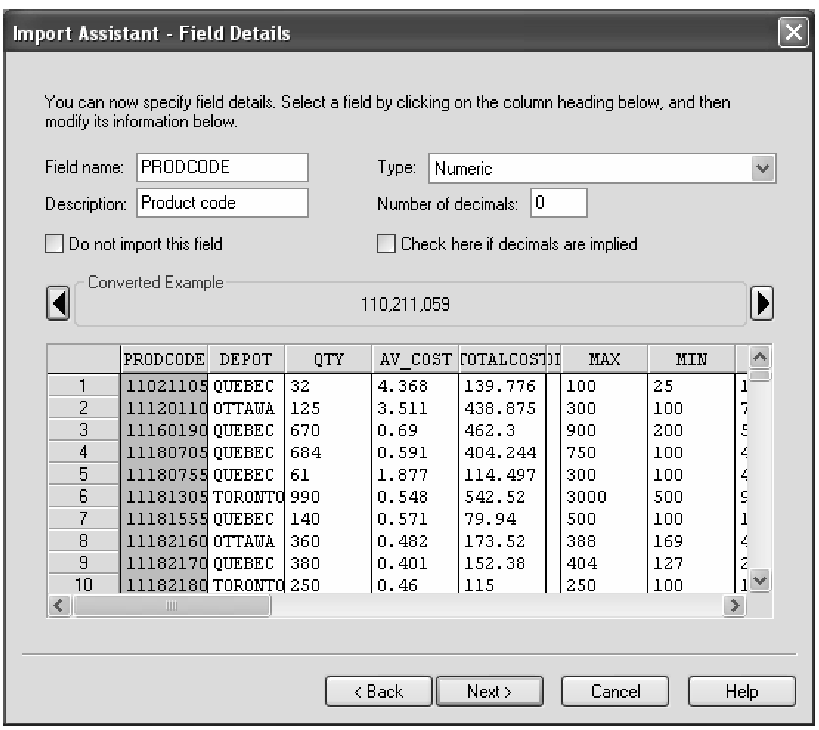
Enter the correct field name in the **Field name** box. The **Import Assistant** suggests the file type for that field in the **Type** box. If this is incorrect then change it to the field type specified in the record definition.

Enter the description of the field in the **Description** box.

If the field is a Numeric field with decimals, it is necessary to specify whether or not the decimals are implied. However, if there are decimals in this database, the decimal place is stored in the number, therefore the **Check here if decimals are implied** check box must **not** be selected. Specify the number of decimals in the **Number of decimals** box.

Define the Date fields as type **Date**, i.e., change from **Character** to **Date** in the **Type** box. A mask must be defined for the Date fields. Click the **Date Mask** box and enter the appropriate date format i.e., YYYYMMDD.

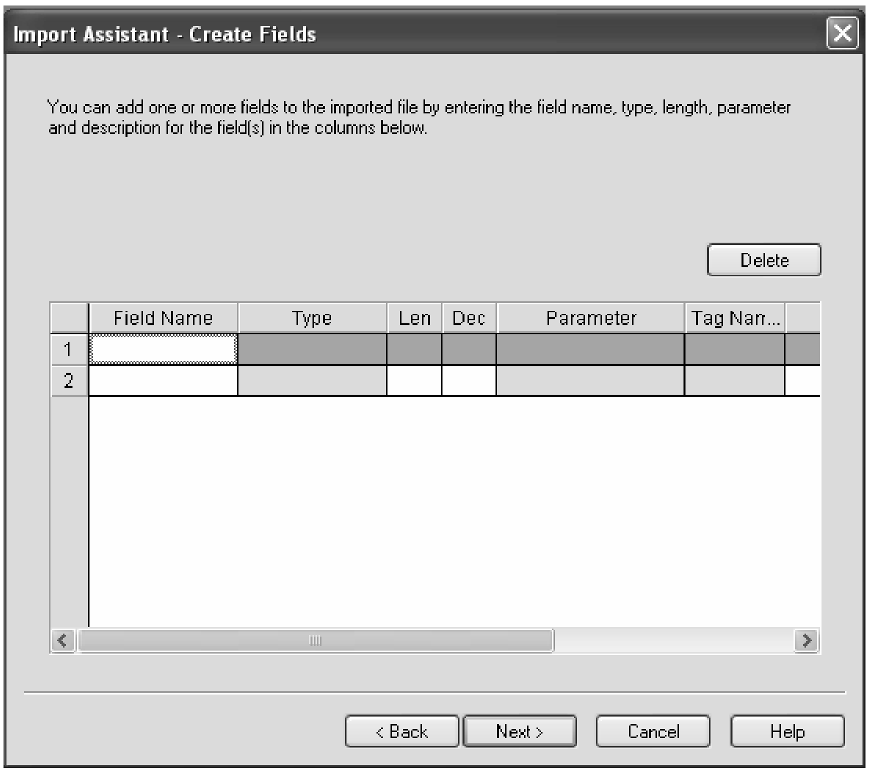

Import Assistant - Field Details
You can now specify field details. Select a field by clicking on the column heading below, and then modify its information below.
Field name: PRODCODE
Type: Numeric
Description: Product code
Number of decimals: 0
Do not import this field
Check here if decimals are implied
Converted Example
110,211,059
PRODCODE DEPOT QTY AV_COST TOTALCOST MAX MIN
1 11021105 QUEBEC 32 4.368 139.776 100 25
2 11120110 OTTAWA 125 3.511 438.875 300 100
3 11160190 QUEBEC 670 0.69 462.3 900 200
4 11180705 QUEBEC 684 0.591 404.244 750 100
5 11180755 QUEBEC 61 1.877 114.497 300 100
6 11181305 TORONTO 990 0.548 542.52 3000 500
7 11181555 QUEBEC 140 0.571 79.94 500 100
8 11182160 OTTAWA 360 0.482 173.52 388 169
9 11182170 QUEBEC 380 0.401 152.38 404 127
10 11182180 TORONTO 250 0.46 115 250 100
< Back
Next >
Cancel
Help


Import Assistant - Field Details
You can now specify field details. Select a field by clicking on the column heading below, and then modify its information below.
Field name: PRODCODE
Type: Numeric
Description: Product code
Number of decimals: 0
Do not import this field
Check here if decimals are implied
Converted Example
110,211,059
PRODCODE DEPOT QTY AV_COST TOTALCOST MAX MIN
1 11021105 QUEBEC 32 4.368 139.776 100 25
2 11120110 OTTAWA 125 3.511 438.875 300 100
3 11160190 QUEBEC 670 0.69 462.3 900 200
4 11180705 QUEBEC 684 0.591 404.244 750 100
5 11180755 QUEBEC 61 1.877 114.497 300 100
6 11181305 TORONTO 990 0.548 542.52 3000 500
7 11181555 QUEBEC 140 0.571 79.94 500 100
8 11182160 OTTAWA 360 0.482 173.52 388 169
9 11182170 QUEBEC 380 0.401 152.38 404 127
10 11182180 TORONTO 250 0.46 115 250 100
< Back
Next >
Cancel
Help

Click **Next**.

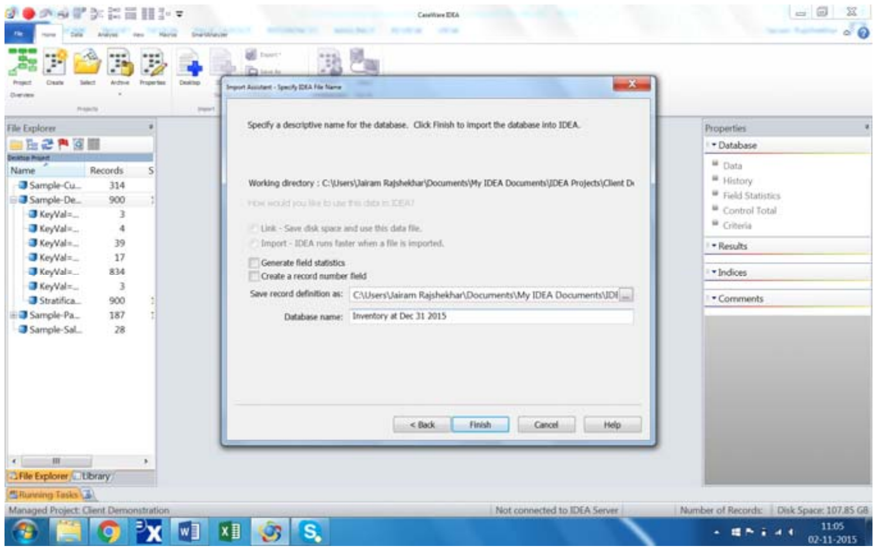
11.4.7 The **Import Assistant - Create Fields** screen will be displayed. The **Import Assistant -Create Fields** screen allows you to add fields in the imported file. This can be done during the import or at any time while using IDEA. For this exercise, no fields will be added.

Click **Next**, and then **Next** again.



11.4.8 The definition will automatically be saved in the **Working Folder**. IDEA will give the definition the same name as the source file. Click the **Browse** button adjacent to the **Save record definition as** box if you want to change the name.

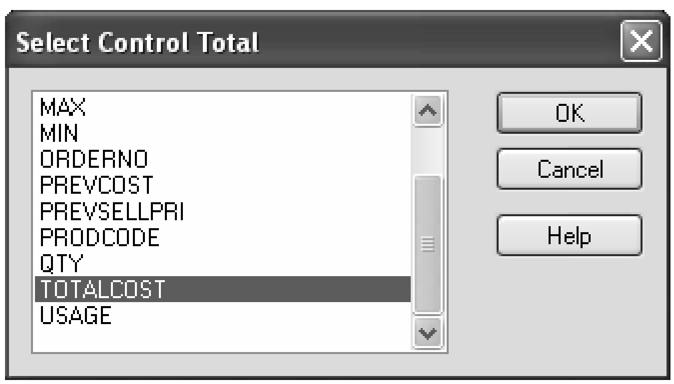
11.4.9 Select the **Generate field statistics** check box and enter **Inventory at Dec 31 2014** in the **Database name** box.



11.4.10 Click **Finish** to import the file into IDEA.

11.4.11 Maximize the *Database Window*.

11.4.12 Click the **Control Total** link in the *Properties Window* and select the TOTALCOST field and then click **OK**.



Results

The control total of **626,963.915** will appear beside the **Control Total** link in the *Properties Window*.

There are **767** records as can be seen on the **Status** bar at the bottom of the screen.

You are now ready to verify that the data has been imported correctly and to commence testing.

Verifying That the Database Has Been Correctly Imported

11.4.13 Click the **Field Statistics** link in the *Properties Window*.

Field Statistics are displayed for the **Numeric** fields in the database.

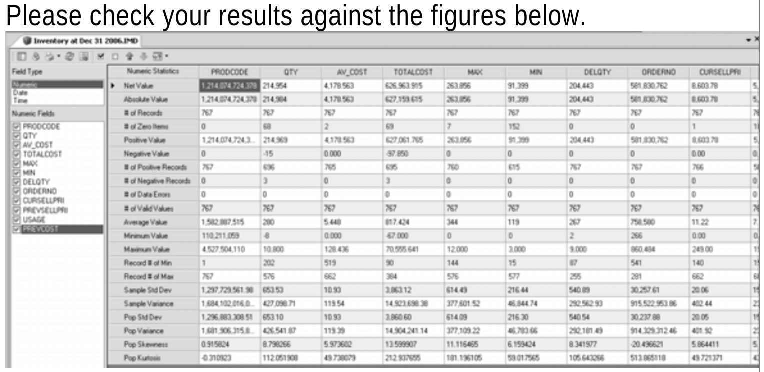
> **Notes**
>
> The **Field Statistics** are available instantly as they were generated when the database was imported.
>
> Field statistics are also available for **Date** fields. These will be reviewed in the following exercise.

11.4.14 Study the field statistics for all Numeric fields, but especially, the QTY and

TOTALCOST fields. Notice in particular the **Net Value**, **Maximum Value**, **Minimum Value**, **# of Zero Items**, and **# of Negative Records** statistics.

Results

Please check your results against the figures below.



Numeric Statistics	PRODCODE	QTY	AV_COST	TOTALCOST	MAX	MIN	DELQTY	ORDERNO	CURSELLPRI
Net Value	1,214,074,724,378	214,954	4,178.563	626,963.915	263,856	91,399	204,443	581,830,762	8,603.78
Absolute Value	1,214,074,724,378	214,984	4,178.563	627,159.615	263,856	91,399	204,443	581,830,762	8,603.78
# of Records	767	767	767	767	767	767	767	767	767
# of Zero Items	0	68	2	69	7	152	0	0	1
Positive Value	1,214,074,724,3...	214,969	4,178.563	627,061.765	263,856	91,399	204,443	581,830,762	8,603.78
Negative Value	0	-15	0.000	-97.850	0	0	0	0	0.00
# of Positive Records	767	696	765	695	760	615	767	767	766
# of Negative Records	0	3	0	3	0	0	0	0	0
# of Data Errors	0	0	0	0	0	0	0	0	0
# of Valid Values	767	767	767	767	767	767	767	767	767
Average Value	1,582,887,515	280	5.448	817.424	344	119	267	758,580	11.22
Minimum Value	110,211,059	-8	0.000	-67.000	0	0	2	266	0.00
Maximum Value	4,527,504,110	10,800	128.436	70,555.641	12,000	3,000	9,000	860,404	249.00
Record # of Min	1	202	519	90	144	15	87	541	140
Record # of Max	767	576	662	384	576	577	255	281	662
Sample Std Dev	1,297,729,561.98	653.53	10.93	3,863.12	614.49	216.44	540.89	30,257.61	20.06
Sample Variance	1,684,102,016,0...	427,098.71	119.54	14,923,698.38	377,601.52	46,844.74	292,562.93	915,522,953.86	402.44
Pop Std Dev	1,296,883,308.51	653.10	10.93	3,860.60	614.09	216.30	540.54	30,237.88	20.05
Pop Variance	1,681,906,315,8...	426,541.87	119.39	14,904,241.14	377,109.22	46,783.66	292,181.49	914,329,312.46	401.92
Pop Skewness	0.915824	8.798266	5.973602	13.599907	11.116465	6.159424	8.341977	-20.496621	5.864411
Pop Kurtosis	-0.310923	112.051908	49.738079	212.937695	181.196105	59.017565	105.643266	513.865118	49.721371

Field: QTY

STATISTIC	VALUE	COMMENT
Net Value	214,954	
Average Value	280	
Minimum Value	-8	Should not be any negative quantities
Maximum Value	11,800	
# of Records	767	
# of Zero Items	68	
# of Positive Records	696	
# of Negative Records	3	These should be identified

Field: TOTALCOST

STATISTIC	VALUE	COMMENT
Net Value	626,963.915	
Average Value	817.42	
Minimum Value	-67.00	
Maximum Value	70,555.64	
# of Records	767	
# of Zero Items	69	Different number or zero-quantity
# of Positive Records	695	
# of Negative Records	3	

11.4.15 If your computer is attached to a printer, print the field statistics for the **QTY** and **TOTALCOST** fields by clicking the **Print** button on the **Field Statistics** toolbar and selecting each field.

11.4.16 The **Field Statistics** reports should be filed with the audit documentation as proof of reconciliation.

11.4.17 Return to viewing the database by clicking the **Data** link in the *Properties Window*.

Conclusion

The file has been imported correctly and the client's report footer total reconcile correctly. However, there are some negative items, which we should identify and report them to the client.

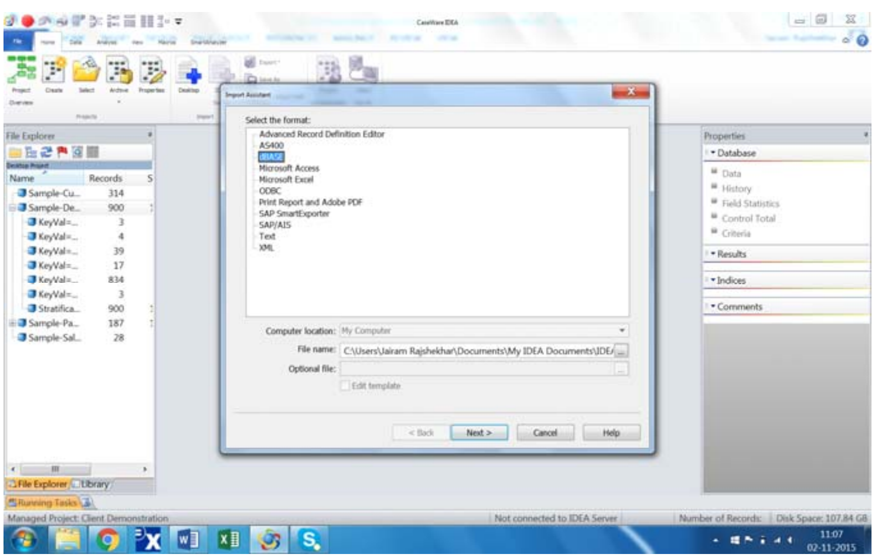
11.5.0 Importing a dBASE File

11.5.1 Introduction

dBASE files are one of the most common computer formats used for exporting data from computer systems and the easiest file format to import into IDEA. The DBASE file structure consists of a file header holding information about the file, including the field definition information. dBASE files are imported directly when the file is dBASE III or newer. If we have older Dbase files, it is recommended that we read them into Microsoft Excel or Microsoft Access and then export or import them to IDEA from there.

11.5.2 From IDEA, select the **Import Assistant** by clicking on the Desktop button under Home in the IDEA toolbar.

11.5.3 Select **dBASE** from the list of available formats.

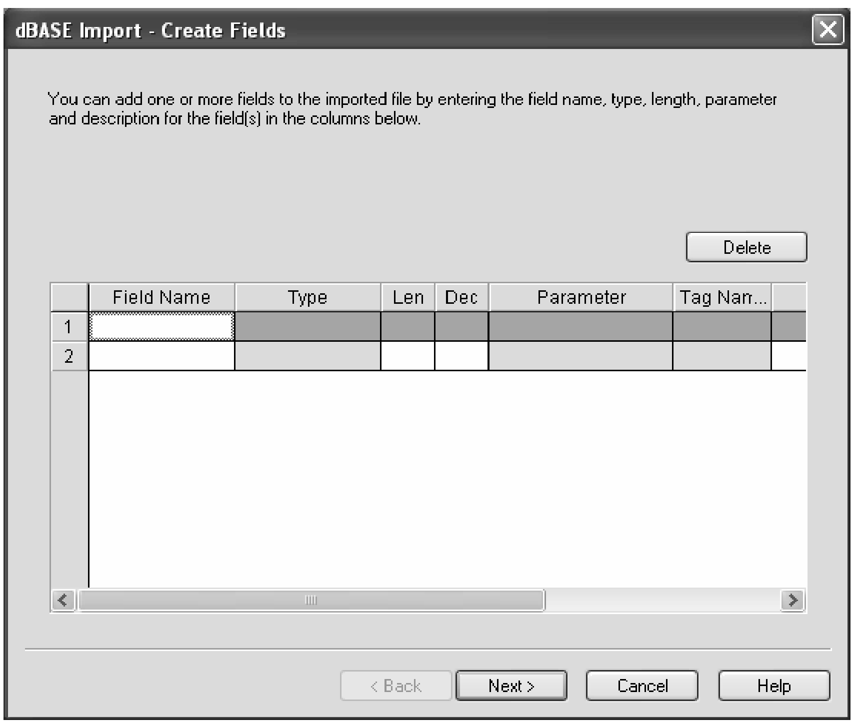


11.5.4 Select the file to import by clicking the **Browse** button then navigate to and select the following file:

Click **Open**, and then click **Next.**

Note	When selecting a file, the **Files of type** may need to be changed to *dBase* or *All Files*.

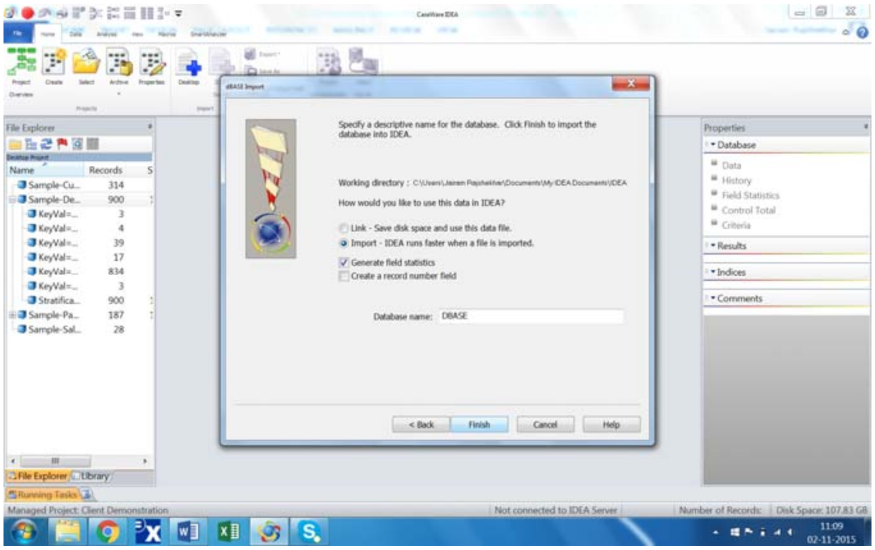
10.5.5 The **dBASE Import – Create Fields** screen dialog box will be displayed. **Create Fields** allows you to add **Virtual** or **Editable** fields in the imported file. This can be done during the import or at any time while using IDEA. For this exercise, no fields will be added. Click **Next.**



10.5.6 The final step of the **Import Assistant** will ask you to change or confirm details about the database to import. Select or confirm the following options:

- Leave the default selection to **Import** rather than **Link** the file
- Accept the option to **Generate field statistics** on import

- Accept the default not to **Create a record number field**
- Enter the database name: **Accounts Receivable**



Click **Finish**. The **Accounts Receivable** database will be imported, opened, and selected as the active database.

Technical Notes

Differences between Importing and Linking to a Data File

Each IDEA import or link results in an IDEA master database, *filename.imd*, being created in the **Project Folder.**

However

- In both cases, a file (*filename.imd)* is created that stores all the information about the database.
- **Importing** results in a copy of the data being included in the *filename.imd* file.
- If the option to **Link** to the database is selected, the original data is not included in the *filename.imd* file. The advantage of linking is that it can save disk space since the IMD file is smaller than it would be if the source data had been copied. However, both the source data file and the *filename.imd* file must be stored because IDEA must have access to the source data file to obtain the data needed to perform analytical functions.
- Testing will be faster on imported data files.
- Housekeeping is simplified when data is imported as there is only one file to backup/restore/copy/move and it is stored in the **Working Folder**.

It is recommended that the **Import** option be used provided we have sufficient disk space.

11.6.0 Importing a Print Report File

11.6.1 Introduction

Report files which can be imported into IDEA have a unique format unlike spreadsheet and delimited files. Report files have headers, footers, sub-headers, sub-totals and transactions.

IDEA's **Import Assistant** provides importing capabilities for data files that are in a report format. **Report Reader** is part of the **Import Assistant** within IDEA and can be accessed through it.

Report Reader extracts data from plain text reports, print reports saved to file or Adobe Acrobat PDF files and translates it into the specialised file formats of IDEA. The reports might have come from an application on the computer or from a mainframe computer.

11.6.2 What Kind Of Files Can Be Imported?

With **Report Reader**, we can import data from any ASCII file (TXT, ASC, CSV, DAT, etc.) or an Adobe Acrobat Portable Document Format (PDF) file.

An ASCII file is a plain text file or a printer output file. Most DOS applications will produce text output by a process called "print to disk". This simply means sending a report to a disk file instead of printing it. Windows applications usually have an ASCII text output option or can print a report to a text file with the Generic/Text Only printer driver.

The following is a list of Text formats that **Report Reader** can read.

Mainframe text reports

Minicomputer text reports

DOS Print to Disk files

Generic/Text Only files (Windows)

Word Processor Text Only files

Text only accounting reports

E-mail reports

Online communications text captures

Print to Files (PRN)

Tab Separated Variables

Standard Data Format (SDF)

Unconverted Files from SAP

Adobe Acrobat PDF files can be opened in **Report Reader**. The text from the PDF file will be converted into plain text format which does not maintain the font size, color, text bounding box and rotation angle as in the PDF file. A page break is inserted at the end of each page and graphics and images are not imported.

11.6.3 How Does Report Reader Work?

Report Reader displays our entire original file in a section of the Report Reader window called the Report view. Using the scroll bars, we can scroll through the report to view the details of the data and the format. Data to be imported can be selected with a simple point-and-click operation like those in Excel and other applications. The user selects the portions of the file to import. No special knowledge of the file structure or command language is needed to use Report Reader.

Report Reader imports the data we select into IDEA. Our selections and instructions for importing data in a report can be saved as a "template". This template can be re-used to import data from other similar data files in the same format. Using templates saves time, particularly if we need to import the same or similar reports on a regular basis.

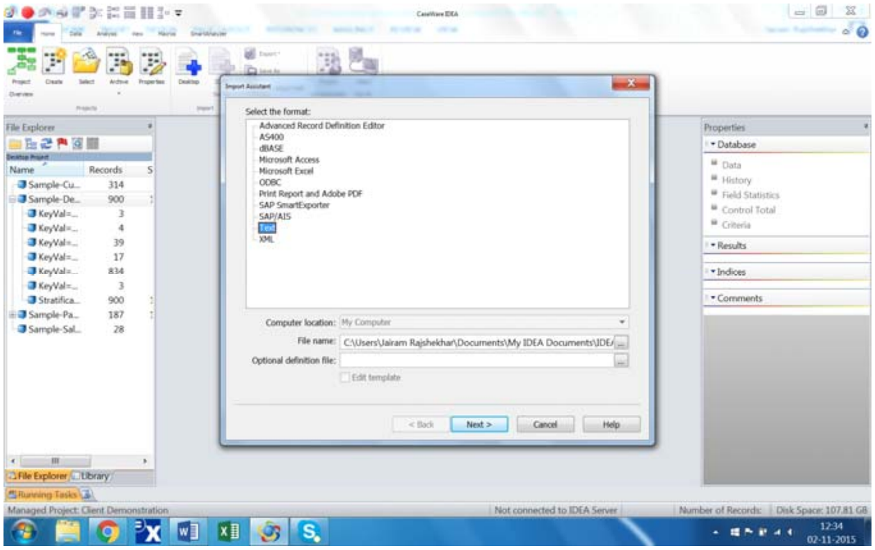
Report Reader does more than simply extract data from one file and put it in another. It distinguishes between numeric and non-numeric information and handles both appropriately. It also automatically formats data and time fields in their native field type without the need for any manual intervention.

To import a sample Print Report file:

11.6.4 Click the Desktop button under Home to access the Import Assistant.

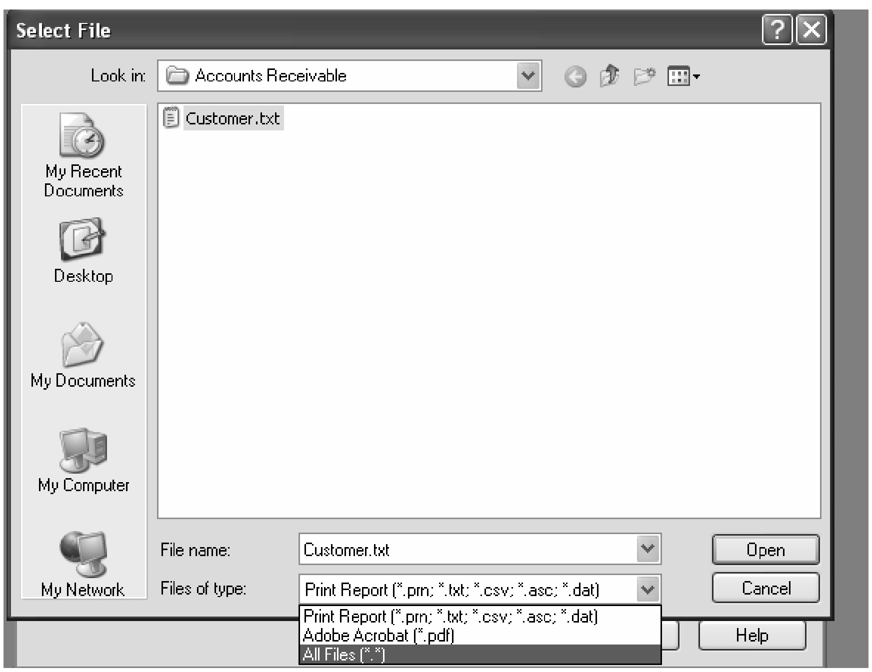
The Import Assistant dialog box will be displayed.

11.6.5 Select Print Report and Adobe PDF.



11.6.6 Click the **Browse** button to navigate to and select the required file.

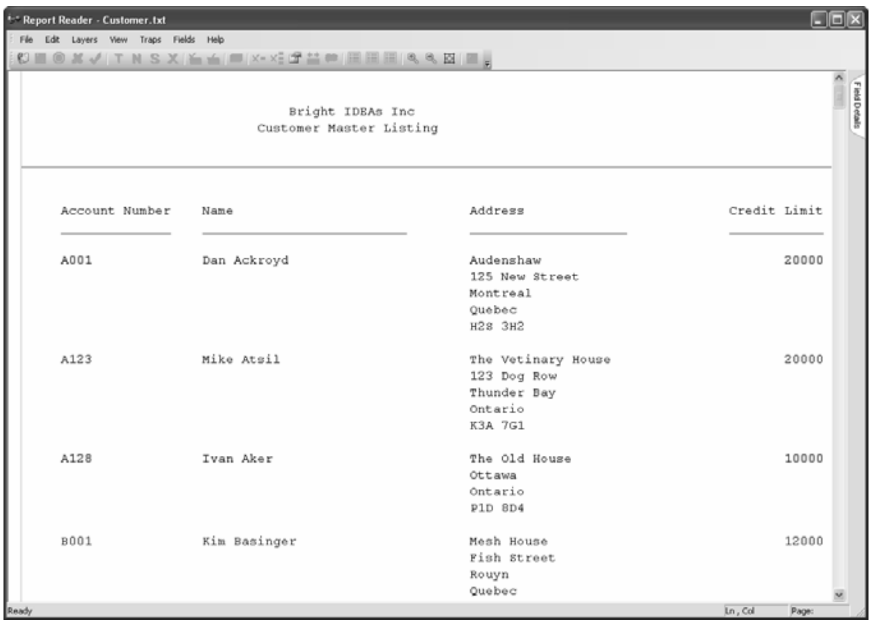
11.6.7 Select the sample file, and then click **Open**. Notice the various types of Print Report files IDEA can import.



11.6.8 Click **Next** to open the **Report Reader**.

11.6.9 Maximize the **Report Reader** window. Scroll through the report and notice the information that is included in the report:

- Account Number
- Name
- Address
- Credit Limit



Technical Notes

The **Report Reader** uses a set of layers to define the data to be imported into IDEA. The first step is to identify the **Base Layer**. The **Base Layer** is the line(s) that contains the lowest level of detail in the report and is often the transaction row in the report. It is very important to select the line from the first record in the report to ensure that all records are duly selected.

WARNING: The **Report Reader** is a top-down processor; therefore, any information that precedes the defining record in the report is ignored. This feature is helpful when very detailed header information is included in the report and we want that data excluded from the database in IDEA.

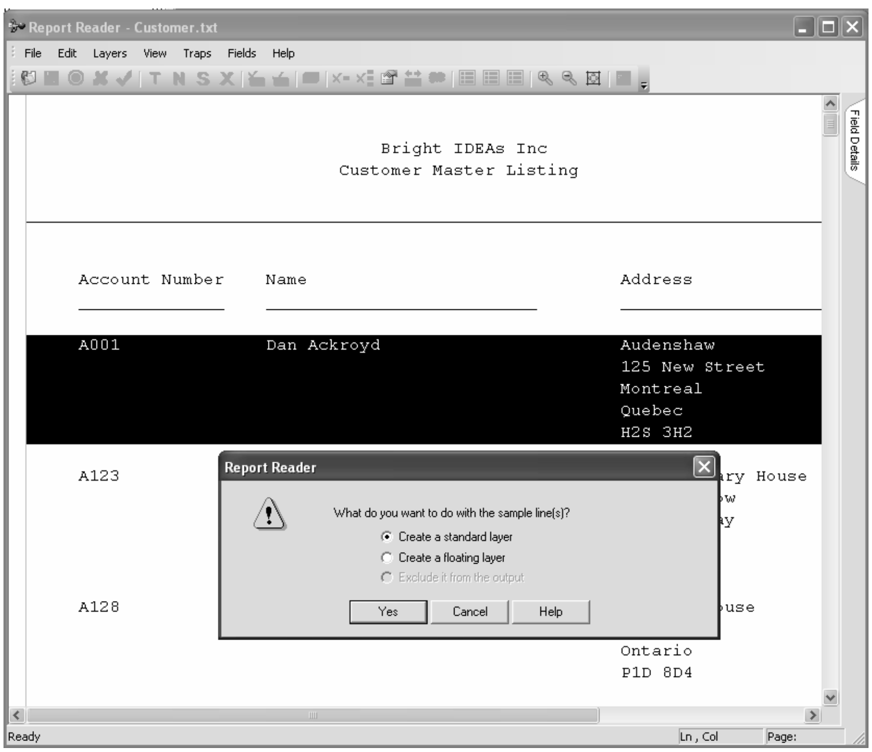
Once the **Base Layer** is defined, **Append Layers** that provide related information to the **Base Layer** can also be defined. The **Base Layer** and **Append Layers** are identified by **Traps**.

Traps can be Text, Numeric, Space, Non-Blank, Floating, or actual words that uniquely identify each occurrence of the record in the report. Keep the **Traps** as simple as possible. Do not overlook the possibility that spaces or spaces combined with numbers or text could be the best **Traps**.

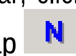
Define a **Base Layer**:

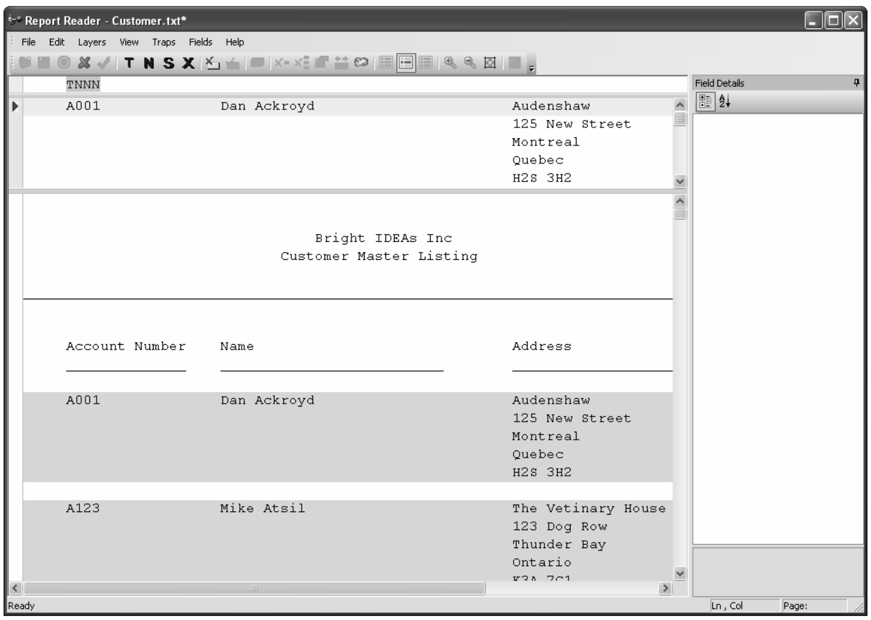
11.6.11 Scroll through the report and identify the line(s) that contain the lowest level of detailed information and highlight the lines with the cursor (click and drag). Since the address information is a multiple line field, the **Base Layer** will contain 5 lines.

11.6.11 Select **Create a standard layer**, and then click **Yes**.



The **Report Reader** will display a **Field Editor** (the section between the two yellow lines) where the fields that need to be included in the database are defined. Scroll through the report and notice each section of data begins with an account number. The best **Trap** for the **Base Layer** is a combination of text and numbers, based on the account number (TNNN).

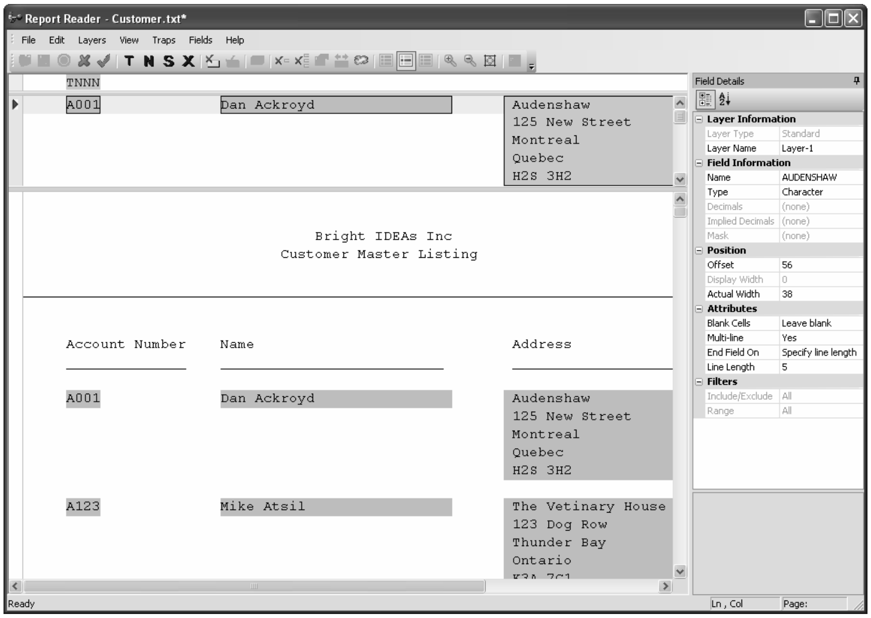
11.6.12 Place the cursor in the **Anchor Editor** area above the account number in the **Field Editor**. On the **Report Reader** toolbar, click the **Text Trap**  button once and click the **Numeric Trap**  button three times.



11.6.13 Notice each section of data that begins with an account number is highlighted. Scroll through the report to make sure all the information is highlighted.

Identify **Field Anchors**:

Selecting the information that will become the data fields in the database is done by identifying **Field Anchors**. Using the cursor, highlight each **Field Anchor** contained in the **Base Layer**. Scroll to the right to include the values for credit limit. Make sure the widths are large enough to include information that may be slightly off position or extremely long. Watch for items that appear as spaces in the **Base Layer** line, but contain values in other lines.



Technical Notes

If a mistake is made while highlighting the fields, click the **Remove Field Anchor**  button on the toolbar to remove the **Field Anchor** or right-click the **Field Anchor** to display a shortcut menu. Both actions remove the **Field Anchor** that is active and highlighted.

Now that each **Field Anchor** in the **Base Layer** has been highlighted, it is necessary to name and detail each field. This is done using the **Field Details** window.

11.6.14 Click the **Save Layer**  button on the toolbar.

11.6.15 The **Field Details** window appears that contains the data values that were in the **Base Layer**. This information will need to be changed. The names that originally appear (for or in?) the data in the **Field Details** window will be the **Field Anchor** values of the data in the database. Notice that the fields are listed in the order that they were highlighted in the **Base Layer**. The order of the fields is easily changed once the data is imported into IDEA.

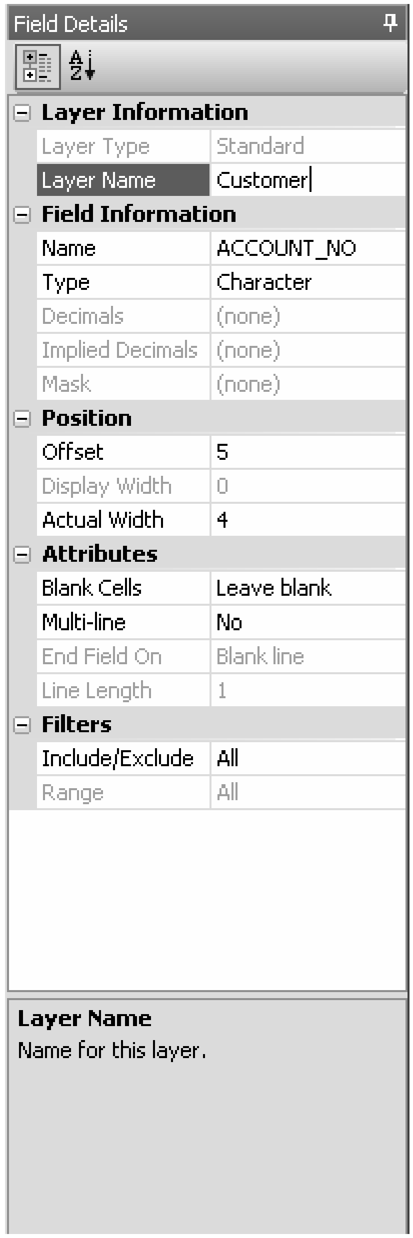
11.6.16 Use the following chart to update the **Base Layer** information. Select each field and change the name and details.

Field as shown in Field Details	Name	Data Type	Offset	Actual Width
A001	ACCOUNT_NO	Character	5	4
DAN_ACKROYD	CUST_NAME	Character	23	28
AUDENSHAW	ADDRESS	Character	57	33
20000	CREDIT_LIM	Numeric	95	7

ADDRESS is a multi-line field. Make sure that the multi-line field option is selected and that the field is 5 lines long.

11.6.17 In the **Layer Name** box, delete the default layer name (Layer-1) and enter **Customer**.

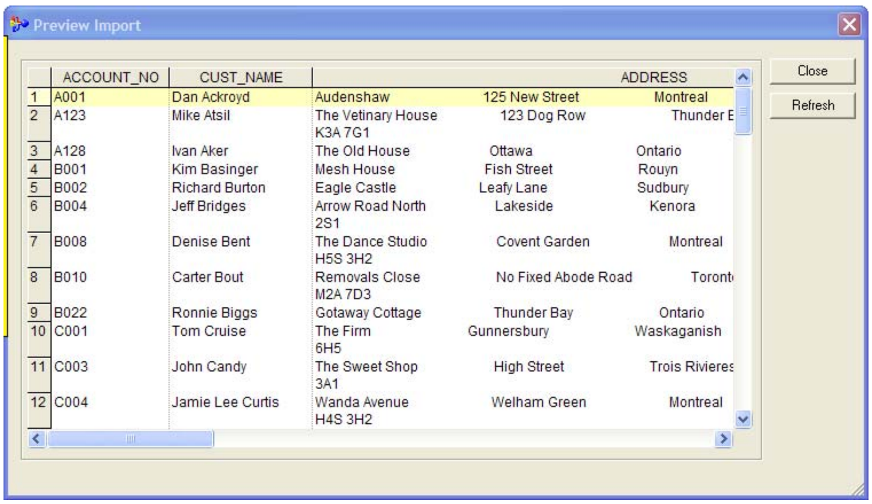
The **Field Details** window should look like the following example.



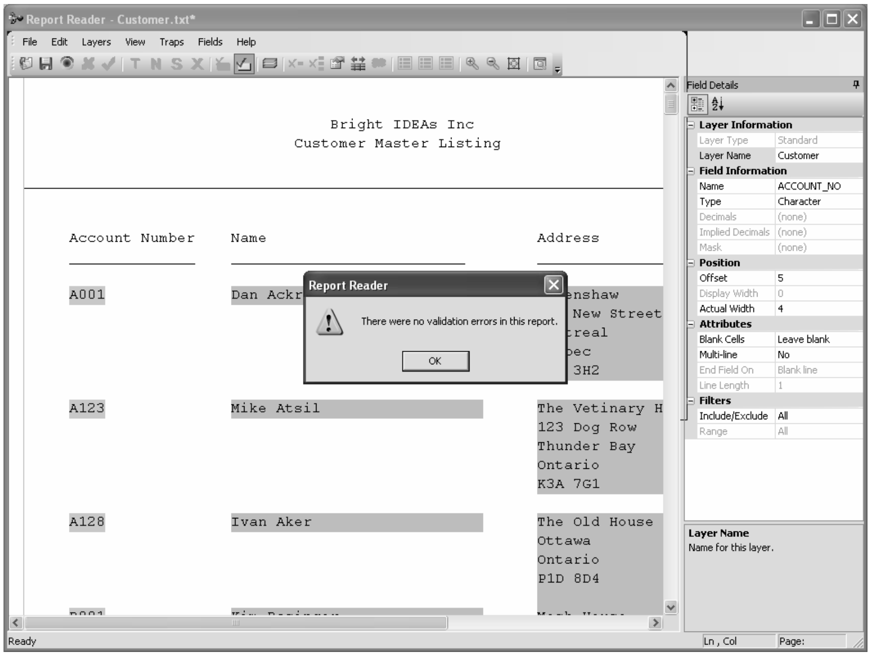
Since all the required information can be defined in the **Base Layer**, **Append Layers** will not be defined for this exercise.

Importing into IDEA:

11.6.18 Before importing the file into IDEA, it is recommended that we preview the entire database. Once the **Base Layer** has been defined, the IDEA database can be previewed at any time by clicking the **Preview Database**  button on the toolbar. The preview should look like the following example.



11.6.19 Before importing the file into IDEA, it is also recommended that we run the **Scan for Errors** task. **Scan for Errors** identifies any items that may not import properly such as fields where the length is too short, fields where there are excessive amounts of space, and fields that are not in alignment with the preceding fields. Click the **Scan for Errors**  button on the toolbar.



11.6.20 Once the **Scan for Errors** validation is complete, save the **Report Reader** template for future use with similar reports. Select **File – Save Template As**. Give the template a name and save it to the **Import Definitions sub-folder under the project folder**. The **Import Definitions** is the preferred location unless the templates are shared items and need to be stored elsewhere.

11.6.21 Click the **Import into IDEA**  button on the toolbar.

11.6.22 Click **Yes** to proceed with the import.

11.6.23 Select the **Generate field statistics** check box.

11.6.24 Name the database **Customer Master** and then click Finish.

Chapter 12
Working with Application Softwares

12.1 Introduction

CAAT – IDEA is a robust, time-tested, stable, data analytical tool. IDEA has evolved steadily to integrate conveniently and seamlessly with well-known Application Softwares worldwide.

IDEA connects well with simple user-friendly Accounting Packages like Tally, ACCPAC, SAGE and Peach Tree. It also gels well with more complex Applications like ERP Applications – SAP, and Oracle. In the Banking Sector, IDEA mixes well with Core Banking Applications like Finacle, Flex Cube and FNS. In the General Insurance sector, IDEA serves well with Applications like Genisys.

The sections that follow illustrate through actual experiences the import and analysis/interrogation of data in diverse Application environments and set-ups.

We have depicted connectivity to

- SAP and SAP Flat Files from the Manufacturing Industry
- Core Banking System – Finacle from the Banking Industry
- Web Based Application Systems – Sabre from the Airlines Industry

12.2 Core Banking System – Finacle from the Banking Industry

12.2.1 Import of Data

Set-up of ODBC environment for linking IDEA to Oracle database for FINACLE Core Banking System (CBS)

- Install IDEA V 9 Electronic License on the designated client machine.

- With the help of the CBS implementation partner, load the relevant Microsoft for Oracle client drivers on the above machine. Note the drivers should match with the Oracle database version
- Update the Internet Protocol Address and Port Number of the destination server in the "TNS Ora" file on the client machine.
- Create an ODBC Data Source (DSN) to the Flash, Training or DR site server as decided. In this case, we connect to the training server 'testbanktraining'.
- To create the DSN, name the DSN connection, input the server name correctly, we select the relevant drivers and enter the user access ID and password for gaining entry to the server.
- Once the DSN has been created, it will appear in the list of 'Select an ODBC driver that is configured to connect to a specific data source:' within IDEA > Import Assistant > Import into IDEA > ODBC.
- In this case, our user access ID is 'testbankuser', password is 'testuser123' and the training server is 'testbanktraining'. So 'testbanktraining' will appear in the list of 'Select an ODBC driver that is configured to connect to a specific data source:'
- Each time the user navigates through the above-mentioned DSN and by entering the password for access, the user can view (read only access) to all the authorized data tables in the server. The tables which the user can view are the ones to which the user has view right access.
- By selecting a specific data table, the user can navigate to 'Advanced' and choose to select specific fields to download while importing the data. The user can also choose to enter an SQL statement, so as to import data for a particular period, branch or some other parameter.

12.2.2 Interrogation of Data

For example, in Deposits –Term Deposit domain, we choose to filter cases where deposit period has exceeded a tenor of more than 120 months (Maximum Tenor of a Term Deposit being 120 Months as per Reserve Bank of India).

Here IDEA connects through the ODBC process mentioned above to the tables -

- GAM – Account Masters
- TAM – Term Deposit Masters.

The fields of reference relevant to the objective being tested which may be chosen through the Advanced Options in IDEA ODBC Import are

- GAM – ACID (Account Number), FORACID (Customer ID)
- TAM – ACID, Deposit_Period_Mths (Deposit Tenor), SOL_ID (Branch Number)

The process of interrogation followed in IDEA:

- Import GAM with the fields of reference.
- GAM has 14 Lac rows. The ODBC takes 5 seconds.
- Open GAM
- Write a criteria where SOL_ID=”00015” where 00015 represents large Branch X.
- Save the criteria as a database within IDEA. **Name it** X
- Import TAM with the fields of reference.
- Open TAM
- Write a criteria where SOL_ID=”00015” where 00015 represents large Branch X.
- Save the criteria as a database within IDEA.
- Perform a direct extraction on the above with the criteria Deposit_Period_mths>120.
- Save the database as **Greater than 120**
- Now join **Greater than 120** and X based on matching field ACID, using the Join option “Matches only”.
- The resultant database **Filename Objective** 4 has given us a list of instances of Term Deposit accounts accepted with tenor over 120 months.

12.3 Web Based Application Systems – Sabre from the Airlines Industry

12.3.1 Introduction

Nowadays we can book air-tickets very conveniently from our home or office. This is possible with the help of Internet based Air Reservation Application System, Sabre. In fact, advanced versions of the Internet Application allow users to check-in their seat through Mobile Short Messaging Service (SMS).

As an application system Sabre generates reports in an Extendible Mark Up (XML) format. The XML data format is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. Nowadays many applications are generating reports in these formats for easy portability and recognition across platforms and applications.

The Airline Reservation System generates Sales Reports and other relevant Flight Capacity Utilization Reports.

These Reports contain the following fields:

- Passenger First Name
- Passenger Last Name
- Address
- Contact Number
- Email
- Destination From
- Destination To
- Basic Web Fare
- Taxes and Levies
- Other Charges
- Payment Information and any other required information

12.3.2 Import of Data

The XML Data Files once saved on the workstation containing IDEA or on the Local Area Network in a shared audit data folder can be accessed through IDEA's Import Assistant AML component. The process of import is simple and easy to apply, since XML formats are readily recognized by IDEA just like importing a spreadsheet file.

12.3.3 Interrogation of IDEA

With the recent credit card fraud in the Airlines Industry where a noted carrier took a revenue hit of 18-20 Crores due to criminal usage of stolen credit cards through web-bookings, Airlines are now closely monitoring Passenger Personal particulars along with their Credit Card particulars.

In this case, the Passenger Personal particulars:

- First Name
- Middle Name
- Last Name

These are extracted from the Sales Account for a given period of flight reservations.

This information is matched with the Credit Card particulars:

- First Name
- Last Name

The match is done through a superior @function in IDEA titled @soundslike.

@soundslike throws up exceptions where names or character strings like addresses match each other not necessarily on syntax but on mnemonics (Sound and Rhyme).

The seemingly matching cases are extracted and taken up for substantive testing to check the bona fide of the cases. Special care is taken to identify group bookings, family bookings, geographical concentrations, repeat bookings, and incomplete names.

12.4 SAP R/3 and SAP Reports from the Manufacturing Industry

12.4.1 International Practice

In certain countries like Germany and Canada, advanced statutory measures have been put in place by the Ministry of Revenue for usage of CAATs with SAP ERP. Clients are mandated to generate reports from SAP - MM, FI, SD, FA Modules in a specific format to meet specific Excise and Customs (Revenue) Audit objectives. This mandate provides a stable platform for the growth of CAATs.

12.4.2 National Practice

Manufacturing majors – IDEA Clientele using IDEA with SAP in specific areas follow the following approach to integrate it with SAP. The practice within India is similar to the practice outside India:

1. The Audit User can generate standard reports through his login ID from SAP.
2. These reports can be generated either from the standard report menu OR

   In case the Company has procured and enables AIS – Audit Information System along with SAP (The audit information system (AIS) is an auditing tool that you can use to analyze security aspects of SAP NetWeaver Application Server (SAP NetWeaver AS) for ABAP system in detail.

   AIS presents its information in the audit infomation structure (similar to the Implementation Guide) so that you can easily determine which activities you need to perform and which you have accomplished.
3. These reports once generated can be saved either as an unconverted file (Text File) or spreadsheet. It is recommended to save the file as an unconverted file if it is larger than a standard spreadsheet row length capacity of 1 million odd rows (Excel 2013).
4. Once saved, these text files or spreadsheet files can be conveniently imported into IDEA using IDEA's Import Assistant.

Merits of the Report Approach are:

- Easy to apply
- Capture meaningful data in report forms
- Generally sufficient to express management assurance on audit objectives

Demerits of the Report Approach are:

- Reports may not reflect the right data

- Reports are a personalized view of data; raw data is comprehensive and cannot be manipulated

12.4.3 Theoretical Approach: Complex and Seldom Followed

IDEA can be integrated directly with the SAP server provided

- The IT department is convinced on the software and non-integrity violation of data in the backend.

  In one of our recent client acquisitions, a leading Engineering and Projects Company decided to opt for IDEA since it has a robust/stable SAP Report Import Component. The General Manager – IT in the subject Company was not for IDEA connecting to the Live or DR Server on grounds of security and confidentiality of production data. Given this concern, IDEA was implemented on Spreadsheet and Unconverted Report Files from SAP.

- The Audit user is knowledgeable about the data tables in the backend, and its several fields. SAP has over 60000 odd physical tables in its back-end. It is a challenge linking the right table for the right objective.

  The following methods can be used to identify the tables within SAP

  - Technical Information button within a transaction
  - Where Used List for Data Elements
  - Where Used List for the Domain
  - Search on Fieldname / Partial Fieldname within table DD03L
  - Search on Fieldname / Partial Fieldname within table DD08L
  - SE16 fuzzy Search on Table Descriptions
  - System Trace

For example: SAP Server - MM module contains tables like EKPO or EKKO which represent Header and Transaction detail tables for Ordering.

Once the integration is approved, the user can establish a link to the SAP Server through the middleware – ODBC, using the Oracle drivers for SAP.

The desktop having IDEA is linked to the server, with due access authorizations, tables viewed and subsequently imported into IDEA. This is the raw data being imported.

Merits of the Direct Link:

- Pure unaltered Data can be mined from the server.

Demerits of the Direct Link –

- To arrive at a logical representation of data, many data tables may have to be joined to arrive at a table having more or less all the logical fields necessary for achieving audit objectives. This will necessitate that the Audit User has a thorough knowledge of each and every table and field for file joins.

Chapter 13

Application of CAATs to General Insurance

13.1.1 Introduction

IDEA Data Analysis Software can play a very vital role in the Insurance industry in delivering enhanced, effective and value added results in a significantly short span of audit execution time.

With significant de-regulation in the Insurance sector, widespread availability of information through integrated insurance application systems and heightened awareness about Audit Software, concerted efforts need to be made to spread the use of the tool amongst insurance auditors, academicians and specialists like risk underwriters and loss prevention teams.

This research publication is an attempt to create awareness about the use and application of IDEA to realize simple, common, and generally accepted audit objectives of the General Insurance industry.

Some assumptions made during this project have been duly mentioned against each research objective. The publication is an attempt to generalize the objectives with reference to common practices across the globe.

Objectives have been established for the main streams of activity covering premium and claims. Analysis of objectives and financial tests have also been explored.

In Premiums, the objectives dealt with include establishing the accuracy of premiums collected, authorizing discounts offered, validating manual policies prior to system integration, justifying policies canceled shortly after inception and accuracy of penalties levied in case of repetitive past claim history.

In Claims, objectives dealt with include establishing the validity of claims, validating close proximity claims, justifying operational delays in claim settlement, accuracy of agency commission and settlement of claims on policies underwriting risks for stolen motor vehicles.

Certain aspects of analysis include rolling comparison of incurred losses over three years to identify cases for premium enhancement in case of repetitive claim history. Development officer business solicitation practices have also been studied in case of high loss policies over time.

Finally, certain financial tests have been undertaken to establish the accuracy and validity of the system of balances and postings between sub ledgers and general ledger control accounts.

13.1.2 Potential Audit Tests

INSURANCE PREMIUM

The main tests are concerned with ensuring that all premiums are raised and the correct rate is used. However, analysis of risk and exception tests of old or forward dated items are useful.

Typical tests include:

Calculations

Recalculate premiums

Prove posting totals

Analysis

Stratify premiums by size and product type

Summarize new business by agent or area

Summarize debts by agent

Exception Tests

Extract old premiums due on policies that have not lapsed

Identify policies that were cancelled soon after inception

Identify policies with initial discounts that have then been lapsed and credited

Extract negative debts

Identify unallocated cash

Extract policies with blank or invalid policy numbers

Identify policy premiums that have only been part paid

Direct Debits that have not been collected

INSURANCE CLAIMS

Claims are a classic payments area. Claims can be matched back to policy details and there are usually various claim types, reason codes and risk codes.

Calculations

Cast (or total) outstanding claims

Prove posting totals

Analysis

Analyze by month and type and calculate average values

Group claims by product type

Matrix analyze date of incident to date claim reported as part of assessing claims incurred but not recorded

Exceptions Tests

High value claims

Identify claims with no movement during the last 6 months (some claims get reported but queries on the claim are not responded to leaving old expired claims on the system)

Identify negative balances (recoveries)

Gaps and Duplicates

Check for duplicate claims (same policy number and either same amount or same date of incident)

Check for multiple claims on the same policy (some policy holders are particularly unlucky; however, others make a business out of claiming more than once for an incident. This can be picked by testing claims for the same amount or the same date of incident)

REASONABLENESS TESTS ACROSS FILES

Existence of a policy in the same time period as the claim

Checking details between claims and policies

New policies experience claims shortly after inception

HEALTHCARE FRAUDS

Excessive procedure billing of same diagnosis, same procedures

Identify excessive number of procedures per day or place of service per day / per patient

Identify multiple billings for same procedures, same date of service

Analyze for mismatched services to diagnosis codes

Review diagnostic fees in excess of masters

Review procedure fees in excess of masters

Identify and report diagnosis and / or treatment that is inconsistent with a patient's age and / or gender

13.1.3 Obtain the Data

The following data files for the period **April 1, 2005 – Oct 31, 2005** will be used for this exercise:

1	Premium Report for Motor Class (Tariff)	Spreadsheet Excel File
2	Premium Report for Earthquake Class (Tariff)	Spreadsheet Excel File
3	Premium Report for Motor Class (Non-Tariff)	Spreadsheet Excel File
4	Premium Report for Motor Class (Non-System)	Text File
5	Canceled Policy Report for Motor Class	Spreadsheet Excel File
6	Cover Note Report for Motor Class	Spreadsheet Excel File
7	Premium Master for Motor Class	Dbase File
8	Premium Master for Earthquake Class	Dbase File
9	Discount Master for Motor Class	Dbase File
10	Policy Number Report for Rolling Policies Motor Class	Spreadsheet Excel File
11	Agent Commission Report for Motor Class	Spreadsheet Excel File
12	Commission Master for Motor Class	Dbase File
13	Claim Report for Motor Class	Text File
14	Stolen Vehicles Report from the National Crime Records Bureau (NCRB)	Text File
15	Surveyor Activity Report for Motor Class	Spreadsheet Excel File
16	Incurred Loss Summary Report for 2005	Spreadsheet Excel File
17	Incurred Loss Summary Report for 2004	Spreadsheet Excel File
18	Incurred Loss Summary Report for 2003	Spreadsheet Excel File

19	Incurred Loss Detail Report for 2005	Spreadsheet Excel File
20	Business Performance Report for 2005	Spreadsheet Excel File
21	Incurred Loss Report Agent wise for 2005	Spreadsheet Excel File
22	Sub Ledger for Cash Deposit Account	Dbase File
23	General Ledger for Cash Deposit Account	Dbase File
24	Healthcare Report on Healthcare Claims	Spreadsheet Excel File
25	Healthcare Master for Healthcare Claims	Spreadsheet Excel File

The reports cited above have been generated from the Insurance Application Software - M.I.S. function and have been captured in different data file formats.

13.1.4 Audit Program

Test Ref	Auditing Procedures	Objective	Page Ref
	Premium		
1.0	Re-verification of premium collected for insurance against natural disasters like earthquakes.	Revenue Integrity	
2.0	Re-verification of premium collected for insurance against motor claims.	Revenue Integrity	
3.0	Re-verification of discretionary discount allowed on premiums in the non-tariff free market insurance segment	Revenue Integrity	
4.0	Verification of rating fields for special additional impose / penalty incase of frequent claim history on sensitive policies.	Revenue Integrity	
5.0	Input field checks on non-system generated or manual policies	Validity	
6.0	Identifying policies canceled soon after inception along with justification by endorsements.	Validity	
7.0	Input field checks on cover notes (temporary policies) for motor insurance and conversion to bona fide policies through receipts.	Validity	

Test Ref	Auditing Procedures	Objective	Page Ref
	Claim		
1.0	Identification of duplicate motor claims in partial loss policies	Validity	
2.0	Identification of duplicate motor claims due to irregularities and discrepancies in survey	Validity & Fraud Detection	
3.0	Isolation of policies experiencing claims shortly within risk commencement – close proximity	Fraud Detection	
4.0	Identification of claims without bona fide policy numbers within the system – orphan claims	Validity	
5.0	As part of operational studies and work flows identification of irregular time lags in claim settlement.	Operational Efficiency	
6.0	Re -verification of agency commission on business solicited	Payments Integrity	
7.0	Identification of motor insurance business claims settled on policies pertaining to stolen vehicles	Fraud Detection	
	Analysis		
1.0	Analysis by client for different class of risks insured	Business Studies	
2.0	Analysis by development officer and / or agent of insured losses	Business Studies	
3.0	Identification of premium enhancement cases for loss making and high risk policies	Business Studies	
	Finance & Accounts		
1.0	Reconciliation of Sub Ledger and General Ledger for principal control accounts	Integrity of Accounts	
2.0	Identification of un adjusted credits in Cash Deposit Account	Integrity of Accounts	

Test Ref	Auditing Procedures	Objective	Page Ref
	Healthcare Frauds		
1.0	Excessive procedure billing of same diagnosis, same procedures	Validity & Fraud Detection	
2.0	Identify excessive number of procedures per day or place of service per day / per patient		
3.0	Identify multiple billings for same procedures, same dates of service		
4.0	Analyze for mismatched services to diagnosis codes		
5.0	Analyze for mismatched services to diagnosis codes		
6.0	Review diagnostic fees in excess of masters		
7.0	Review procedure fees in excess of masters		
8.0	Identify and report diagnosis and / or treatment that is inconsistent with patient's age and / or gender		

PROJECT SET-UP

LOAD DATA

OBJECTIVE

To be able to load data, run IDEA 9 and set a projct folder for the audit within IDEA

EXERCISE DESCRIPTION

This exercise covers copying data into a folder, loading IDEA, setting a project folder and entering client properties that will be printed on all reports.

IDEA FUNCTIONALITY COVERED

- Setting a Project Folder
- Entering Client Properties

The following data files have been copied on the hard disk in the folder c:\Program Files\IDEA\Insurance Research

Control totals have also been provided for each of the files.

1	Premium_Source – Premium – Motor. xls	Spreadsheet Excel File	Sum Insured – 98,70,000 Net Premium – 2,17,464.37
2	Premium_Source – Premium – Earthquake. xls	Spreadsheet Excel File	Sum Insured – 5,82,95,000 Net Premium – 12,96,162.5
3	Premium_Source – Premium – Motor Non Tariff. xls	Spreadsheet Excel File	Sum Insured – 98,70,000 Additional Excess Impose – 1000 Net Premium – 2,18,464.37
4	Non system policies. txt	Text File	Sum Insured – 98,70,000 Net Premium – 2,17,464.37
5	Premium_Source – Canceled Policies. xls	Spreadsheet Excel File	Sum Insured – 98,70,000 Net Premium – 2,18,464.37
6	Premium_Source – Cover Note. xls	Spreadsheet Excel File	Sum Insured – 98,70,000 Net Premium – 2,18,464.37
7	Motor Premium Master. dbf	Dbase File	No. of Master Records – 6
8	Earthquake Premium Masters. dbf	Dbase File	No. of Master Records – 7
9	Discount Master. dbf	Dbase File	No. of Master Records – 6

10	Premium_Source – Rolling Policies. xls	Spreadsheet Excel File	No. of Records – 4
11	Premium_Source – Commission. xls	Spreadsheet Excel File	Net Premium – 2,17,464.37 Gross Commission – 9,366.03
12	Commission Master. dbf	Dbase File	No. of Master Records – 2
13	Claim Transaction. txt	Text File	Estimated Loss Amount – 3,29,19,750 Estimated Expense Amount – 4,40,000 Paid Loss Amount – 2,46,89,812.50 Paid Expense Amount – 2,42,000
14	Stolen Vehicles.txt	Text File	Number of Records – 10
15	Claim_Source – Surveyor Activity. xls	Spreadsheet Excel File	Number of Records – 19
16	Analysis – Incurred Loss 05. xls	Spreadsheet Excel File	Premium – 2,17,464.38 Claim – 8,59,980 Loss – 7,74,080
17	Analysis – Incurred Loss 04. xls	Spreadsheet Excel File	Premium – 2,17,464.38 Claim- 1,33,400 Loss – 61,112.5
18	Analysis – Incurred Loss 03. xls	Spreadsheet Excel File	Premium – 2,17,464.38 Claim – 1,21,00 Loss – 32,925
19	Analysis - Incurred	Spreadsheet Excel	Premium –

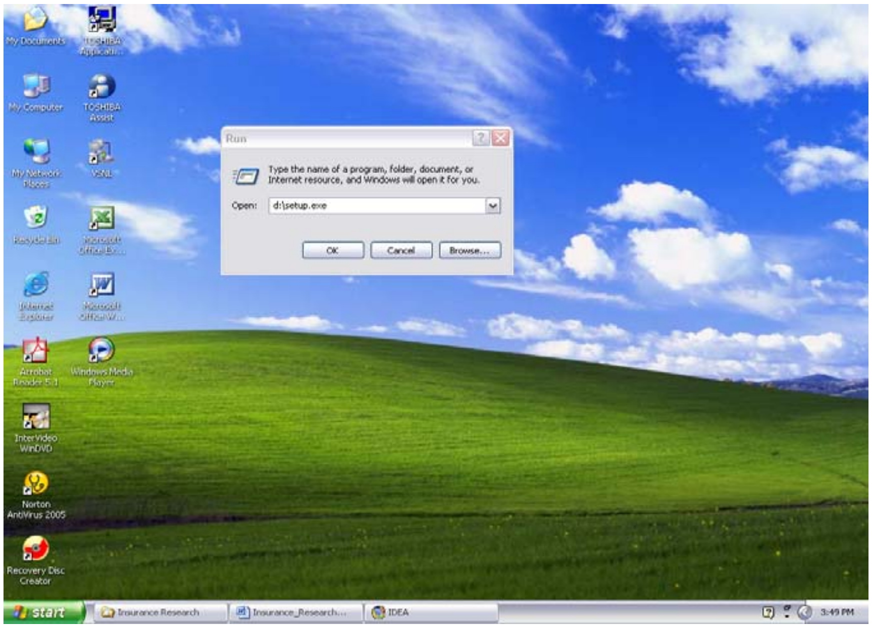
	Loss Detail 05. xls	File	6,17,808.75 Claim – 25,04,940 Loss – 22,49,115
20	Analysis - Business. xls	Spreadsheet Excel File	Premium – 6,17,808.75 Claim – 25,04,940 Loss – 22,49,115
21	Analysis – Incurred Loss Agent. xls	Spreadsheet Excel File	Sum Insured – 98,70,000 Net Premium – 2,17,464.37
22	Sub Ledger CD. dbf	Dbase File	Debit – 2,57,500 Credit – 8,50,000 Balance – 63,42,500
23	General Ledger CD. dbf	Dbase File	Opening Balance – 23,43,500 Debit – 2,57,500 Credit – 8,50,000 Closing Balance – 29,36,000
24	Healthcare Fraud Source Files – Healthcare. xls	Spreadsheet Excel File	Diagnostic Fees – 14,550 Procedure Fees – 10,40,000
25	Healthcare Fraud Source Files – Masters. xls	Spreadsheet Excel File	Diagnostic Fees – 5,250 Procedure Fees – 3,65,000

Throughout this Chapter we refer to the files located in c:\Program Files\IDEA\Insurance Research.

If the SETUP.EXE file has been run earlier, then there is no requirement to run it again.

Place the compact disk into the CD reader.

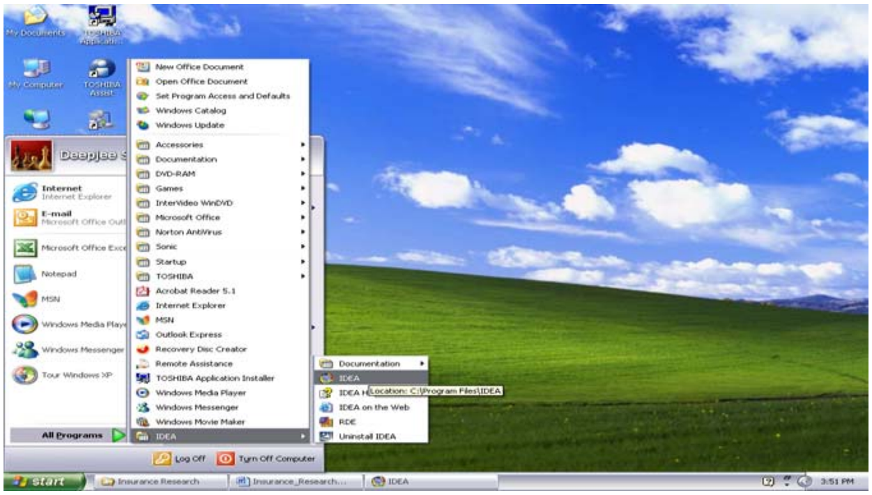
- Click the **Start** button
- Select **Run**
- Type **D:\SETUP.EXE.** (D: being the letter assigned to your CD reader)



The files are installed into the proper directory.

Accessing IDEA

From the Windows **START** Menu select: **Programs – IDEA – IDEA**



Alternatively, click the **IDEA** shortcut icon on the desktop, if available. IDEA will load.

Select a Working Folder and Enter Client Information

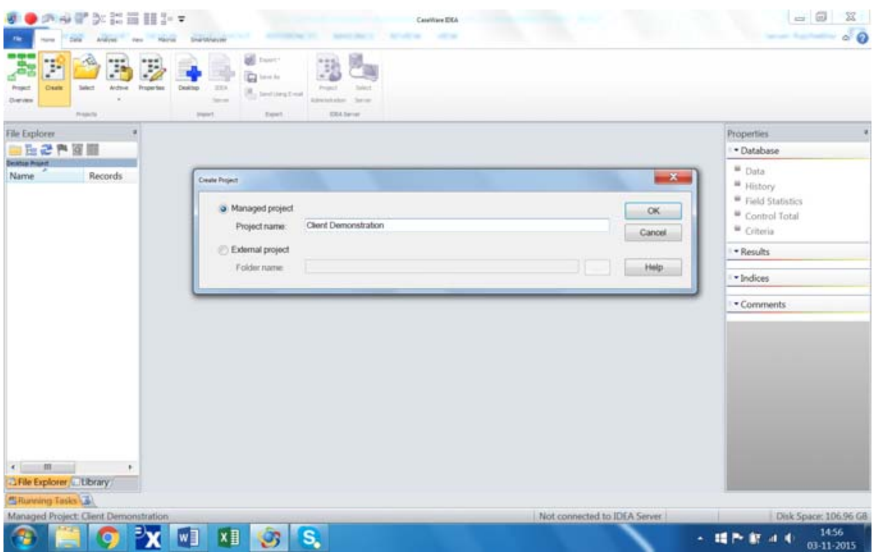
To facilitate housekeeping, it is recommended that a separate project folder be used for each audit / investigation.

All information relating to the audit, including data files, equations, views / report definitions import definitions must be stored in the project folder.

The project folder and client information will be printed on **all** reports. Once the project folder is set, it remains the active folder until changed.

1. To set the project folder, click the **Create** button under Home in the IDEA menu tool-bar:

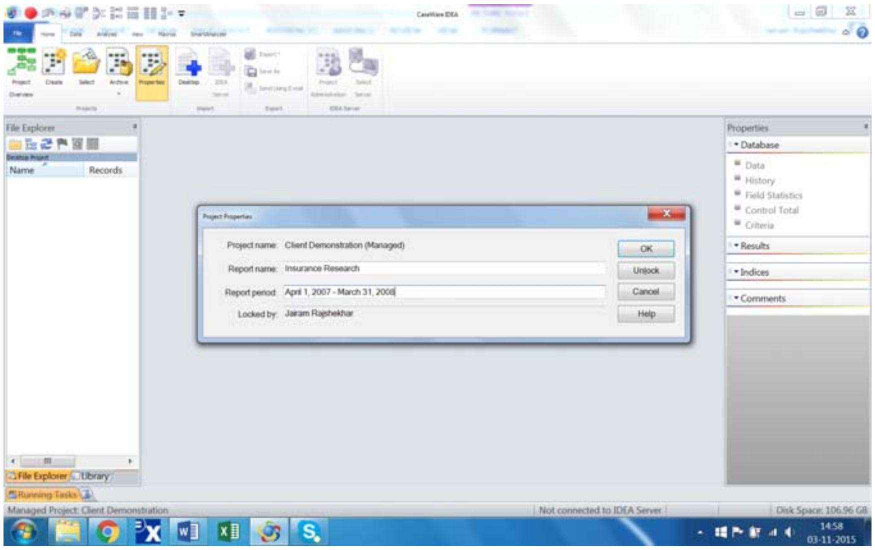
   Enter Client Demonstration as the Managed Project Name



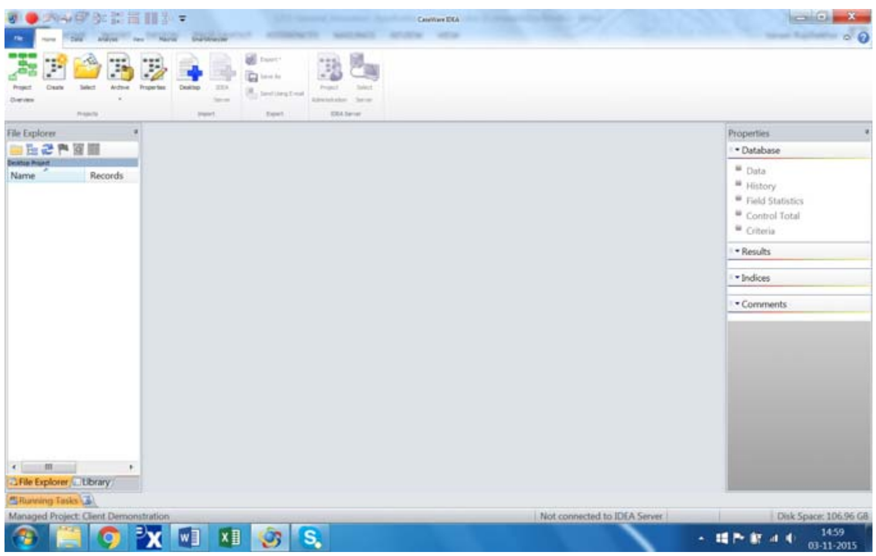
2. Click on **Properties** under Home in the IDEA menu tool-bar and enter the Report Name and Report Period:

   Report Name: Insurance Research

   Report Period: April 1, 2007 – March 31, 2008



Click the OK button. Client Demonstration will become the Managed Project Folder, as is seen from the working folder path at the left bottom corner of the IDEA screen.



13.1.5 Import of Data Files

OBJECTIVE:

To import data files for testing.

EXERCISE DESCRIPTION:

Three categories of data file types need to be imported: Spreadsheet Excel, Dbase and Text. Spreadsheet Excel and Dbase files can be directly imported into IDEA. Text files can be imported through the Report Reader, where a text file in report form is converted into a spreadsheet file in an array of rows and columns within IDEA.

IDEA FUNCTIONALITY COVERED

This exercise will cover:

- Import of Spreadsheet Excel files
- Import of Dbase files.
- Import of text files.

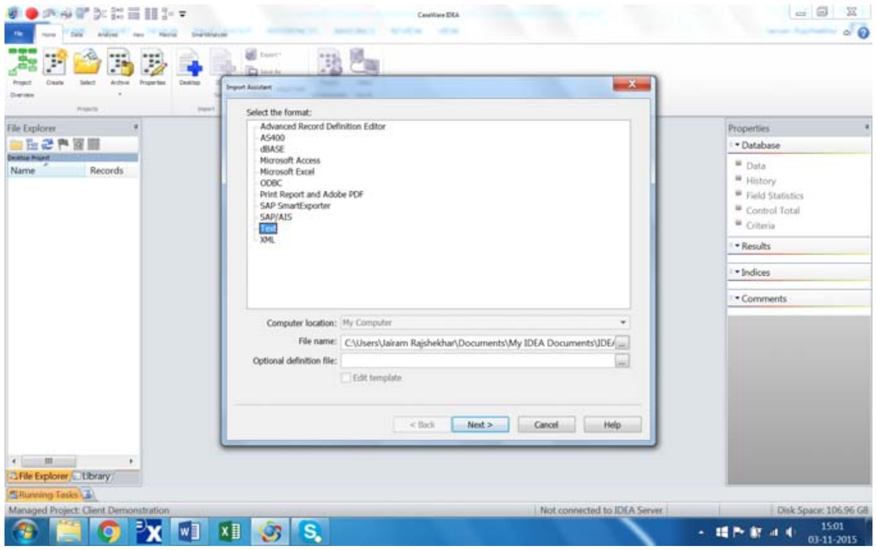
IMPORT OF SPREADSHEET EXCEL FILES

Access the Import Assistant by clicking the **Desktop** icon under Home in the IDEA menu tool-bar.

The Import Assistant loads to guide us through the process of importing the data.

Select Microsoft Excel.

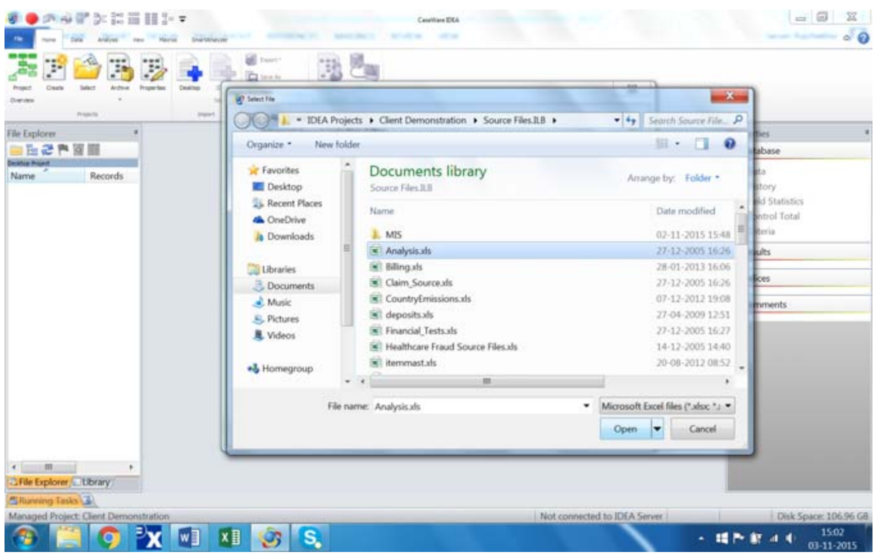
Click **Finish**



Click the **Input file name** box.

This takes us to the Source Files folder.

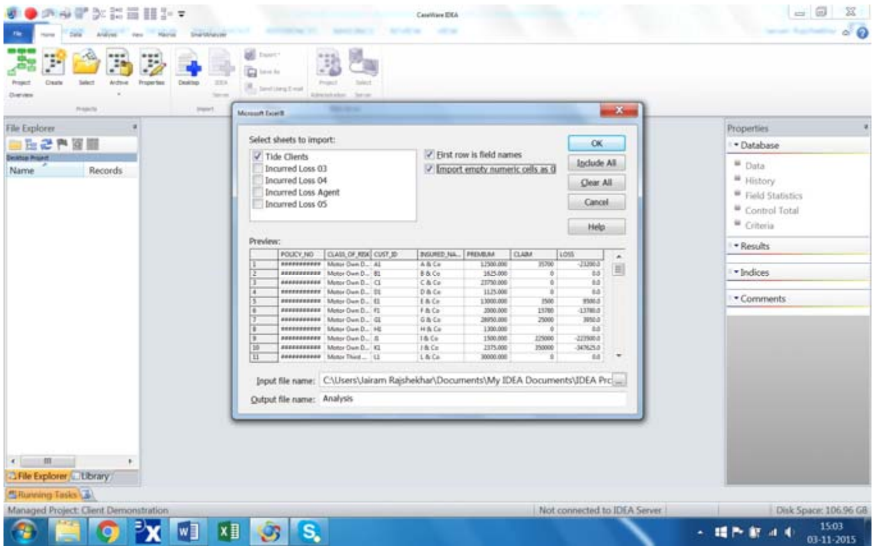
Select the MS-Excel file **Analysis**, and click **Open.**



The preview of the first worksheet in the file Analysis appears in the **Excel** box.

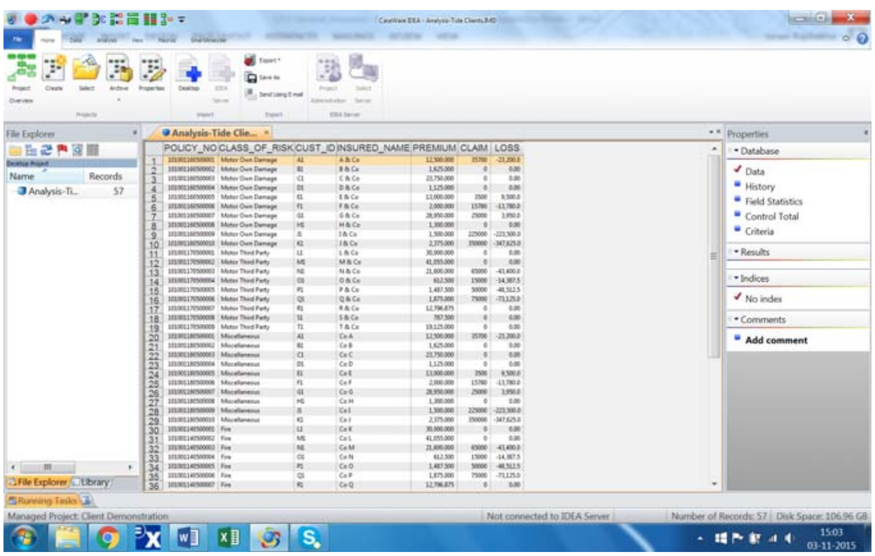
Check the box **First row is field names**, to enable the transaction headers in Excel to get captured in the IDEA headers.

Click OK.



The MS-Excel file once imported appears as below.

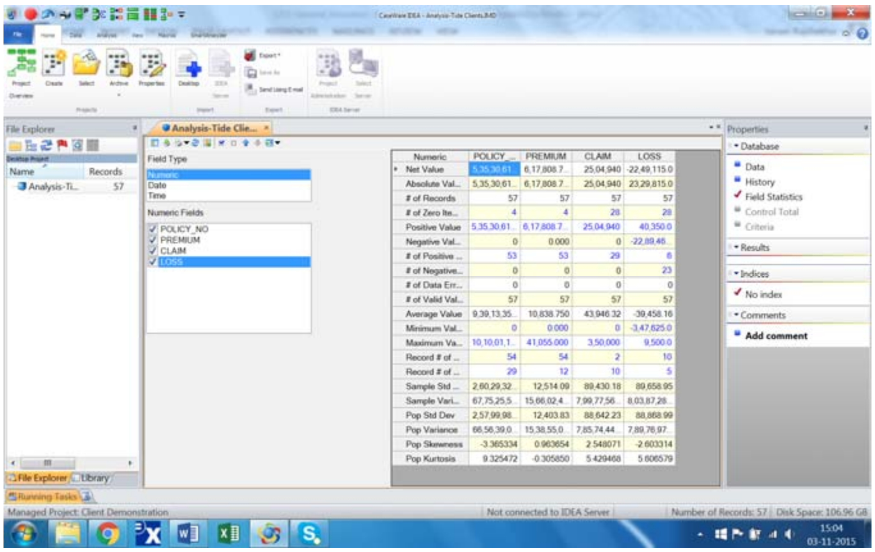
The name of the file in IDEA is Analysis – Business. IMD, Analysis is the name of the Excel file and Business is the name of the worksheet.



Invoke the **Field Statistic** Tab and view the **Net Value** of Premium, Claim and Loss fields.

A close comparison with the Control Totals, provided with the file, show that the totals match.

Now the file is ready for IDEA testing.



The process of import is the same for the other MS-Excel files intended to be used for various Insurance Audit Tests.

While undertaking the Insurance Audit Tests, only a passing reference will be made to the import of the MS-Excel files required for each of the tests.

IMPORT OF TEXT FILES

Access the Import Assistant by clicking the **Desktop** button under Home in the IDEA menu tool-bar

The Import Assistant guides us through the process of importing the data.

Click **Print Report and Adobe PDF.**

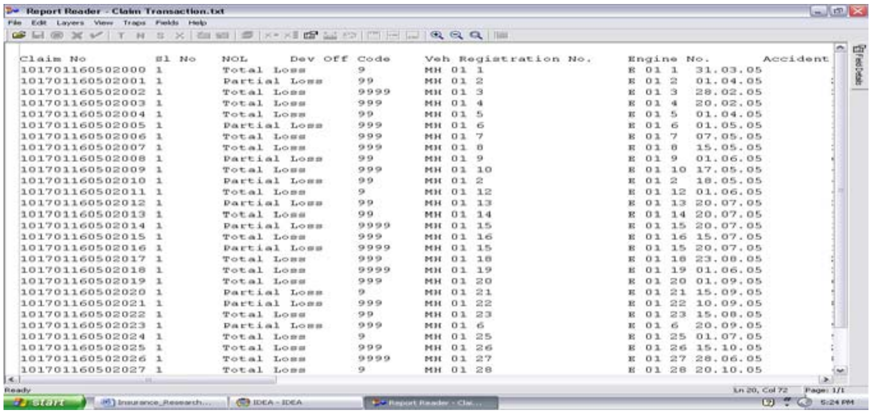
Select the text file to be imported from the Source Files folder by clicking on the select file browse box.

The **Select File** dialog opens, displaying the list of text files available in the working folder.

Select the file Claim Transaction.txt

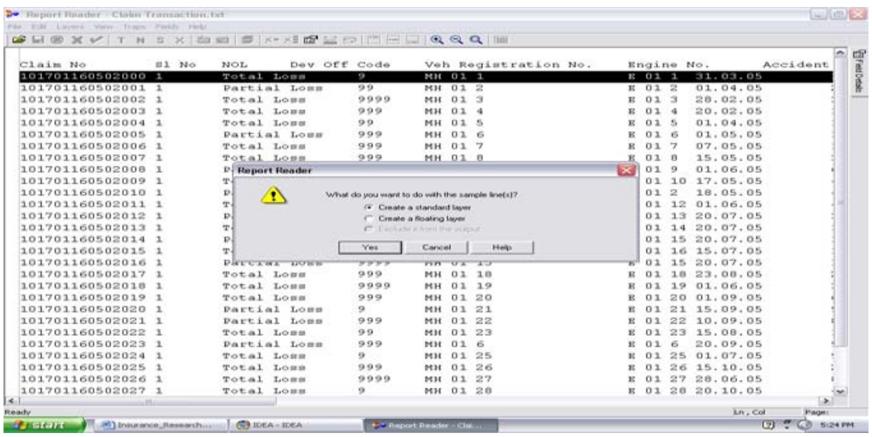
Click Next.

The text file opens in the Report Reader software.

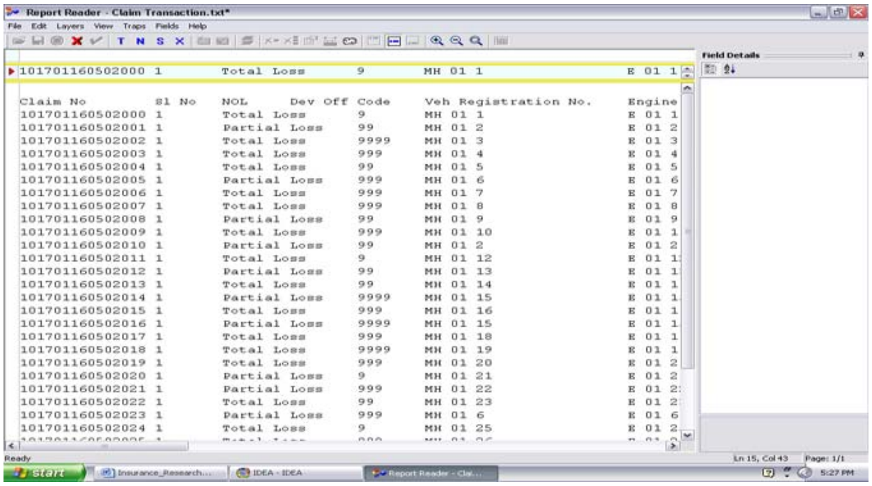


Select the first transaction in the text file as the base layer / standard layer.

Click Yes.



The first transaction is captured between two yellow lines called the Field Editor.



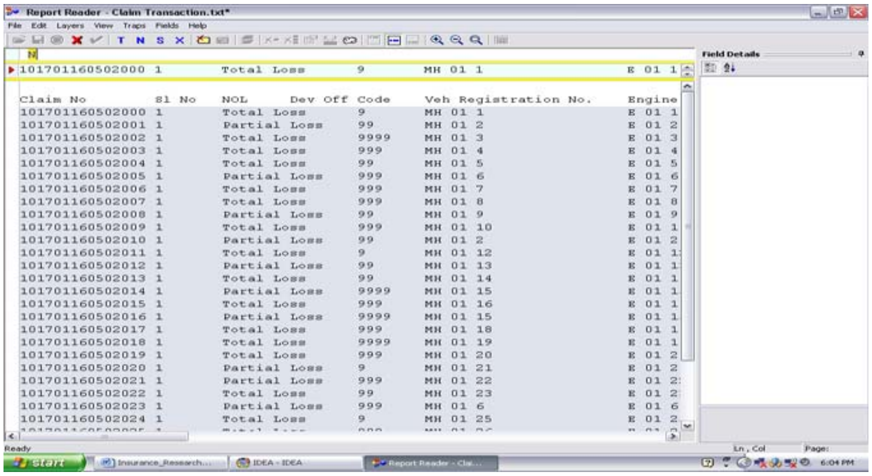
Specific Traps (Numeric N, Text T, Space S or Special Character X) can be set depending upon the data which needs to be trapped for output into IDEA. The Trap is a mask that identifies the common character or space vertically amongst all the records and picks only those records that meet the criteria of the Trap.

In the file under import, we can place a Numeric N Trap over the first digit of the Claim No. field.

The Trap is set by clicking N in the menu tool-bar of the Report Reader.

The Trap is set in the space above the top yellow line called the Anchor Editor, and the Traps are anchored to a specific layer.

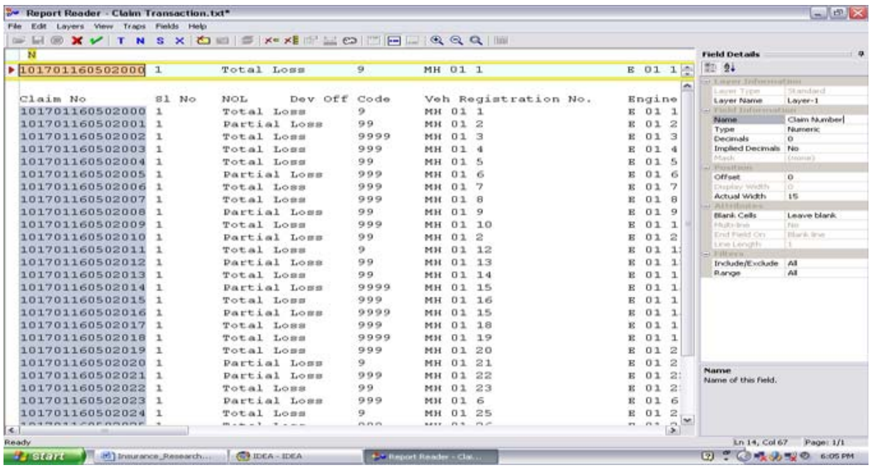
Upon setting the Numeric N Trap, the records which meet the Trap criteria are t highlighted in blue.



We then start selecting the fields that need to be present in the IDEA file.

After selection of each field, the **Field Details** appear on the right hand side of the screen.

In the Field Details – the Name and Type of the field is entered. IDEA by default sets the Type of the field based on the characteristics of the field. The field type can be changed from the default type to a user choice type.

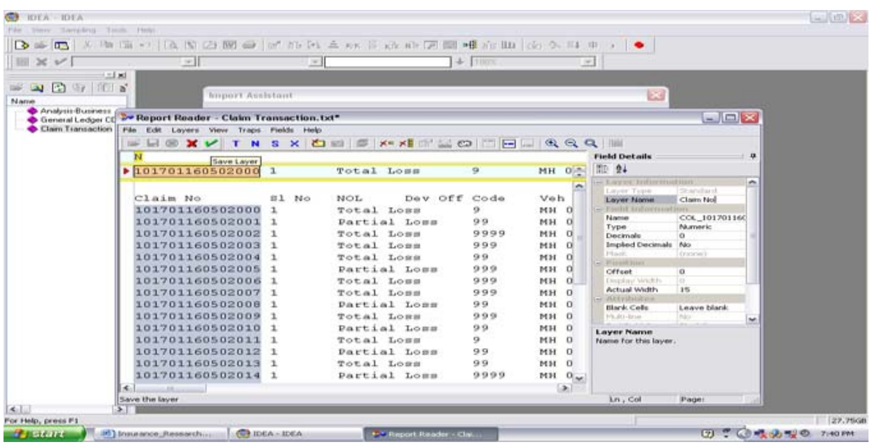


All the fields in the text file are selected and defined based on the record layout provided below –

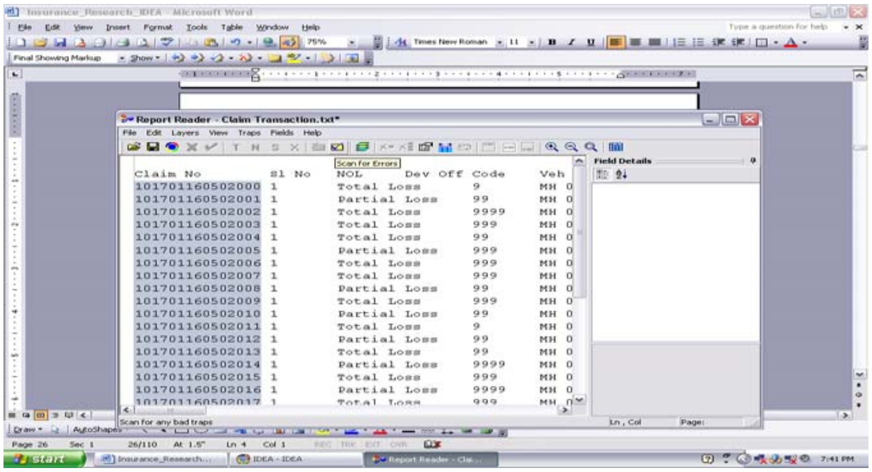
Field Name	Field Description	Field Type
Claim No	Claim Number	Numeric
SL No	Serial Number	Numeric

NOL	Nature of Loss	Character
Dev Off Code	Development Officer Code	Numeric
Veh Regn No	Vehicle Registration Number	Character
Engine No	Engine Number	Character
Accident Date	Accident Date	Date
ELA	Estimated Loss Amount	Numeric
EEA	Estimated Expense Amount	Numeric
PLA	Paid Loss Amount	Numeric
PEA	Paid Expense Amount	Numeric
V No	Voucher Number	Numeric
V Date	Voucher Date	Date
Policy No	Policy Number	Numeric
Receiver	Receiver	Character
Chq No	Cheque Number	Numeric
Chq Date	Cheque Date	Date

After selecting and defining the fields, click on the Green Check Button on the Report Reader menu bar. This saves the activities performed on the layer.

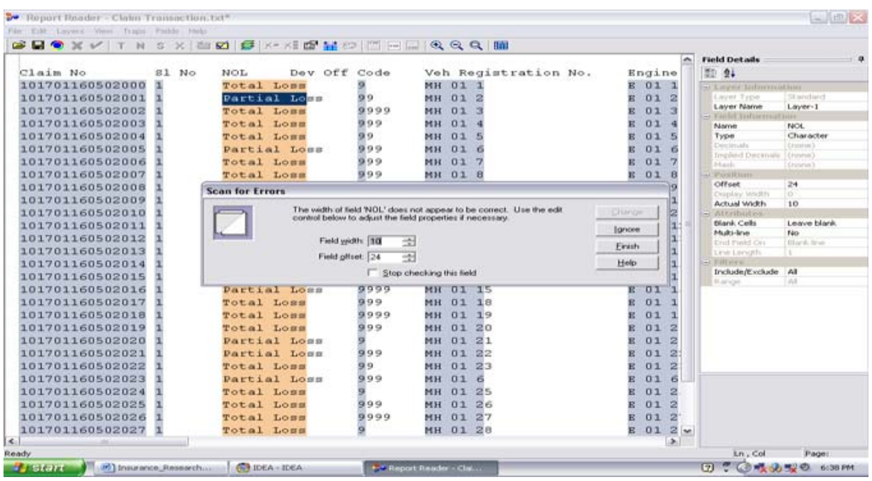


Then click the Scan for Errors Button on the Report reader menu bar. This invokes the Scan for Errors dialog, which corrects data which has been left out of the selection in the earlier step.

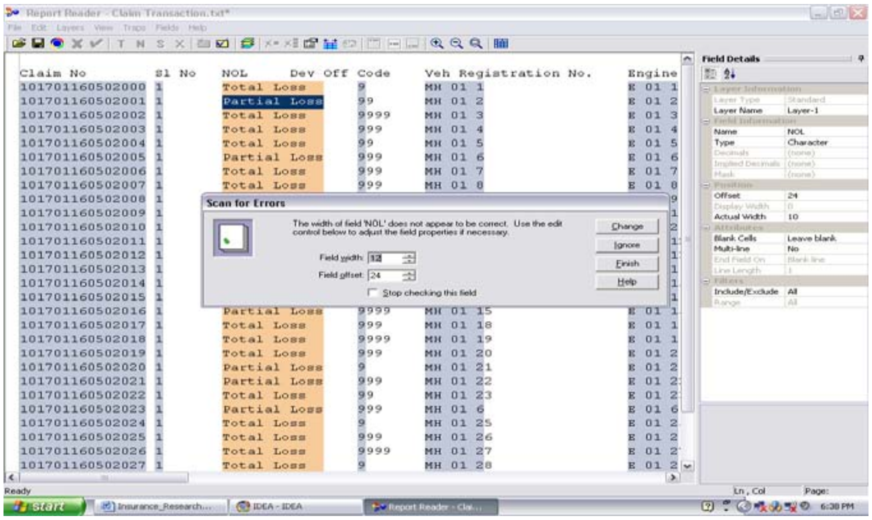


Increase or decrease the Field Width to adjust the right indent. Increase or decrease the Field Offset, to adjust the left indent of any field.

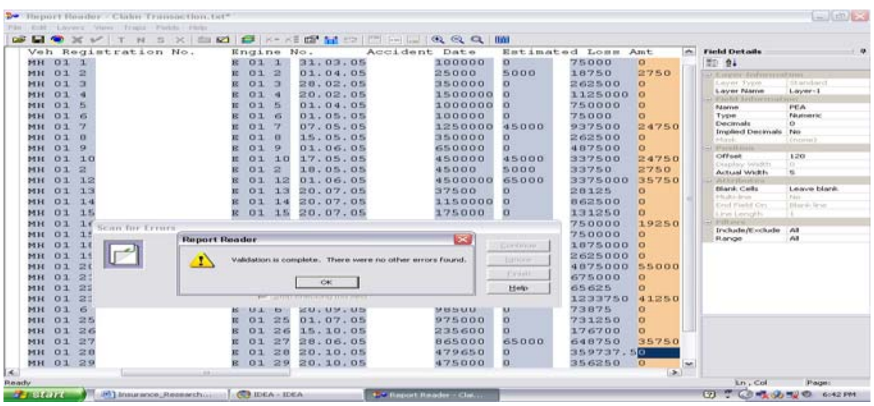
Notice the first case – Partial Loss (Seen Below). In this case, the letters 'ss' from Loss have been left out of the selection. Also notice that the Field Width for this specific cell is 10.



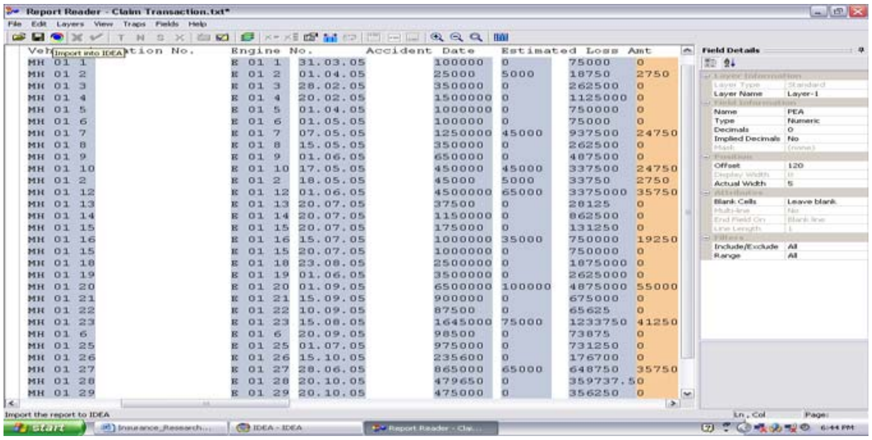
Notice that by increasing the Field Width, we have covered the whole cell and the whole word 'Partial Loss'



We continue scanning the file by adjusting the width and offset as shown above, till we get the message Validation is Complete. No other errors were found.

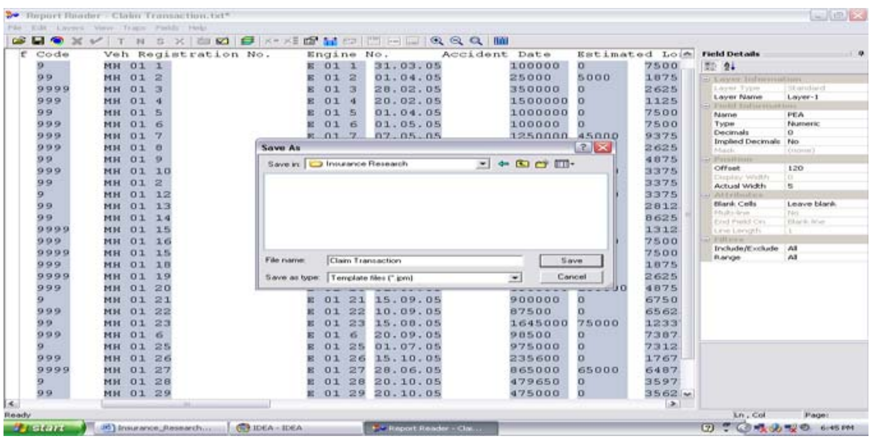


Click the button Import into IDEA



Report Reader prompts us to create and save a template for the file in the working folder.

This template allows the user to import similar files in future, with the use of the template without going through the Report Reader process.



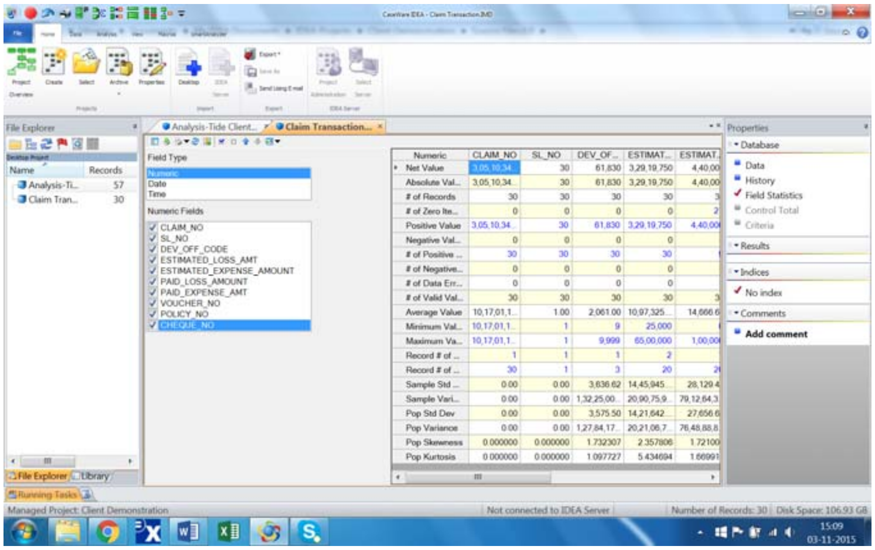
We proceed to the final screen 'Specify IDEA Filename'.

We may retain the name as reflected by the source file as default name or give a user defined name.

The Net Value for the fields ELA, EEA, PLA and PEA are viewed from the Field Statistics and reconciled back to the control totals provided with the text file.

The totals match.

We can now use this file for further testing.



The process of import is similar for the other Text files intended to be used for various Insurance Audit Tests.

13.1.6 Premium Case Studies

1.0 RE-VERIFICATION OF PREMIUM COLLECTED FOR INSURANCE AGAINST NATURAL DISASTERS LIKE EARTHQUAKES

OBJECTIVE

To verify whether the correct value of Premium on Earthquake Class of risks has been recovered from the insured as per the Premium Masters and the correct City Code.

EXERCISE DESCRIPTION

In this exercise we will join the Premium Transaction and Premium Master File based on the City and Location field respectively.

Append a computed field - virtual numeric field to re-calculate premium on sum insured by applying the master rate for the City to the Sum Insured.

Arrive at the differences between premium recovered and computed premium

Understand the reasons for differences if any in premium recovered.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

This exercise will show us how to:

- Join two files.
- Append computed fields – virtual numeric fields.
- Use the Equation Editor.
- Extract data with a specified criterion.

Data Import

Import the files Premium_Source – Premium – Earthquake. xls and Earthquake Premium Masters. dbf into IDEA as per the steps demonstrated in the session on IMPORT OF DATA FILES.

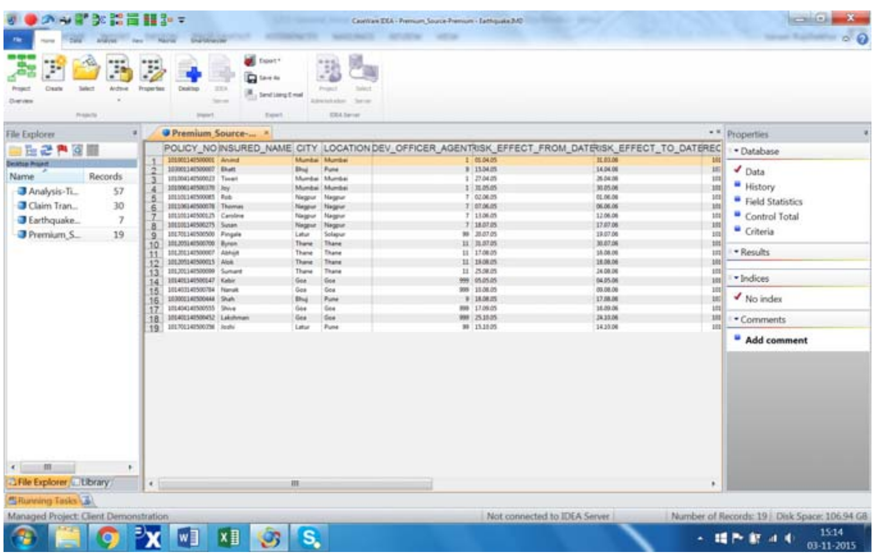
The files are imported as two independent IDEA files as seen in the File Explorer Section of IDEA.

JOIN TWO FILES

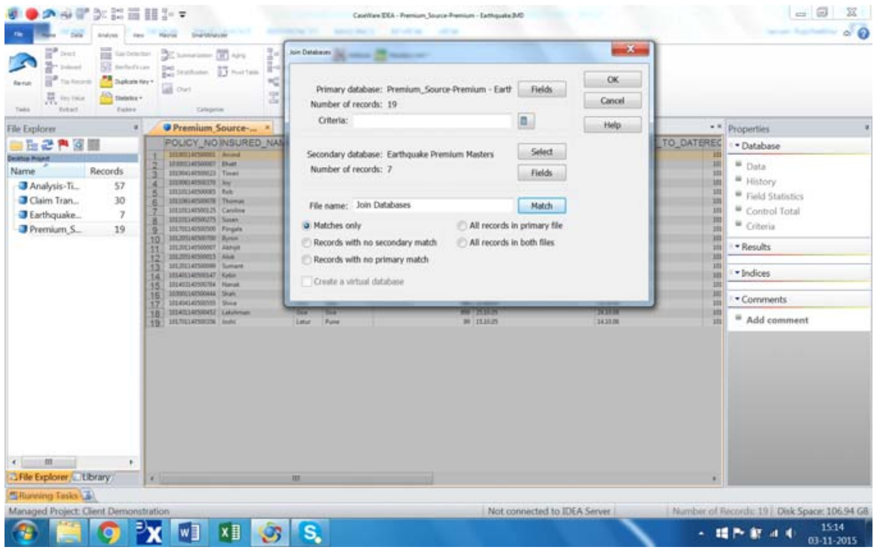
Designate Premium_Source – Premium – Earthquake. IMD as the Primary Database and Earthquake Premium Masters. IMD as the Secondary Database.

This is based on the logic that the premium masters are to be applied to the premium transaction file.

Open Premium_Source – Premium – Earthquake. IMD, and then click the Join Database option on the File Explorer Toolbar.



The Premium_Source – Premium – Earthquake. IMD is automatically seen as the Primary Database. Select Earthquake Premium Masters. IMD as the Secondary Database.



In earthquake insurance, the location is the decisive factor for premium application.

For example, Bhuj in Gujarat and Latur in Maharashra are sensitive seismic zones and command higher premium.

The Primary database has a field 'Location' and a field 'City'.

Both the 'City' and the 'Location' fields in the Policy Creation stage are to be selected from an index list of cities within the insurance application software.

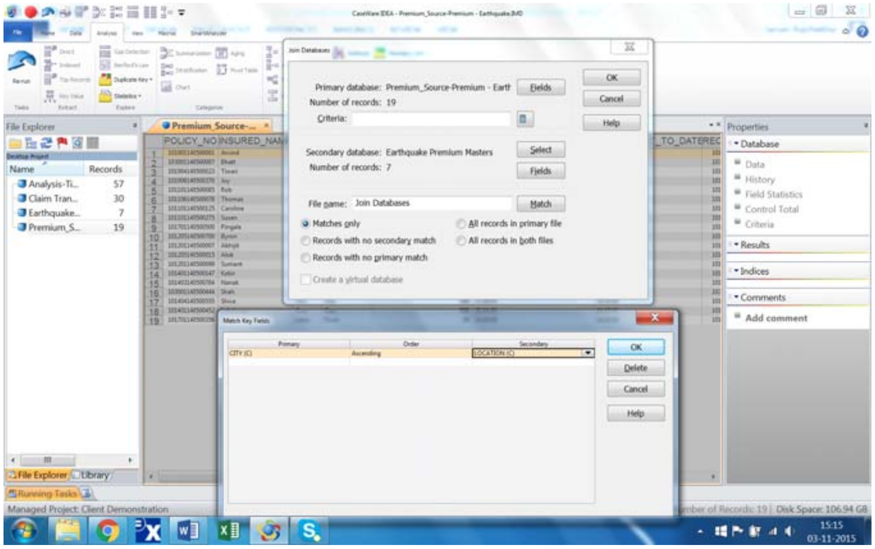
The 'City' field is selected at the entry stage, based on the documentary evidence submitted and site visits.

The 'Location' field on the other hand has no referential link to the 'City' field, and can be independently entered.

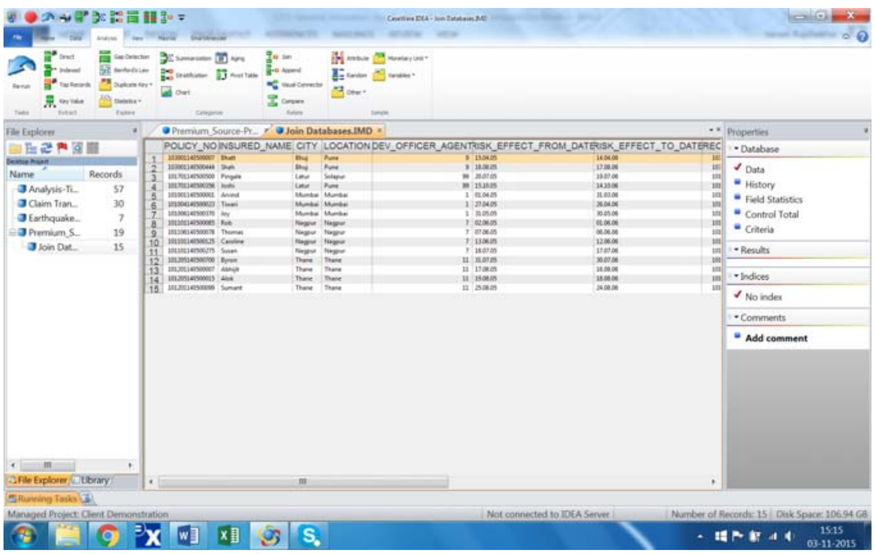
The Premium Master Rate applied to the sum insured is dependent on the 'Location' and not the 'City'.

To consider the possibility of an incorrect selection of the 'location' affecting the Master Premium Rate applied by the system, we match the two databases based on the 'City' field in the Primary Database and 'Location' field in the Secondary Database.

Click Matches Only.



The Joined database appears as a Child database of the Parent - Primary database - Premium_Source – Premium – Earthquake. IMD



APPEND A CALCULATED FIELD – VIRTUAL NUMERIC FIELDS

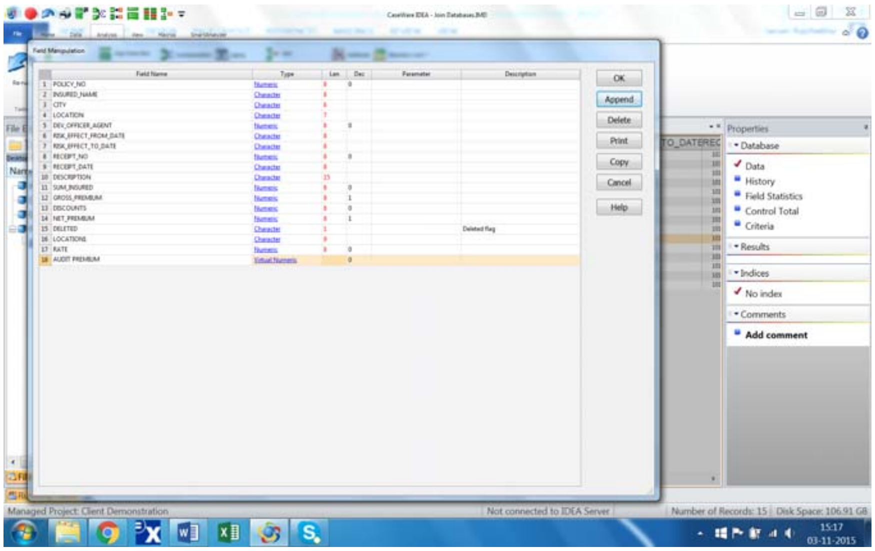
Double click on any field cell in the joined database

This invokes the Field Manipulation dialog box.

Append a field with the following specifics:

Name:Write ‘Audit Premium’

Type: Select ‘Virtual Numeric’ from the drop down list of Field Types.



Click the Parameter cell.

USE THE EQUATION EDITOR

This opens the Equation Editor.

Logical criteria in IDEA are written through the Equation Editor.

In this specific audit test, apply the master premium rate to the sum insured.

The result is to be deducted from the premium recovered to identify errors in premium recovery.

The logical criteria / equation can be entered in two ways.

The first one is used if we don’t know the exact field name, but are aware of the starting letter of the field.

Type the starting letter of the field to be entered and then select the field from the List of fields displayed on the screen that begin with that letter.

Type ‘s’. IDEA displays ‘sum_insured’. Press enter.

The second way is to click ‘Insert database field’.

This option gives a list of all the fields in the database.

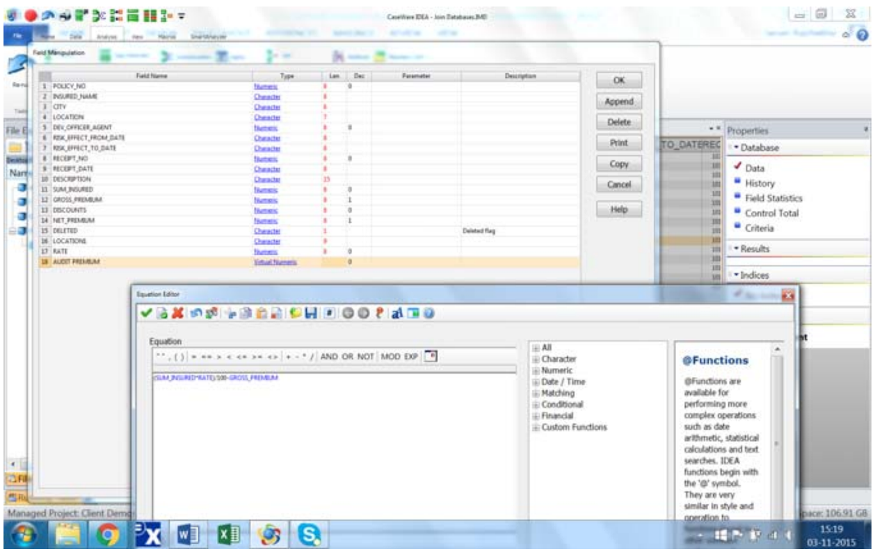
Scroll to the field of choice, and click the field.

In this way enter the equation (sum_insured*rate)/100-Gross_Premium.

Constants and logical operators can be entered by clicking on the Equation Editor calculator or using the keyboard.

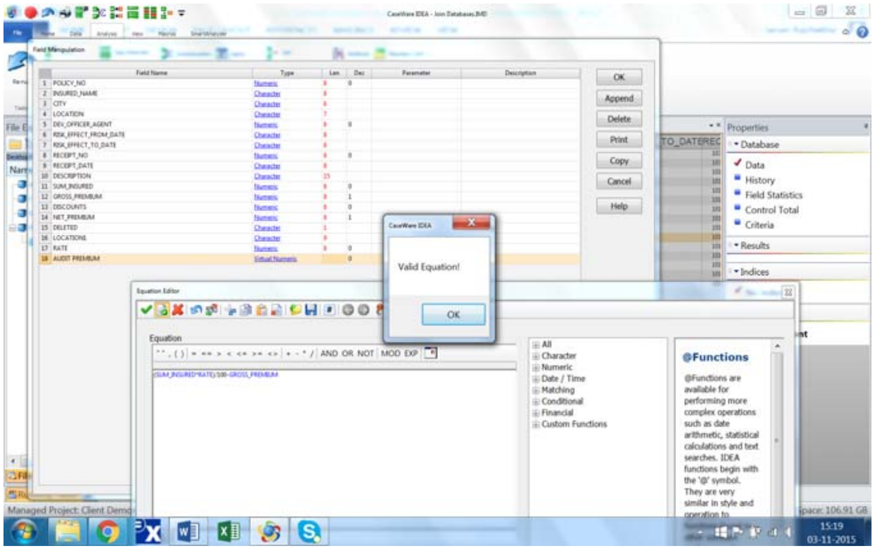
After entering the equation in the equation box, click the Validate Equation button.

This is done to check the syntax, logic and structure of the equation.



The equation is checked by IDEA.

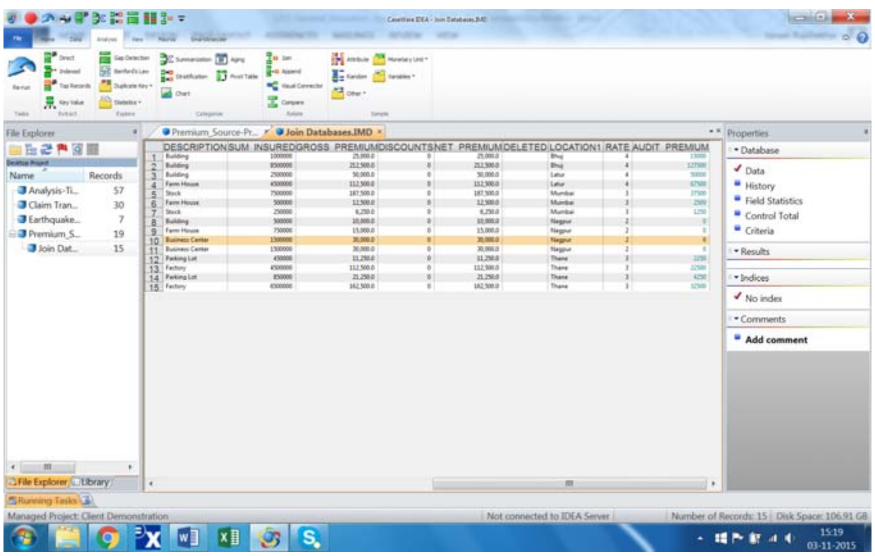
The equation is valid. Click OK in Valid Equation box.



Click on the Validate and Exit button to process the equation on the open Database.

Click OK in the Field Manipulation dialog box.

The resultant computed virtual numeric field appears in a GREEN FONT



EXTRACT DATA WITH A SPECIFIED CRITERIA

Now perform a data extraction so as to identify all the differences in the premium charged vis-à-vis audit premium from the computed field.

Click Analysis – Extract - Direct

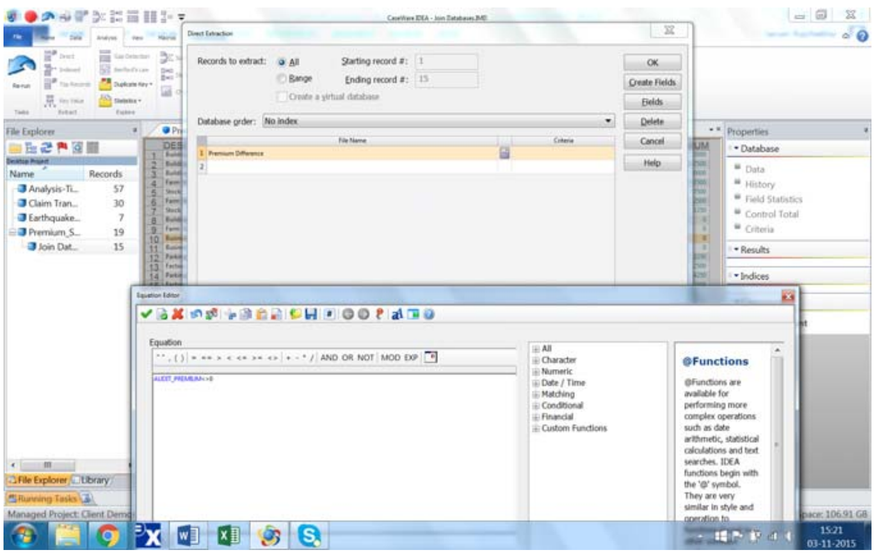
This invokes the Direct Extraction dialog box

Enter the File Name as ‘Premium Difference’ and click on the calculator button.

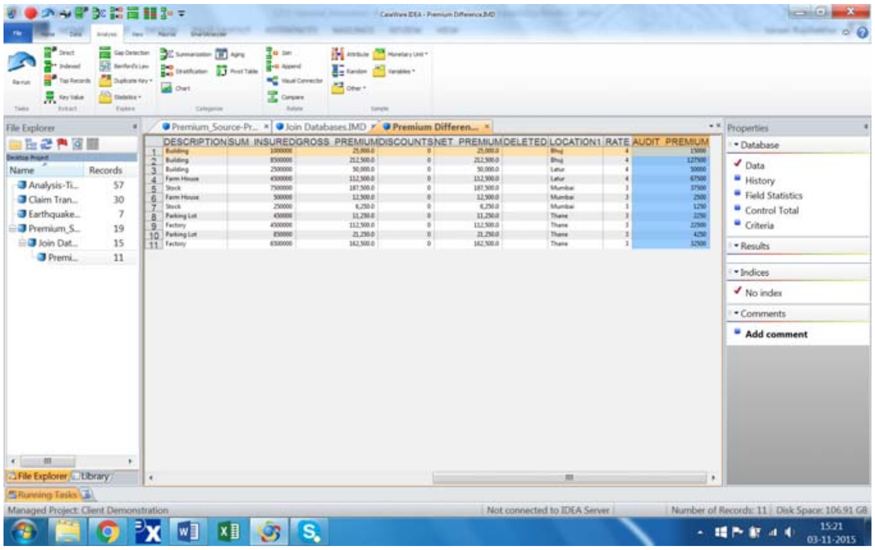
This opens the Equation Editor.

Enter the equation ‘audit_premium<>0’ in the Equation Box.

This equation will identify separately differences in premium charged vis-à-vis the computed field.



The extraction gives 11 cases of short premium charged, as seen from the field “Audit_Premium”.



Click the Set Control total button.

Set the Control Total for the computed field Audit_Premium

The Control Total appears as 362750.

This is indicative of the Net Premium Under-Recovered during the period of review.

CAUSE FOR THE UNDER-RECOVERY

A closer review reveals the following:

In Policy Number 103001140500007 and 103001140500444 the city being BHUJ, location was wrongly selected as PUNE while preparing the policy. Hence the system picked the premium rate for PUNE, which is 2.5, whereas 4 should have been applied for BHUJ.

In Policy Number 101701140500500 and 101701140500356 the city being LATUR, location was wrongly entered as SHOLAPUR and PUNE respectively. Hence incorrect rates were applied by the system based on incorrect location selected by the development officer.

LEARNING

The development officer while selecting the Location field in the policy generation process has inadvertently selected the wrong location from the index in the insurance application software.

As premium applied is based on the location field, and the location does not match with the city in certain cases, incorrect (short) premium has been applied.

As a system suggestion the Location field should not be a stand-alone field. The field should be linked to the city field in the application system.

2.0 RE-VERIFICATION OF PREMIUM COLLECTED FOR INSURANCE AGAINST MOTOR CLASS OF RISKS

OBJECTIVE

To verify whether the correct value of Premium on Motor Class of risks has been recovered from the insured as per the Premium Masters and the correct City Code represented by the Vehicle Registration Number.

EXERCISE DESCRIPTION

In this exercise we will join the Premium Transaction and Premium Master File based on the Zones respectively.

Append a computed field - virtual numeric field to re-calculate premium on sum insured by applying the master rate for the Zone to the Sum Insured.

Arrive at possible differences between premiums recovered and computed premium

Understand the reasons for differences if any in premium recovered.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Use @ Functions
- Join two files.
- Append computed fields – virtual character fields.
- Use the Equation Editor.
- Extract data with a specified criterion.

Data Import

Import the files Premium_Source – Premium – Motor. xls and Motor Premium Master. Dbf into IDEA as per the steps demonstrated in the session on IMPORT OF DATA FILES.

USE @ FUNCTIONS

The location in the Policy Creation stage is ordinarily guided by the first 5 digits or characters or location codes of the Vehicle registration number.

This represents the city of registration and use, with the assumption that the vehicle is insured in the same city in which it has been registered and used.

Since the master file also contains the location code, the location code should be captured as a separate field in the premium file for further comparison.

The location code should be captured as a separate field from the Vehicle Registration field by writing an @ function - @ left in the Equation Editor.

The @ Left function isolates the left most occurrences of a fixed number of digits; in our case the first 5 digits of the Vehicle Registration number.

Go to Field Manipulation in Data in the Menu Tool-Bar.

Append a field with the following specifics:

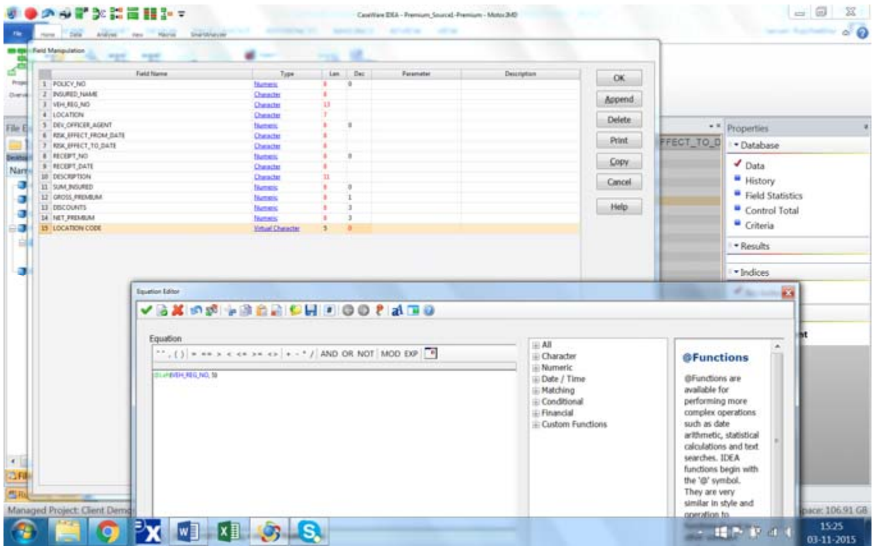
Name: Write ‘Location Code’

Type: Select ‘Virtual Character’ from the drop down list of Field Types.

Length: 5

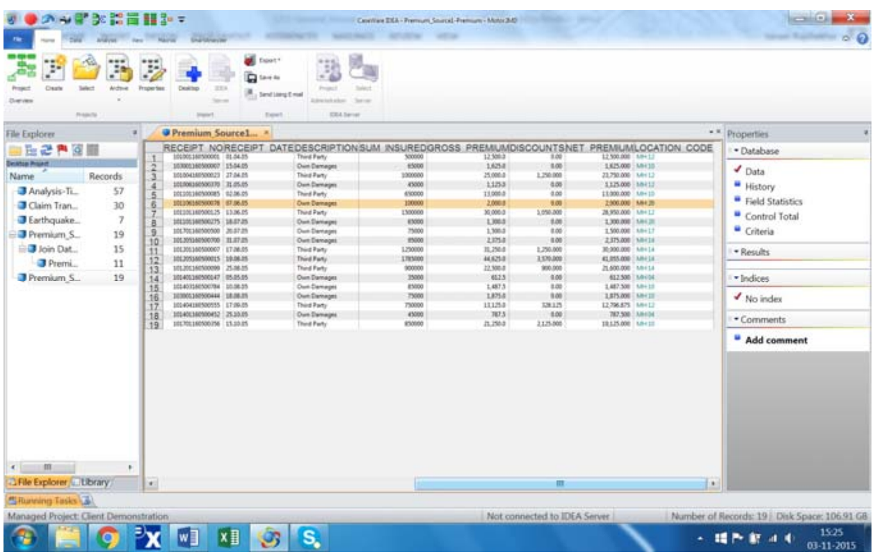
Click the Parameter cell.

Enter the equation ‘@left(Veh_Reg_No, 5) in the Equation Box of the Equation Editor.



The virtual character field 'Location Code' appears in GREEN FONT.

Notice the contents of this field. It represents the location code from the vehicle registration number.

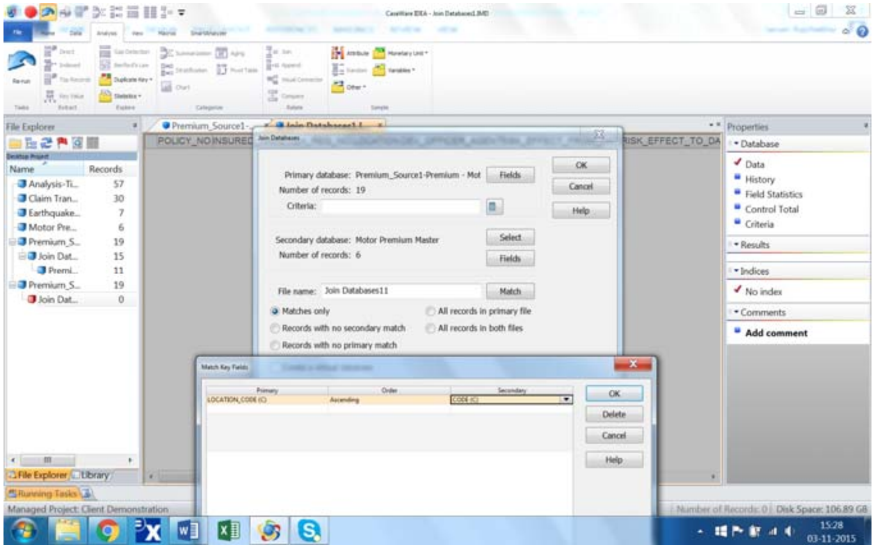


JOIN TWO FILES

Designate Premium_Source – Premium – Motor. IMD as the Primary Database and Motor Premium Masters. IMD as the Secondary Database.

Join the two databases on the basis of 'Location Code' from the Primary

Database and 'Code' from the Secondary Database.



Tip – For a file join the matching fields do not need to have the same field name. Only the field type should be the same for the match to be effective.

USE THE EQUATION EDITOR

Append a field through Field Manipulation with the following specifics:

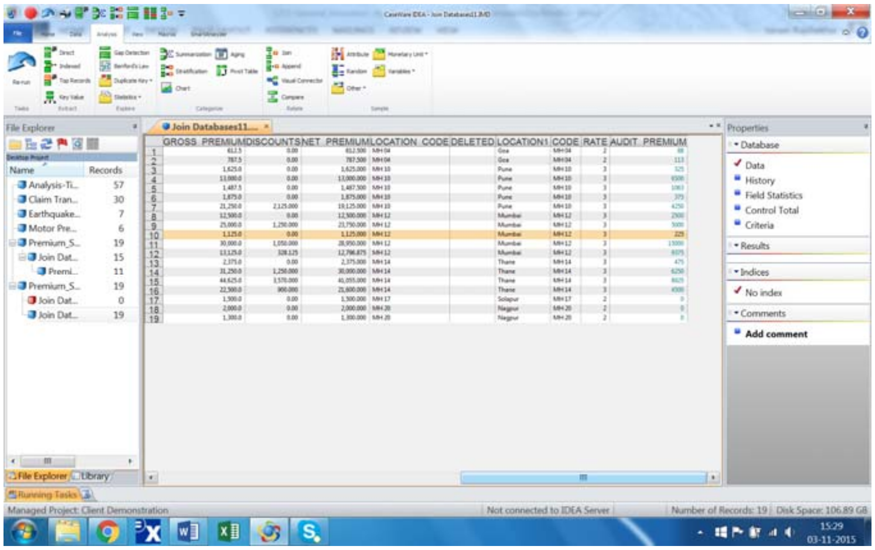
Field Name: Audit Premium

Type:Virtual Numeric

Enter an equation in the Equation Box of the Equation Editor as

(sum_insured*rate)/100-gross_premium

The resultant computed virtual numeric field appears as –

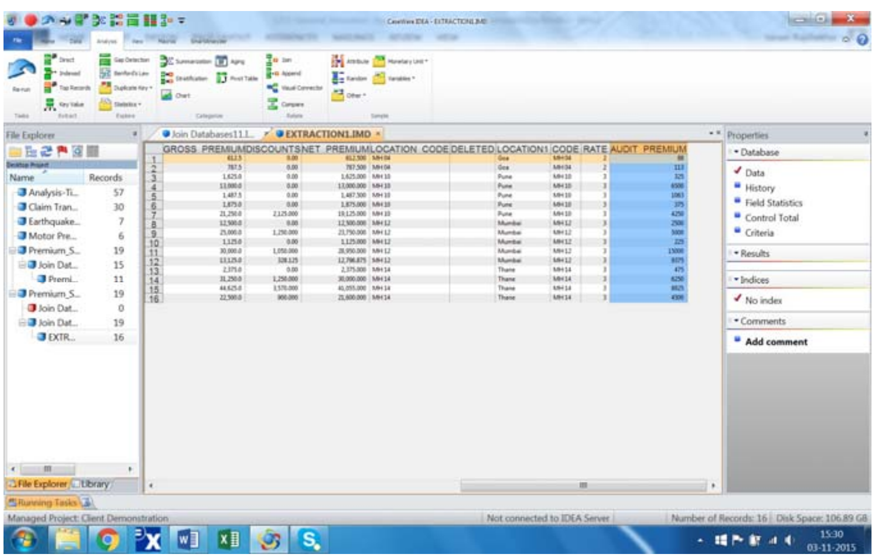


EXTRACT DATA WITH A SPECIFIED CRITERIA

Now extract data through Analysis – Extract - Direct with the help of the equation

audit_premium<>0

The Extraction gives 16 cases where motor premium has been under – recovered. The Control total of such cases is 64964.



CAUSE FOR THE UNDER-RECOVERY

A closer review reveals the following:

In Policy No 101101160500085 and 101403160500784 the vehicle has been registered and is being used in Pune, but the officer has wrongly selected the Location as Nagpur and Goa respectively. Hence 2% and 1.75% have been applied by the system for Nagpur and Goa respectively in place of 2.5% for Pune.

CONCLUSION

The development officer while selecting the Location field in the policy generation process has inadvertently selected the wrong location from the index in the insurance application software.

As premium is applied based on the location field, and the location does not match with the Location code in certain cases, it means that incorrect (short) premium has been applied.

As a system suggestion, the Location field should not be a stand-alone field. The field should be linked to the Location Code in the application system.

3.0 RE-VERIFICATION OF DISCRETIONARY DISCOUNT ALLOWED ON PREMIUMS IN THE NON-TARIFF FREE MARKET INSURANCE SEGMENT

OBJECTIVE

To verify whether the discretionary discounts have been correctly allowed as per the Discount Masters

EXERCISE DESCRIPTION

In this exercise we will join the Premium Transaction and Discount Master File based on the Development Officer Code.

Append a computed field, -virtual numeric field to re-calculate discount on gross premium by applying the master rate for the Development Officer to the Gross Premium.

Arrive at the difference between discount allowed and computed discount

Understand the reasons for difference, if any, in the discount allowed.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Join two files.

- Append computed fields, virtual numeric fields.
- Use the Equation Editor.
- Extract data with a specified criterion.

Data Import

Import the file Premium Source-Premium Motor Non Tariff. xls and Discount Master. dbf into IDEA as per the steps in IMPORT OF DATA FILES.

EXTRACT DATA WITH A SPECIFIED CRITERION

Perform a Direct Extraction from Data in the Menu Tool-Bar.

Isolate only those premium receipts where discount has been allowed.

Do the same by writing an equation – discounts<>0

The resultant database will be a sub-database by the name 'Discount Allowed'.

This sub-database will be used to test the correctness of discount allowed.

JOIN TWO FILES

Designate Discount Allowed. IMD as the Primary Database and Discount Masters. IMD as the Secondary Database.

Join the two databases on the basis of 'Development Officer Code' available in both the databases.

APPEND COMPUTED FIELDS – VIRTUAL NUMERIC FIELDS

To the joined file, append a field through Field Manipulation with the following specifics:

Field Name: Audit Discount

Type: Virtual Numeric

Enter an equation in the Equation Box of the Equation Editor as

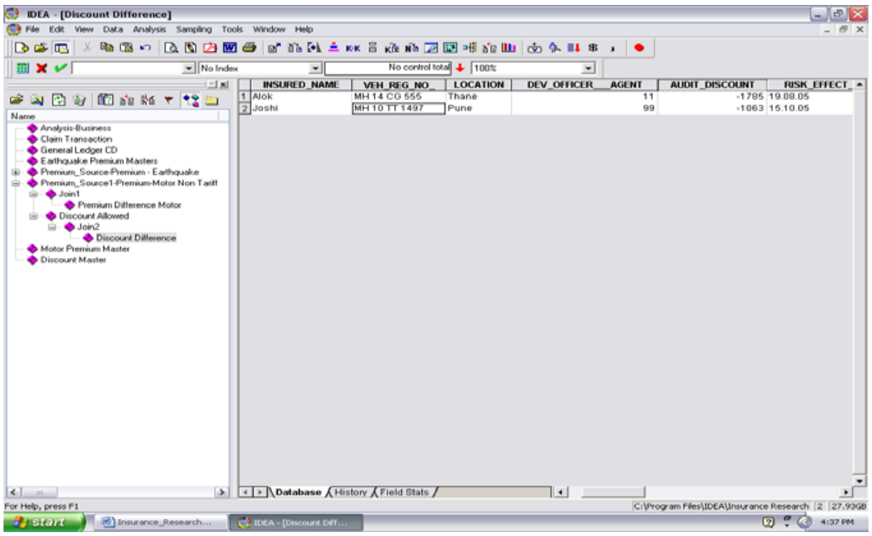
(gross_premium*discount_r)/100-Discounts

The computed numeric field appears in PINK FONT.

Extract the differences to a separate file through Data Extraction. Write an equation

Audit_Discount<>0

The resultant difference in discount computed vis-à-vis discount allowed is reflected in a separate child file.



CAUSE FOR THE EXCESS DISCOUNT ALLOWED

A closer review reveals the following:

For Vehicle registration number MH 14 CG 555, Insured Alok, and Agent code 11, the discount allowed is 8% whereas agent 11 has the authority to give discount only up to 4% of the Gross Premium.

The Division Office has given a special approval for Policy No. 101205160500015, Insured Alok for a higher discount rate from the norm, for which the necessary approval documents are available on record prior to policy release.

CONCLUSION

The agent while exercising his discretionary right for discount has gone beyond the limit in two cases.

The system has inbuilt controls to control the discount rate as per the masters, and any exceptions over the masters have to be electronically approved by the DO, which was actually done in the first case.

It appears that the system has been bypassed in the second case without a preventive message. This matter needs to be taken up with the in-house IT Team for their comments and action.

4.0 VERIFICATION OF RATING FIELDS FOR SPECIAL ADDITIONAL IMPOSES / PENALTY IN CASE OF FREQUENT CLAIM HISTORY ON SENSITIVE POLICIES

OBJECTIVE

To verify whether special additional impose / penalty has been correctly recovered in the rating fields, as per the instructions of the Divisional Office for policies experiencing frequent past claims.

EXERCISE DESCRIPTION

In this exercise we will join the Premium Transaction and Rolling Policy File based on the Old Policy Number.

Identify policies where special additional imposes have not been charged

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Join two files.
- Extract data with a specified criterion.

BACKGROUND

The management audit team at the Divisional Office has identified the following Policies in the earlier year having excessive claim history: 101106160400045, 103001160400333, 101004160400478, & 101701160400174.

This inference has been made from the Incurred Loss Statement for Motor Claims for 2004.

Due to repetitive claim history the Divisional Office has ordered an Additional Excess Impose of Rs. 500 to be charged as penalty for every renewal in the current year.

Based on the policy numbers forwarded by the Divisional Office to the Branch office, the management audit team at the branch has used the conversion facility in the insurance application software to arrive at the corresponding current policy numbers.

This is seen in the file ‘Premium Source 2 – Rolling Policies.IMD’ as imported into IDEA.

JOIN FILES

Designate Premium Source1-Premium-Motor Non Tariff. IMD as the Primary Database and Premium Source 2- Rolling Policies. IMD as the Secondary Database.

Join the two databases on the basis of 'Policy No' available in both the databases.

The Policy No is the current file to be matched with the Policy No 2005 in the Rolling file.

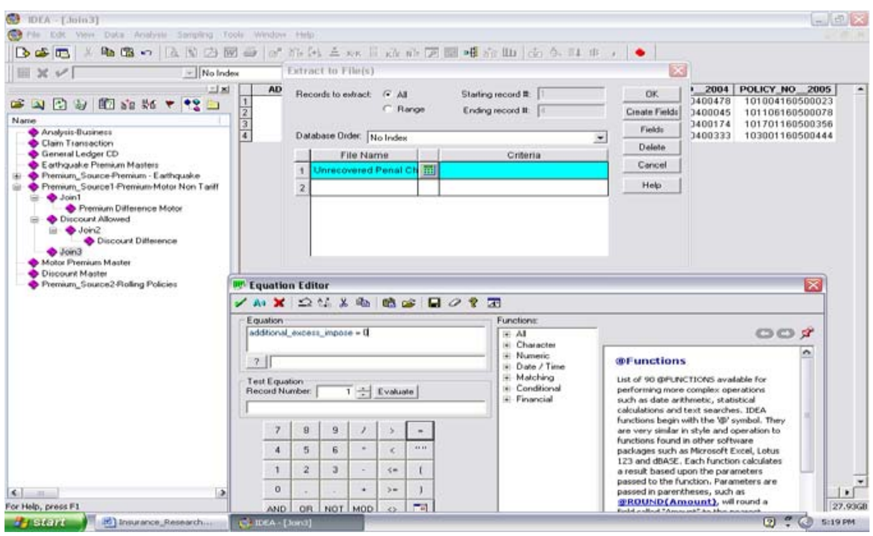
EXTRACT DATA WITH A SPECIFIED CRITERIA

The resultant joined file represents policies prior to 2005, which have experienced repetitive claims and have been subsequently renewed with clear instructions to levy penal charges.

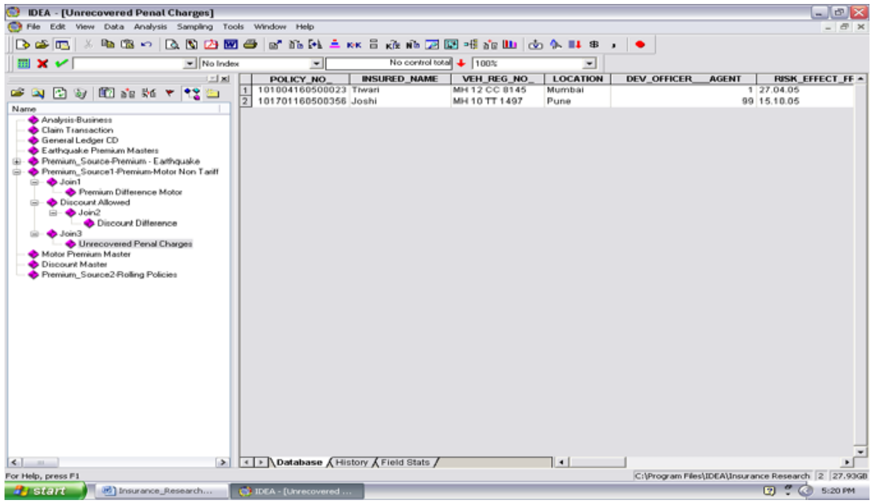
As a corollary, all the policies finding place in this file should have a special additional impose / penalty of 500 added in the rating field.

To test this file perform a direct data extraction through the equation

Additional_Excess_Impose=0.



The extraction gives a sub-file with 2 record exceptions.



CAUSE FOR THE NON-RECOVERY

A closer review reveals the following:

The development officer while renewing the existing policy in the system has not referred to the list of policies experiencing repetitive claim history as informed by the Divisional Office to the branch.

CONCLUSION

The procedure of manual comparison of current policies against past policies is flawed, tedious and subject to error of oversight or advertent omission.

List of policies experiencing frequent claims in the past should be mastered within the insurance application system.

Whenever an agent renews an existing policy that comes within the frequent claim history category, the system should automatically prompt the agent to levy the impose, without which the new policy cannot be created in the system.

The discretion to waive the impose should vest with the Divisional office through electronic validation.

5.0 INPUT FIELD CHECKS ON THE(?) NON-SYSTEM GENERATED OR MANUAL POLICIES

OBJECTIVE
To verify whether there is any missing, duplicate, fabricated policy number in the list of non-system or manual insurance policies.

EXERCISE DESCRIPTION

In this exercise we will run tests for duplicate detection on the Policy Number field.

Specific tests for number format verification will be undertaken on the Policy Number field to establish conformity with the standard numbering pattern.

Gap tests will be run to identify missing numbers.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Identify Duplicates
- Perform Number format Tests
- Display Function
- Identify Gaps
- Extract data with a specified criterion.
- Use @ Functions

Data Import

Import the file Non system policies. prn into IDEA as per the steps in IMPORT OF DATA FILES.

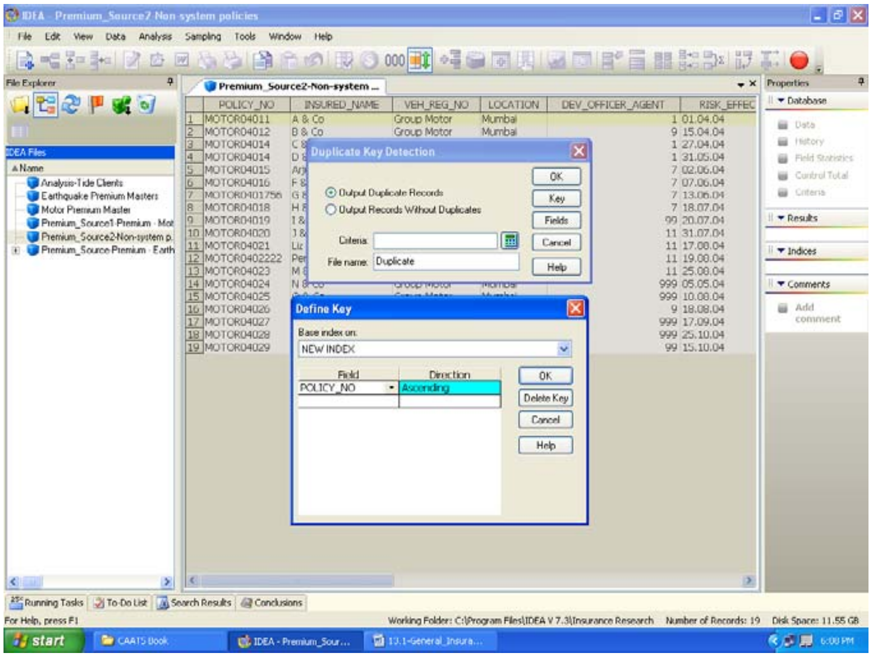
IDENTIFY DUPLICATES

Select Duplicate Key Detection from Analysis on the Menu tool bar.

This opens the Duplicate Key Detection dialog box.

Click Key. This opens the Define Key dialog box. Let the Field be defined as POLICY_NO.

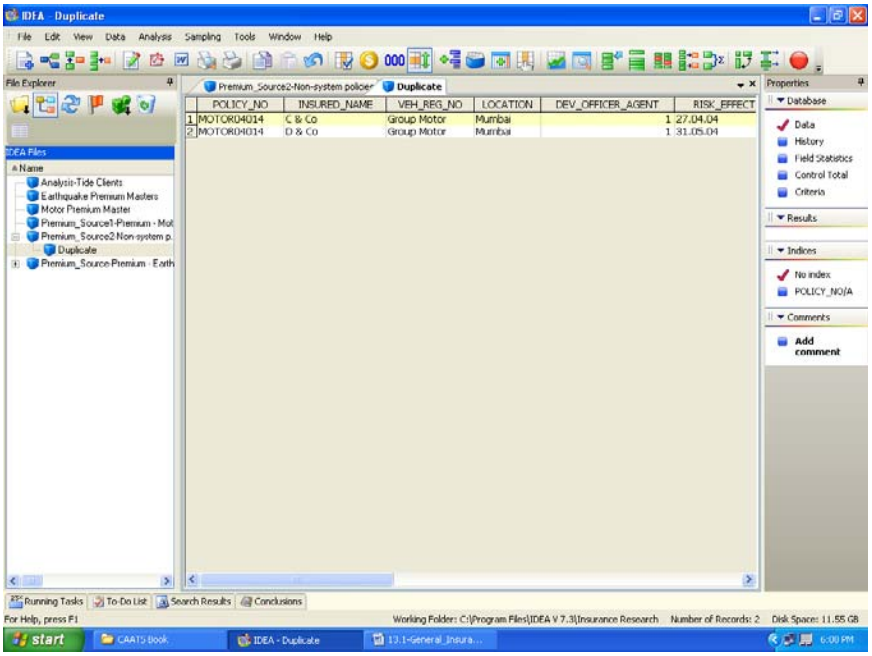
Click OK. Check Output Duplicate Records in the Duplicate Key Detection dialog box. Click OK.



The test gives us two instances of policies having the same number MOTOR04014.

These cases need to be taken up for further investigation with the development officer and / or agent.

The Insured would need to be contacted in secret to establish his or her identity / validity.



PERFORM NUMBER FORMAT TESTS

Each manual non-system generated policy is in a standard format of 10 digits and has an alpha numeric field type.

The purpose of this test is to identify policies that do not match the standard number format.

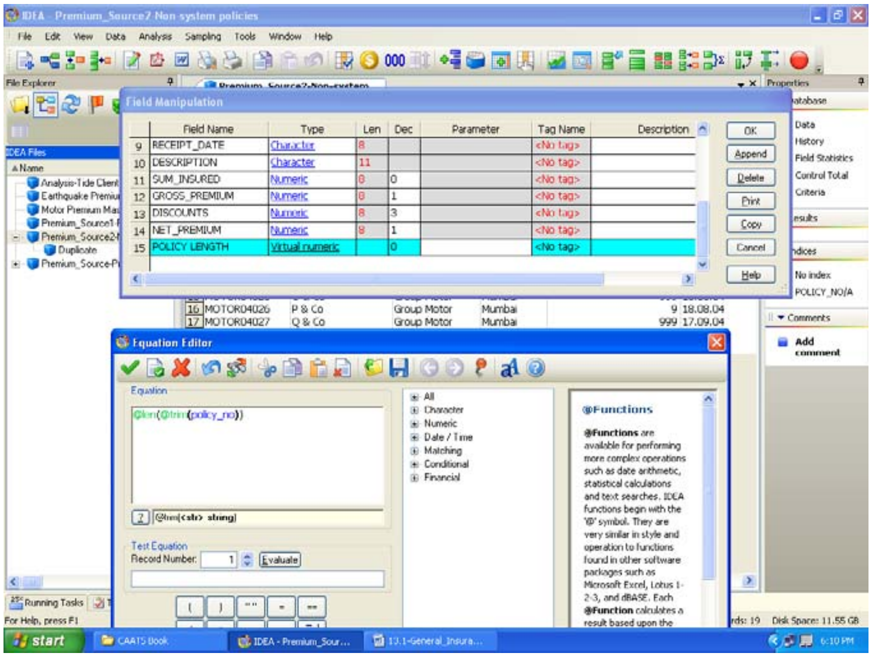
In the first stage we write an @ Function to give us the actual length of each policy.

Append a field through Field Manipulation with the following specifics:

Field Name: Policy Length

Type:Virtual Numeric

Write an equation @len(@trim(policy_no)) in the Equation Editor.



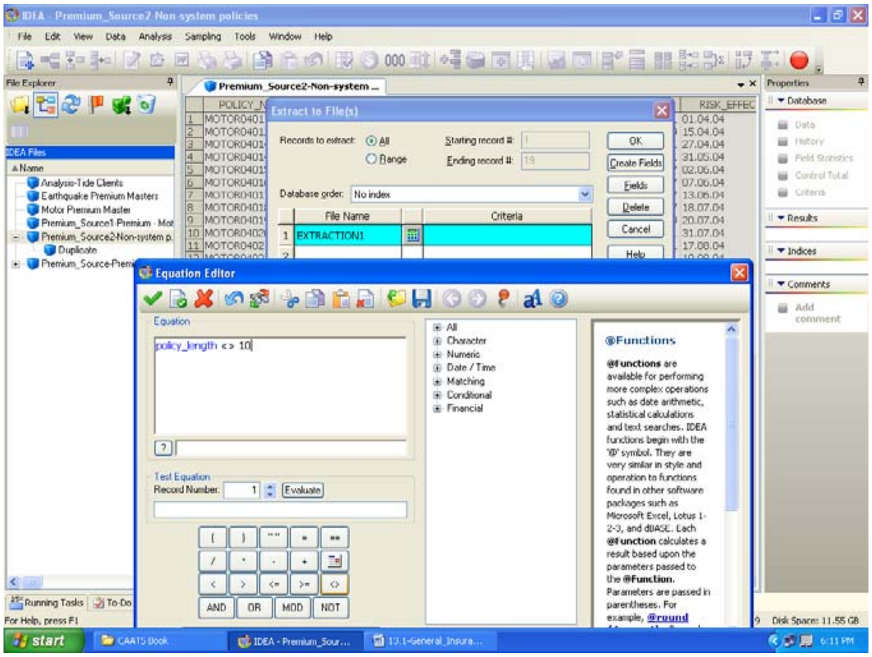
Tip – If @ Len is to be used on a character field as in the case above then always use the function in conjunction with @trim so as to trim the field of spaces before calculating the length. Blank spaces are also interpreted as digits for the purpose of length, and hence the need to trim the field prior to applying the length function.

Now perform a direct data extraction on the computed virtual numeric field.

Write a logical expression

Policy_Length<>10

This expression will extract all records where the policy number does not match the standard numbering convention of 10 digits.

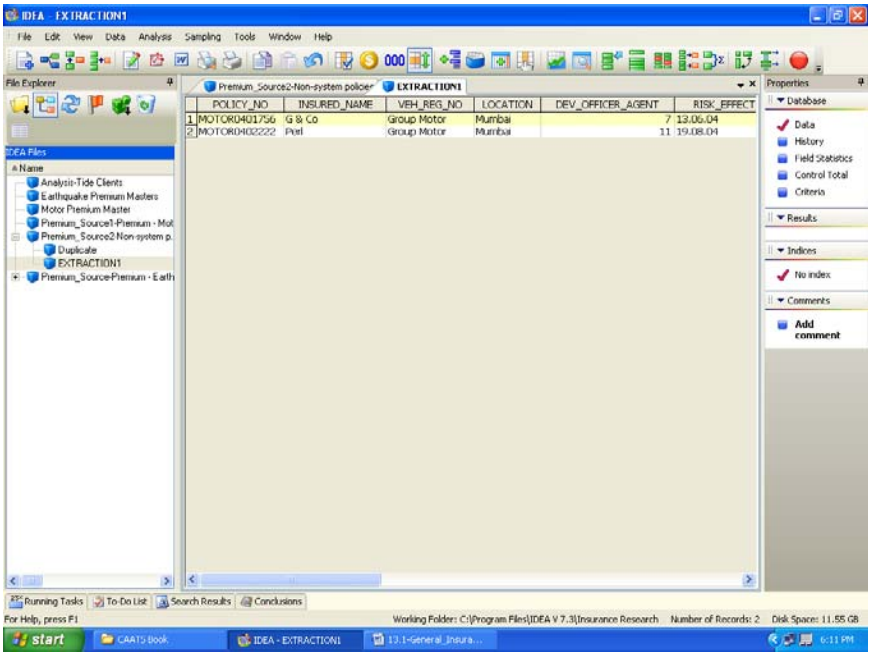


The extraction gives two cases where the numbering convention has not been met.

In the case of G & Co and Perl the policies do not follow the standard convention of 10 digits and actually show 12 digits.

These cases need to be taken up for further investigation with the development officer and / or agent.

The insured would need to be contacted in secret to establish his or her identity / validity.



IDENTIFY GAPS

Let us now identify missing policies.

Tip – The Gap test has to be run on the file containing only those policies that meet the 10 digit numbering convention so that only real gaps are identified, and not gaps on account of differences between the 10 digit and 12 digit policies.

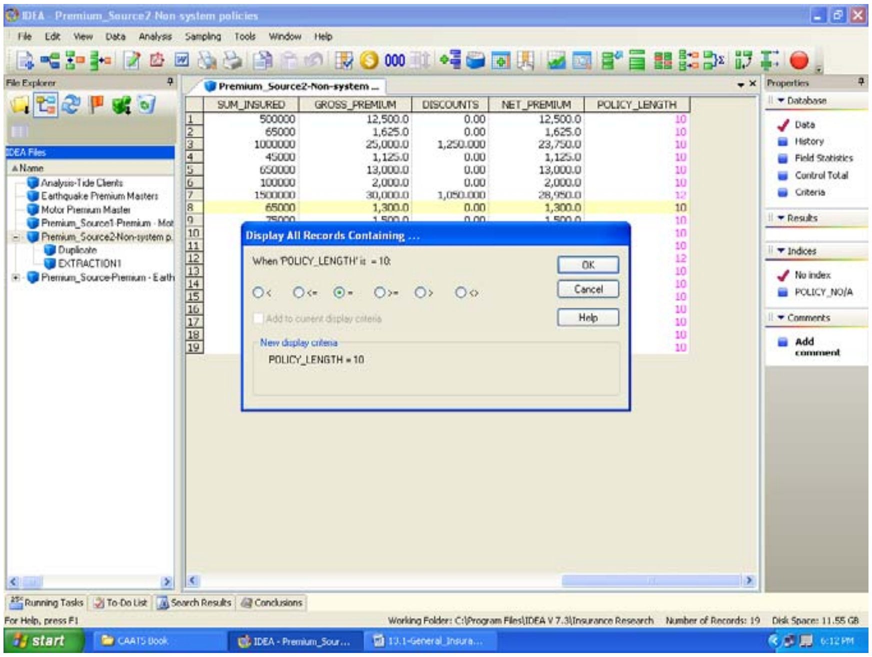
Extract a file containing only the 10 digit policy numbers.

Scroll to the virtual field ‘Policy_Length’ as created in the last exercise.

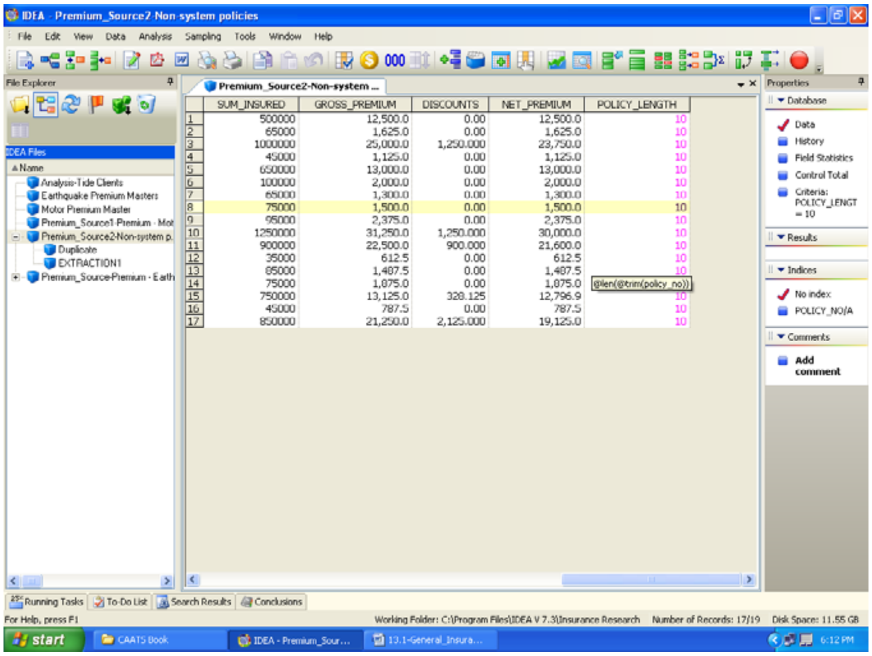
Locate the first record containing 10.

Place your cursor over the cell, right click, and scroll to the option “Display all records containing “10”.

Click OK.



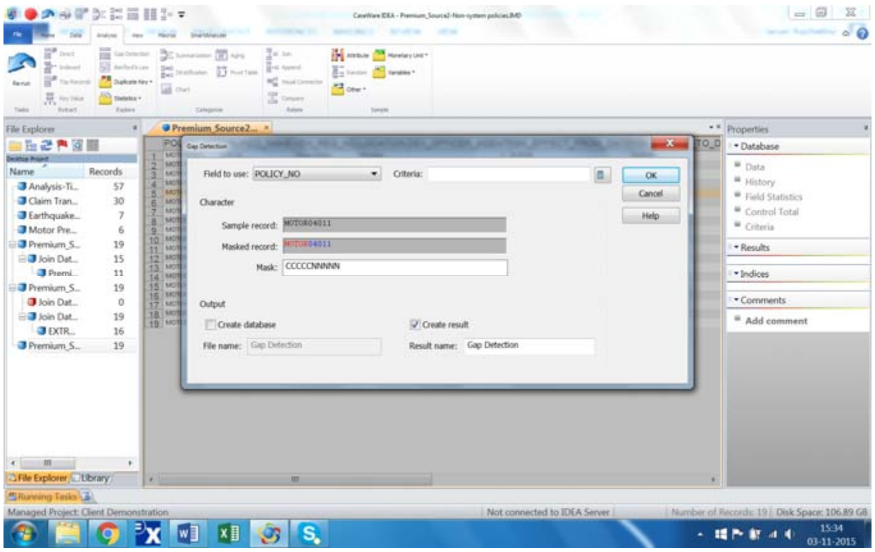
The view generated contains all the policies having 10 digits only.



Proceed to run a gap test on this view to locate the missing policy numbers.

As the Policy No is an alpha-numeric field type, navigate to Character Gap Detection under Data in the menu tool bar.

The MASK is automatically generated by IDEA in the Character Gap Detection dialog box.

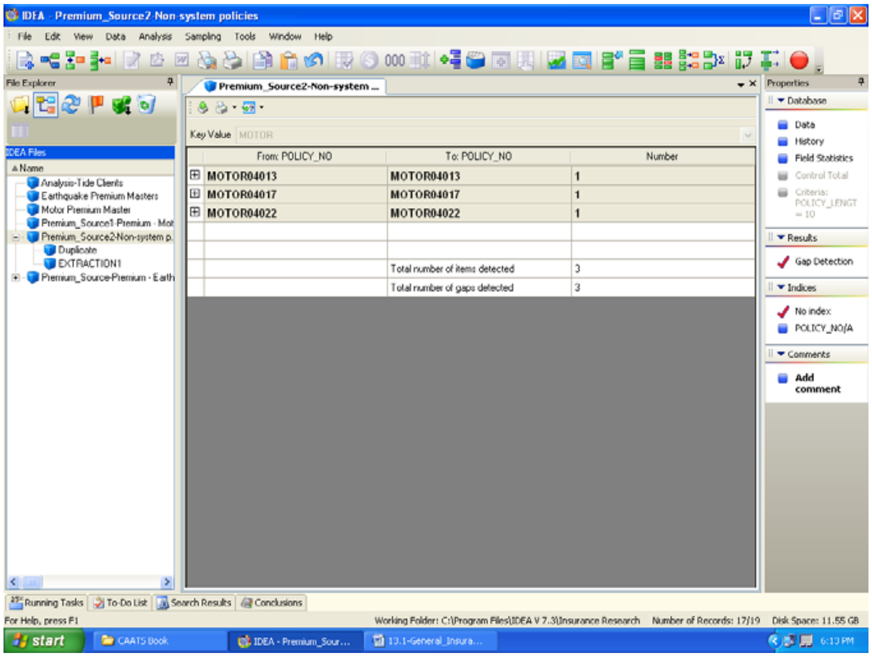


Click OK.

After running the gap test, the result appears as a TAB along with Database, History and Field Statistics.

The test reveals three gaps / missing policy numbers from the list of manual policies.

The cases need to be further investigated as to whether the policies were really not created or created but not entered into the system by the agent.



6.0 IDENTIFYING POLICIES CANCELED SOON AFTER INCEPTION ALONG WITH JUSTIFICATION BY ENDORSEMENTS

OBJECTIVE

To verify whether any policies were canceled soon after risk inception, and if so, whether they are supported by endorsement notes.

EXERCISE DESCRIPTION

In this exercise we will write an @ Function to arrive at the time lag between policy cancellation and policy issue.

Extract policies canceled shortly after issue for close scrutiny.

Extract policies canceled, not backed by endorsement notes.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Append a Field
- Extract data with a specified criterion.
- Use @ Functions

Data Import

Import the file Premium Source 3-Canceled Policies. xls into IDEA as per the steps in IMPORT OF DATA FILES. (check if this change is appropriate?)

BACKGROUND

The insurance company management audit guidelines have a norm that policies canceled within 7 days of risk commencement should be identified and taken up for close scrutiny along with the endorsement records and documentary proof.

APPEND A FIELD

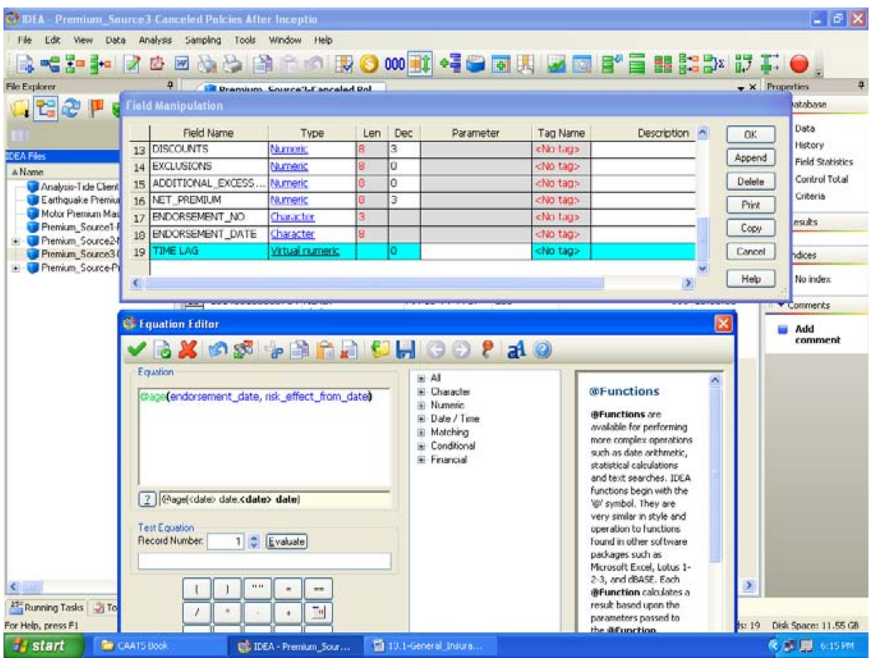
Append a field through Field Manipulation with the following specifics:

Field Name:– Time Lag

Type: Virtual Numeric

Write an @ Function in the Equation Editor.

@age gives the difference between two dates. In this specific test we arrive at the difference between the Risk Effect from Date and the Endorsement Date (Canceled Date)



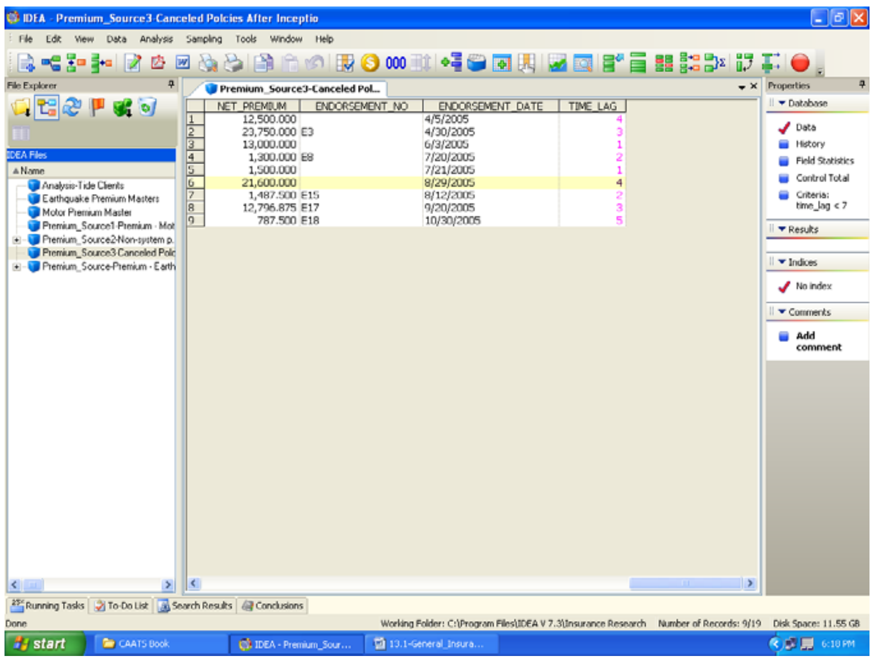
EXTRACT DATA

Based on the virtual numeric field, Time Lag isolates all the policies canceled within 7 days of risk commencement.

Perform a data extraction using the equation

Time Lag < 7

The extraction gives 9 cases of policies canceled within 7 days of risk commencement.

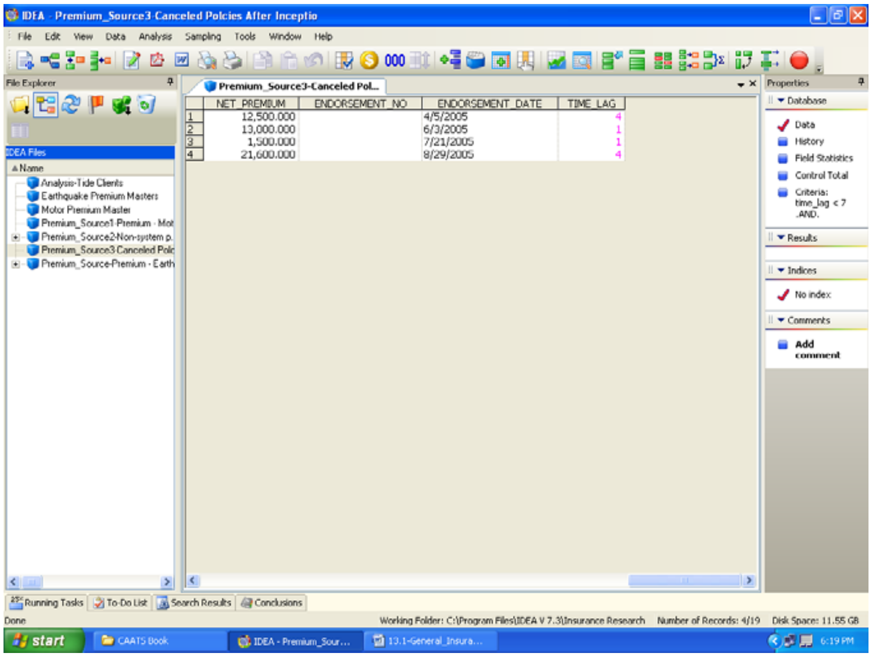


As per the Management Audit guidelines, all policies canceled within 7 days of risk commencement, should be supported by an Endorsement note reference.

Perform a data extraction using the equation

Endorsement_No=" "

The extraction gives 4 cancellations without Endorsement references.



CONCLUSION

It seems that the insurance application system does not compulsorily insist on electronic validation of canceled policies through endorsement note references. This is a major shortcoming of the software.

The development officer should not be able to proceed with policy cancellation unless the event is authorized by an endorsement reference with valid reasons.

Moreover. policy cancellation should not be the prerogative of just one officer. At least, two officers should handle this sensitive high exposure area.

All canceled policies should get reflected in the exception statement, which should be duly checked and authorized by the Branch Manager on a daily basis, filed and kept on record.

7.0 INPUT FIELD CHECKS ON COVER NOTES (TEMPORARY POLICIES) FOR MOTOR INSURANCE AND CONVERSION TO BONA FIDE POLICIES THROUGH RECEIPTS

OBJECTIVE

To verify if there is any missing, duplicate, fabricated cover note and whether receipts support all cover notes.

EXERCISE DESCRIPTION

In this exercise we will run tests for duplicate detection on the Cover Note field.

Specific tests for number format verification will be undertaken on the Cover Note field to establish conformity with the standard numbering pattern.

Gap tests will be run to identify missing cover note numbers.

Tests to identify cover notes not supported by receipts

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Identify Duplicates
- Perform Number format Tests
- Display Function
- Identify Gaps
- Extract data with a specified criterion.
- Use @ Functions

Data Import

Import the file Premium Source 4 – Cover Note. xls into IDEA as per the steps demonstrated in the session on IMPORT OF DATA FILES.

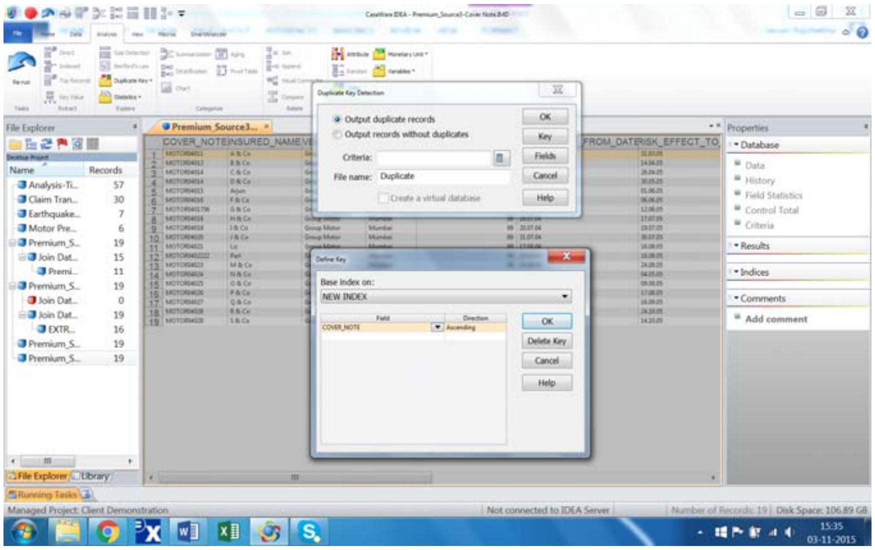
IDENTIFY DUPLICATES

Select Duplicate Key Detection from Analysis on the Menu tool bar.

This opens the Duplicate Key Detection dialog box.

Click Key. This opens the Define Key dialog box. Let the Field be defined as COVER_NOTE.

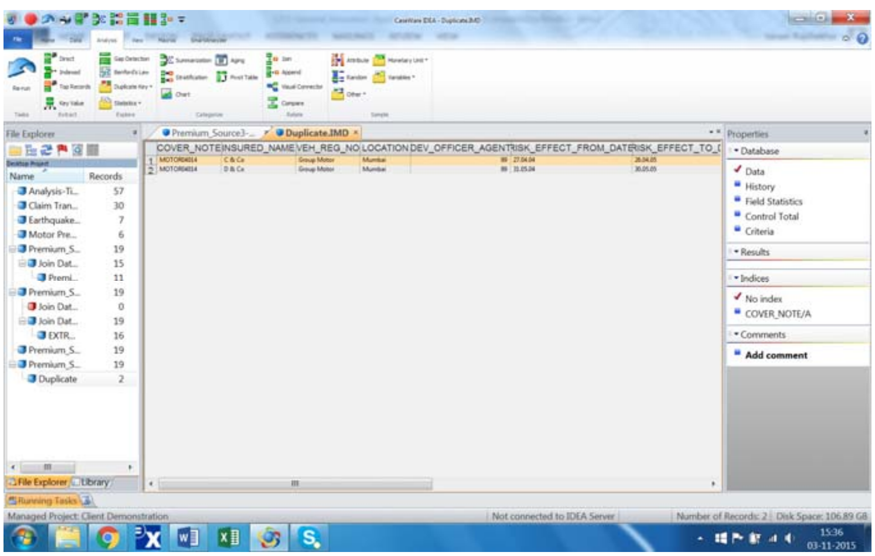
Click OK. Check Output Duplicate Records in the Duplicate Key Detection dialog box. Click OK.



The test gives us two instances of cover notes having the same number MOTOR04014.

These cases need to be taken up for further investigation with the development officer and / or agent.

The Insured would need to be contacted in secret to establish his or her identity / validity.



PERFORM NUMBER FORMAT TESTS

Each cover note is in a standard format of 10 digits and has an alpha numeric field type.

The purpose of this test is to identify cover notes which do not match the standard number format.

In the first stage we write an @ Function to give us the actual length of each cover note.

Append a field through Field Manipulation with the following specifics:

Field Name: Cover Note Length

Type:Virtual Numeric

Write an equation @len(@trim(cover_note)) in the Equation Editor.

> **Tip** – If @ Len is to be used on a character field as in the case above then we always use the function in conjunction with @trim so as to trim the field of spaces before calculating the length. Blank spaces are also interpreted as digits for the purpose of length, and hence the need to trim the field prior to applying the length function.

Now perform a direct data extraction on the computed virtual numeric field.

Write a logical expression

Cover_Note<>10

This expression will extract all records where the cover note does not match the standard numbering convention of 10 digits.

The extraction gives two cases where the numbering convention has not been met.

In the case of G & Co and Perl the cover notes do not follow the standard convention of 10 digits..

These cases need to be taken up for further investigation with the development officer and / or agent.

The insured would need to be contacted in secret to establish his or her identity / validity.

IDENTIFY GAPS

Let us now identify missing cover notes.

> **Tip** – The Gap test has to be run on the file containing only those cover notes that meet the 10 digit numbering convention so that only real gaps are identified, and not gaps on account of differences between the 10 digit and 12 digit cover notes. .

Extract a file containing only the 10 digit Cover Notes.

Scroll to the virtual field 'Cover_Note_length' as created in the last exercise.

Locate the first record containing 10.

Place the cursor over the cell, right click, and scroll to the option "Display all records containing "10".

Click OK.

The view generated contains all the cover notes having 10 digits only.

Proceed to run a gap test on this view to locate missing cover note numbers.

As the Cover Note is an alpha-numeric field type, navigate to Character Gap Detection under Data in the menu tool bar.

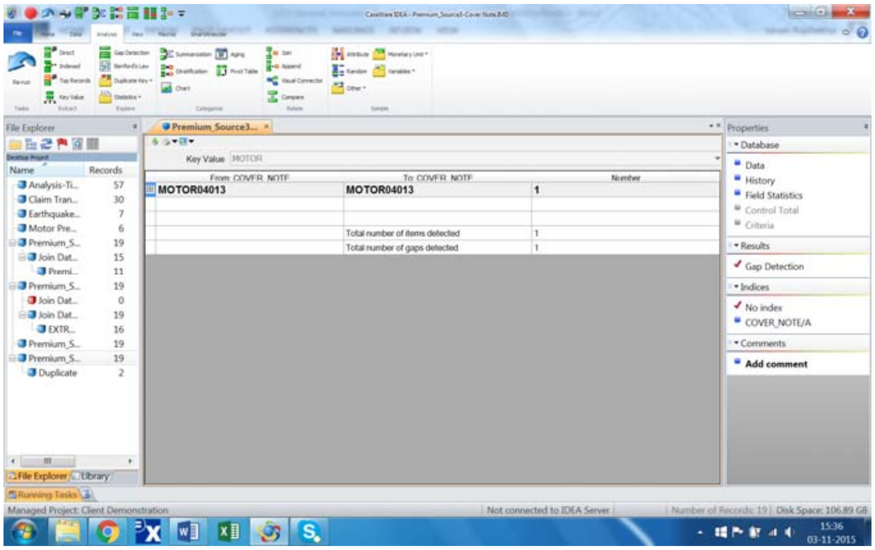
The MASK is automatically generated by IDEA in the Character Gap Detection dialog box.

Click OK.

After running the gap test, the result appears as a TAB along with Database, History and Field Statistics.

The test reveals three gaps / missing cover note numbers.

The cases need to be further investigated to confirm if the cover notes were really created or created but not entered into the system by the agent.



EXTRACT DATA WITH A SPECIFIED CRITERION

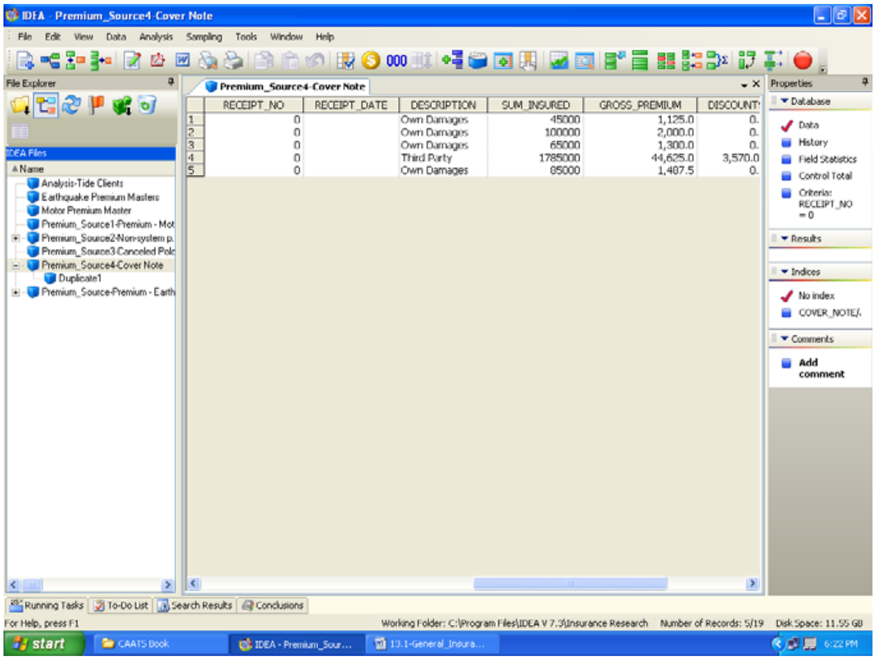
Every cover note should be supported by a receipt generated by the system.

Proceed to identify cover notes not supported by receipts.

Perform a data extraction using the equation

Receipt_No=" "

The extraction reveals 5 cover notes not backed by receipt numbers.



CONCLUSION

It seems that the insurance application system does not compulsorily insist on electronic validation of cover notes while converting them into bona fide policies through the receipt and cover note conversion process. This is a major shortcoming of the software.

The development officer and / or agent should not be able to proceed with policy booking unless the cover note is coupled with the receipt.

13.1.7 Claim Case Studies

1.0 IDENTIFICATION OF DUPLICATE MOTOR CLAIMS IN PARTIAL LOSS POLICIES

OBJECTIVE

To identify duplicate motor claims in partial loss policies.

EXERCISE DESCRIPTION

In this exercise we will run tests for duplicate detection on the policy number and vehicle registration number field.

Prior to running the duplicate detection, we will isolate partial loss claims from the file of total and partial loss claims.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Identify Duplicates
- Extract data with a specified criterion.

Data Import

Import the file Claim_Source - Transaction. xls into IDEA as per the steps in IMPORT OF DATA FILES.

EXTRACT DATA WITH A SPECIFIED CRITERION

A close examination of the file imported will reveal that the claim file is an assortment of total and partial loss claims.

The first step in this test is to isolate the partial loss claims, and convey these claims into a separate file.

Run a direct data extraction on the file imported through the equation

Nol = "Partial Loss"

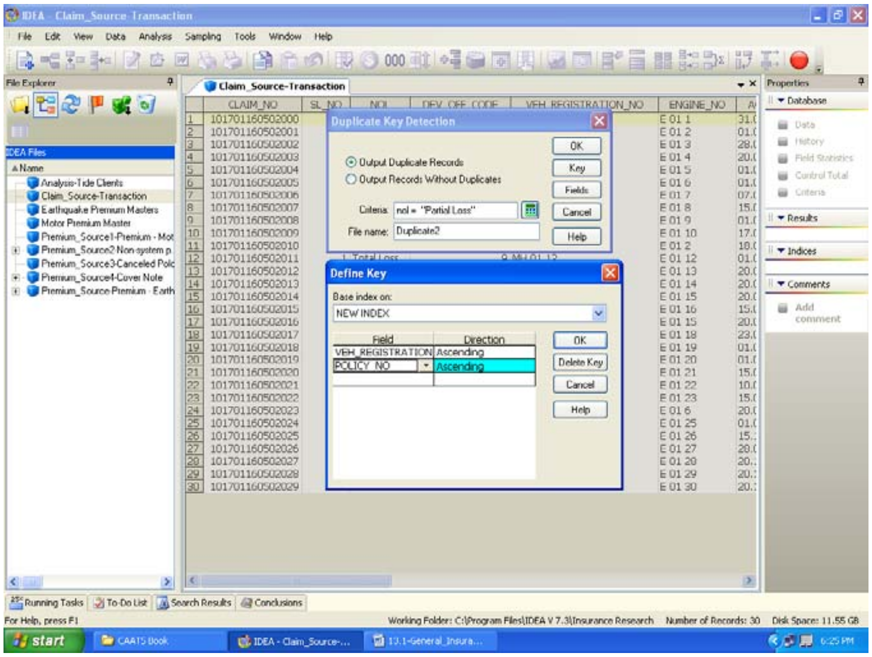
IDENTIFY DUPLICATES

Now proceed with running a duplicate key detection test on the file Partial Loss. IMD generated through the data extraction.

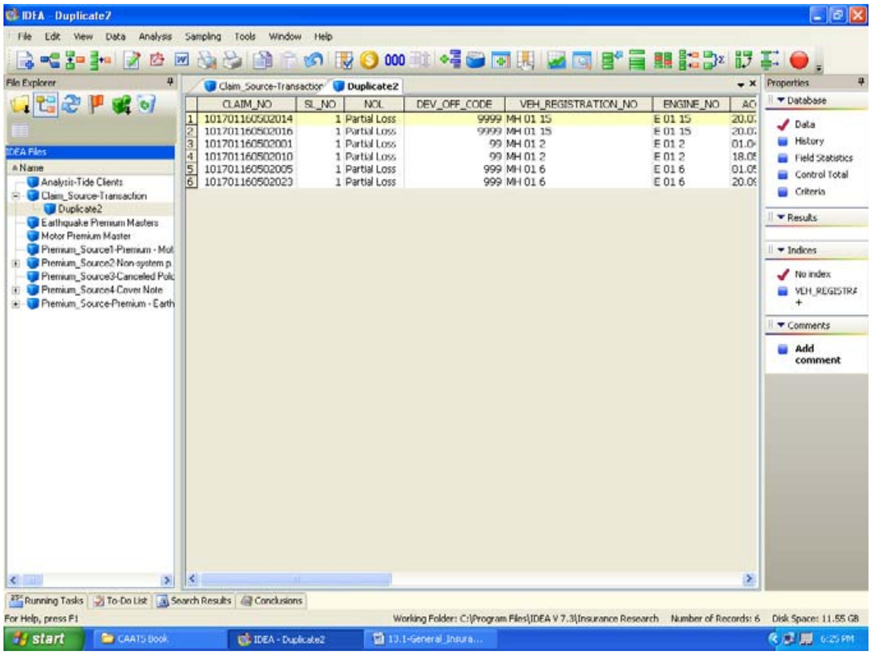
The objective of the test is to identify claims having the same policy number and vehicle registration number.

So select POLICY_NO and VEH_REGISTRATION_NO as the Key fields in the Define Key dialog box.

IDEA will run the duplicate check on these specific fields only.



The test gives cases of Policy Nos 11050180000075, 11050180000235 and 11050180000666 that have duplicate claims per policy.



CONCLUSION

Since these claims are towards settlement of partial losses, the list of duplicates may not reveal inconsistencies or irregularities.

Some policyholders may have genuinely suffered accidents repeatedly on the same vehicle as seen above. However, one cannot deny the need for an inquiry into these duplicates.

The internal audit team should take up these cases for further scrutiny with the claim file, surveyor reports, and picture evidence.

These cases serve to provide empirical evidence to r the audit team of negligent policyholders or policyholders who continuously run into bad luck, and validate future premium rate enhancements on such policies.

2.0 IDENTIFICATION OF DUPLICATE MOTOR CLAIMS DUE TO IRREGULARITIES AND DISCREPANCIES IN SURVEY

OBJECTIVE

To identify repetitive motor claims due to irregularities and / or discrepancies in the surveyor's report.

EXERCISE DESCRIPTION

In this exercise we will run tests for duplicate detection on the policy number and accident date field.

Prior to running the duplicate detection, we will isolate partial loss claims from the file of total and partial loss claims.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Identify Duplicates
- Extract data with a specified criterion.

Use the child file Partial Loss. IMD generated in the earlier exercise.

IDENTIFY DUPLICATES

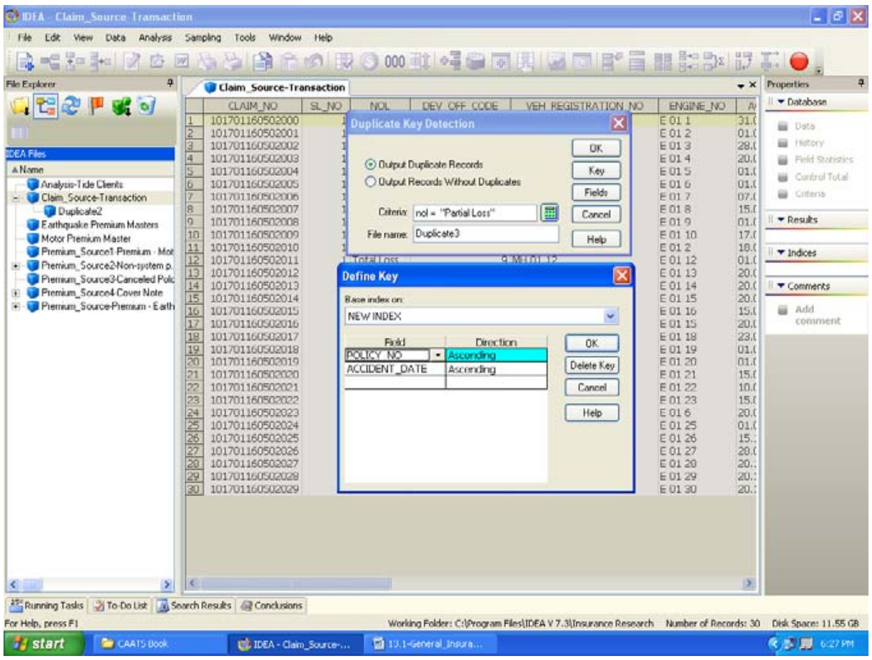
Run a duplicate key detection test on the file Partial Loss. IMD

The objective of the test is to identify claims having the same policy number and accident date.

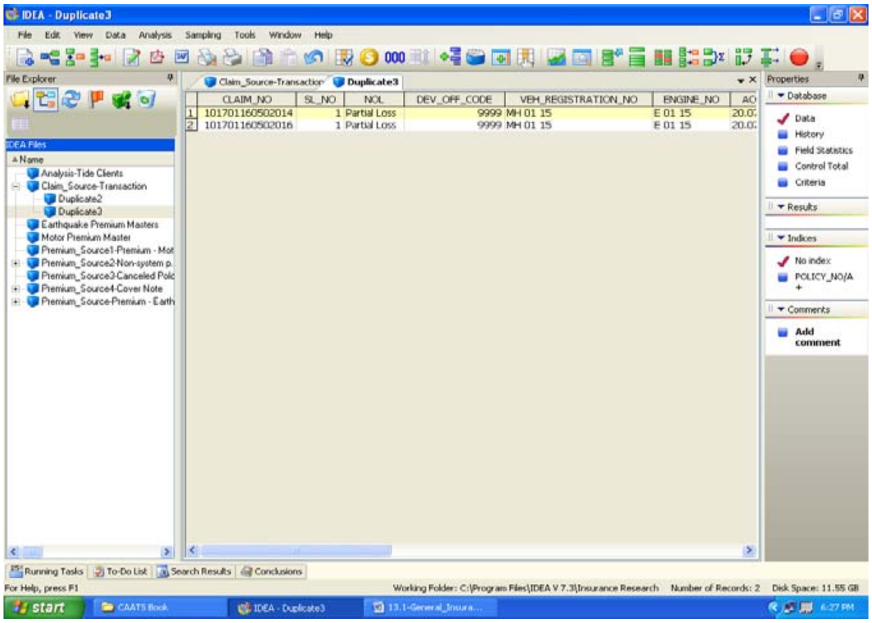
The reason for selecting the ACCIDENT_DATE as a critical key field is to give weightage to oversight, and / or deliberate manipulation in the survey results by the surveyor. It is not far fetched to visualize a situation in which the surveyor in connivance with the insured tries to fabricate an accident along with the required documents. To trap such cases we also select ACCIDENT_DATE as a critical field.

So select POLICY_NO and ACCIDENT_DATE as the Key fields in the Define Key dialog box.

IDEA will run the duplicate check on these specific fields only.



The test gives (2 cases or one case?) of Policy No 11050180000666, which has (witnessed?) claims twice.



CONCLUSION

The internal audit team should take up (these or this) (cases or case) for further scrutiny with the claim file, surveyor reports, and picture evidence.

If the internal audit head suspects long standing association between particular surveyors, insured and even development officers of the insurance company, then an attempt should be made to gather vital information on past dealings for such surveyors, insured and development officer to examine the length and depth of their involvement.

As a mode of building on a case for systematic well informed fraud allegation, the internal audit head should interview close associates of such surveyors, and development officers in strict confidence to attempt identifying rampant, extravagant changes in life style patterns, social habits and other references.

3.0 ISOLATION OF POLICIES EXPERIENCING CLAIMS SHORTLY WITHIN RISK COMMENCEMENT - CLOSE PROXIMITY

OBJECTIVE

To identify policies experiencing claims within a short span of risk commencement – Close Proximity Claims

EXERCISE DESCRIPTION

In this exercise, we will join the claim transaction file to the policy transaction file, so as to capture the risk commencement date and the accident date in a joined file.

We will then append a computed - virtual numeric field containing an @ Function to arrive at the difference between the risk commencement date and claim date.

Subsequently, we will extract those cases where claims have been submitted within 5 days of risk commencement, as per the difference in dates field computed above.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Join Files
- Append a Field
- Write an @ Function
- Extract data as per Logical Criterion

BACKGROUND

As per the internal guidelines, close proximity claims are claims, which appear within 5 days of risk inception against the relevant policy.

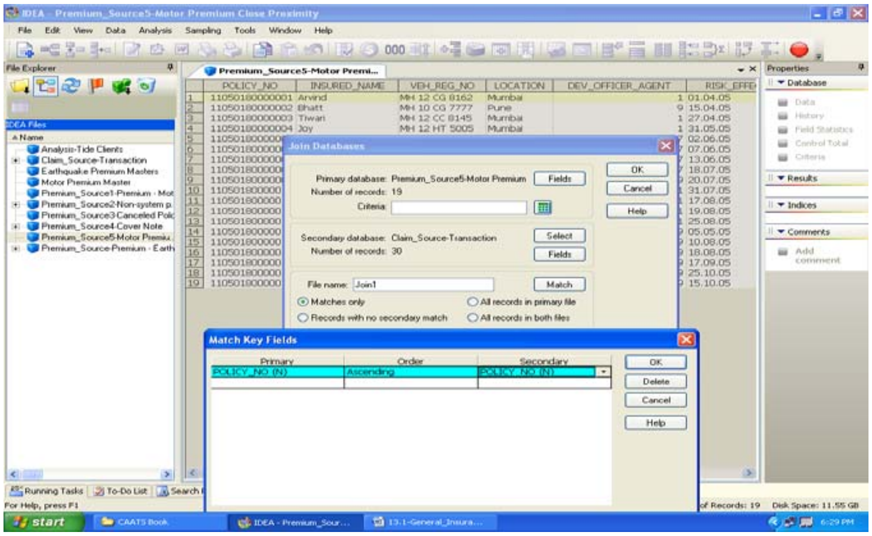
DATA IMPORT

Import the file Premium Source-Motor Premium Close Proximity. xls into IDEA as in IMPORT OF DATA FILES. (check for accuracy)

JOIN FILES

Designate Premium_Source 5 – Motor Premium Close Proximity. IMD as the Primary Database and Claim Transaction. IMD as the Secondary Database.

Select the Policy_No field from both the Databases as the matching field.



CONCLUSION

Close proximity claims should be viewed very carefully by the internal audit team for its occurrence, nature and frequency. The auditors should try to establish whether these claims have genuinely arisen due to the bad luck of the insured or whether they have resorted to any malpractice with the intent to deceive or defraud the insurance company. The role of the development officer and / or surveyor should also be investigated in such cases through discussions and inquiry. The auditor should apply the principles of Arms Length dealing and fair practices in such cases.

4.0 IDENTIFICATION OF CLAIMS WITHOUT BONAFIDE POLICY NUMBERS WITHIN THE SYSTEM – ORPHAN CLAIMS

OBJECTIVE

To identify claims processed and settled without bona fide policy numbers within the system.

EXERCISE DESCRIPTION

In this exercise, we will run an extraction to identify claims processed without matching policy numbers within the system.

The exceptions will be independently validated with the Premium Non-System Policies file to prove the existence and validity of claims.

IDEA FUNCTIONALITY COVERED
This exercise will show us how to:
• Extract data as per Logical Criterion
• Use the Search Function

BACKGROUND

Orphan claims are claims admitted, processed and paid, for which there are no matching policy numbers within the insurance application system.

These claims generally pertain to policies that have been booked prior to system migration from old legacy systems.

These claims need to be dealt with separately by the auditors, as there is no internal control mechanism for cross-reference within the system.

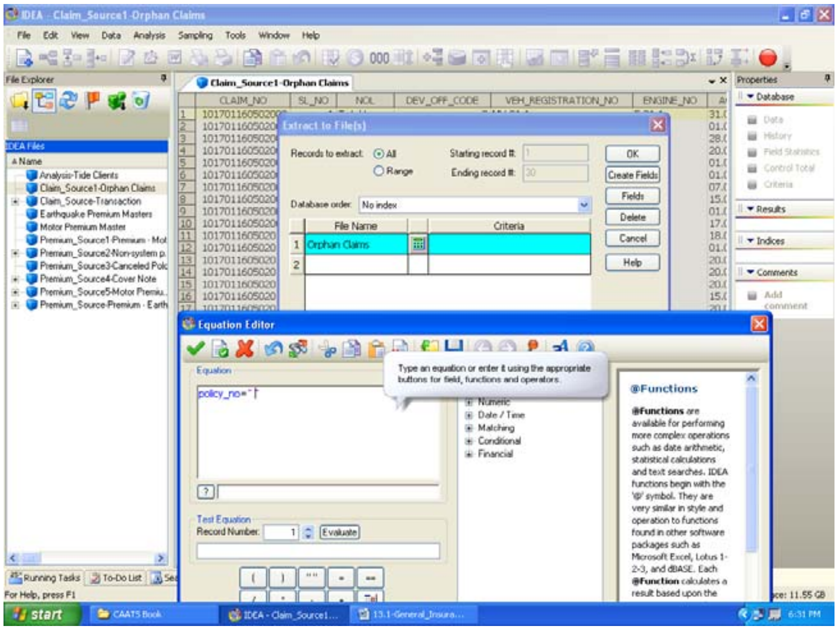
DATA IMPORT

Import the file Claim Source-Orphan Claims. xls into IDEA

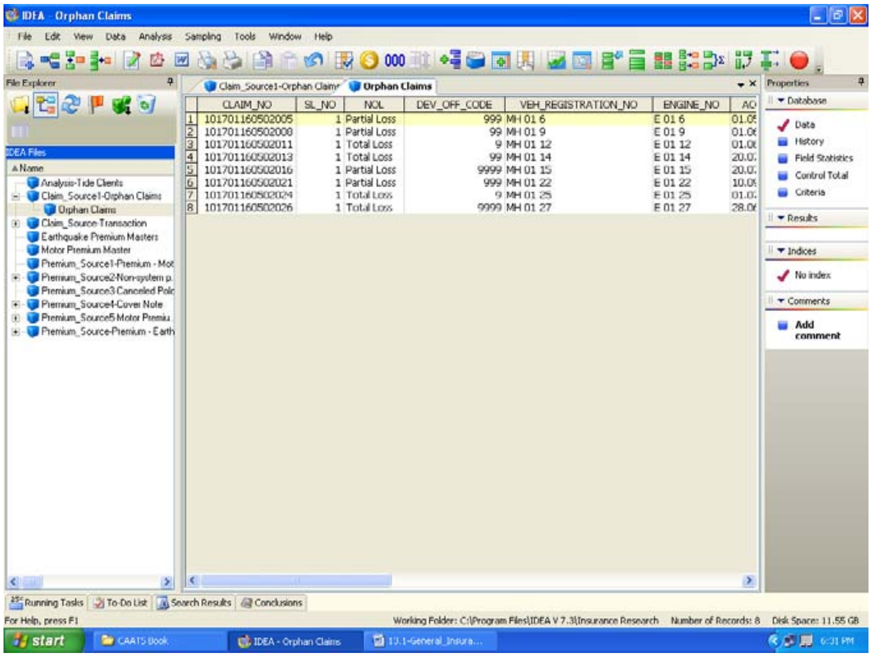
EXTRACT DATA AS PER LOGICAL CRITERION

Perform a data extraction through the equation

Policy_no=" "



The extraction will give us an independent file of 8 claims paid where the policy number field is blank.



USE THE SEARCH FUNCTION

To prove the validity of the 8 claims, compare the claim details with the available list of non-system policies.

A direct join with the Insured Name as the matching field may not yield the right results as consistency in the name entry may not be maintained by the development officers and the claim officers.

We now introduce the concept of DATA SEARCH where variants of names as entered by the claim officer can be located and reviewed.

Open the file Non system policies. prn. IMD in IDEA

The first claim paid is pertaining to Arjun S.

Now navigate to Search within Data

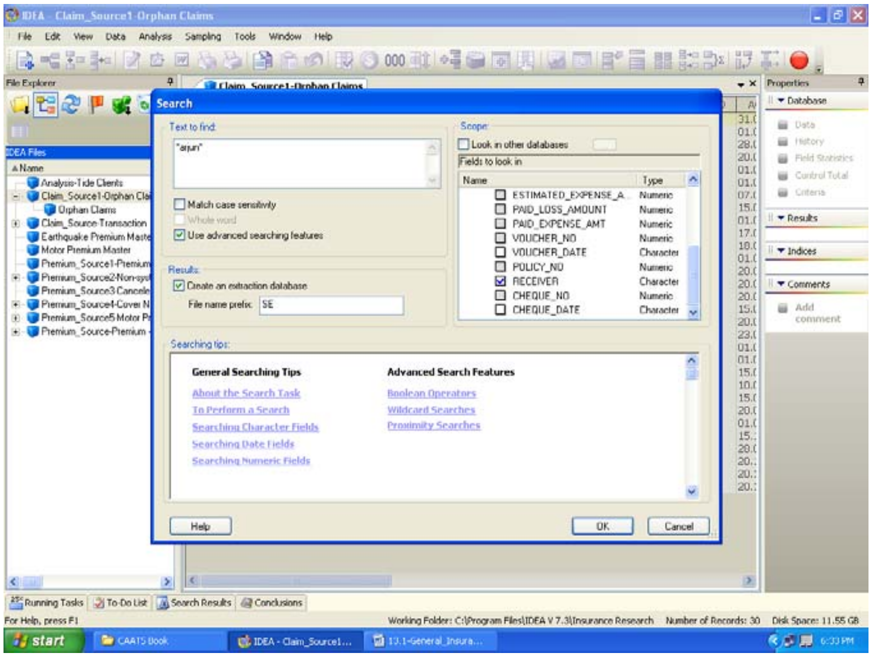
In the box 'Look for the following text' enter *arjun*.

The logic of using the string *arjun* is to include every possible variant and juxtaposition of the name and intial. * denotes a wild card character that can take any form.

The search needs to be made in the field INSURED. Check the field box for INSURED.

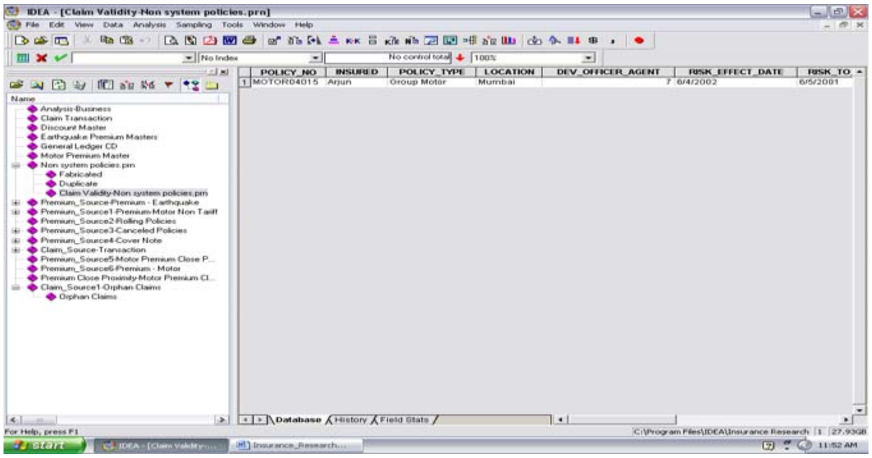
The search should be character case independent, so we do not check the box MATCH CASE SENSITIVITY

The output should be captured in an independent file, so check 'Create an extraction database'



The SEARCH gives us a single policy record with the name Arjun. This proves the claim validity.

If an attempt had been made to match the claim file and the policy file on the Insured field, no matches would have resulted, because the policy file has Arjun and the claim file has Arjun S.



CONCLUSION

Orphan cases need to be specially audited by the internal audit team.

Details should be sought from the claims officer and after running our own search tests as seen above, identifying the corresponding manual policy numbers.

The policies should be carefully studied and examined by the auditors to check on the insured, asset, period of coverage, sum insured and type of policy, to prove the validity of the claim.

5.0 AS PART OF OPERATIONAL STUDIES AND WORK FLOWS IDENTIFICATION OF IRREGULAR TIME LAGS IN CLAIM SETTLEMENT

OBJECTIVE

To identify irregular time lags in claim settlement due to process delays in survey.

EXERCISE DESCRIPTION

In this exercise, we will append computed – virtual numeric fields to arrive at differences between the accident date, surveyor intimation date, surveyor report receipt date and cheque paid date.

The day differences in each of the survey related activities will then be summarized surveyor-wise.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Append Fields
- Use the Equation Editor
- Apply Quick Field Summarization

DATA IMPORT

Import the file Claim Source-Surveyor Activity. xls into IDEA as in IMPORT OF DATA FILES (check?).

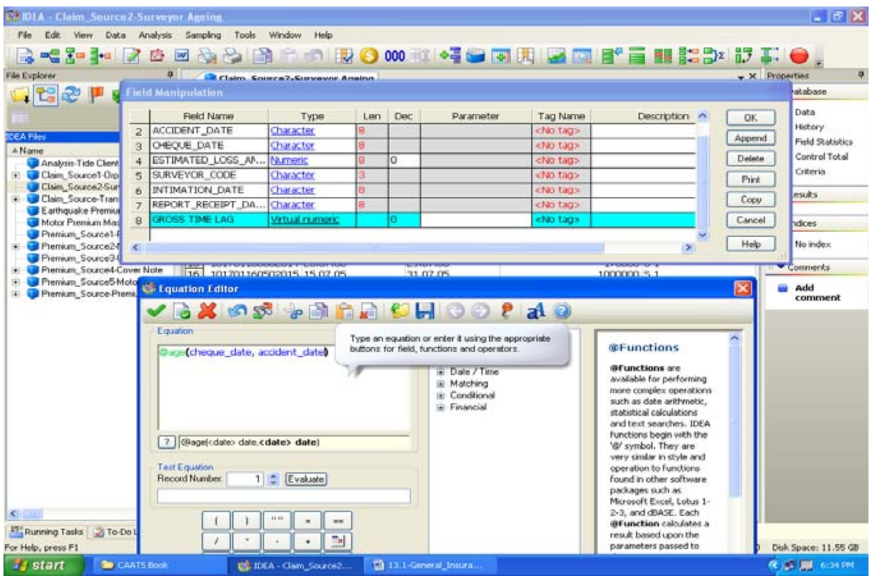
APPEND FIELDS, USE THE EQUATION EDITOR

Append 4 computed virtual numeric fields with the following specifics:

Field Name: Gross Time Lag

Type:Virtual Numeric

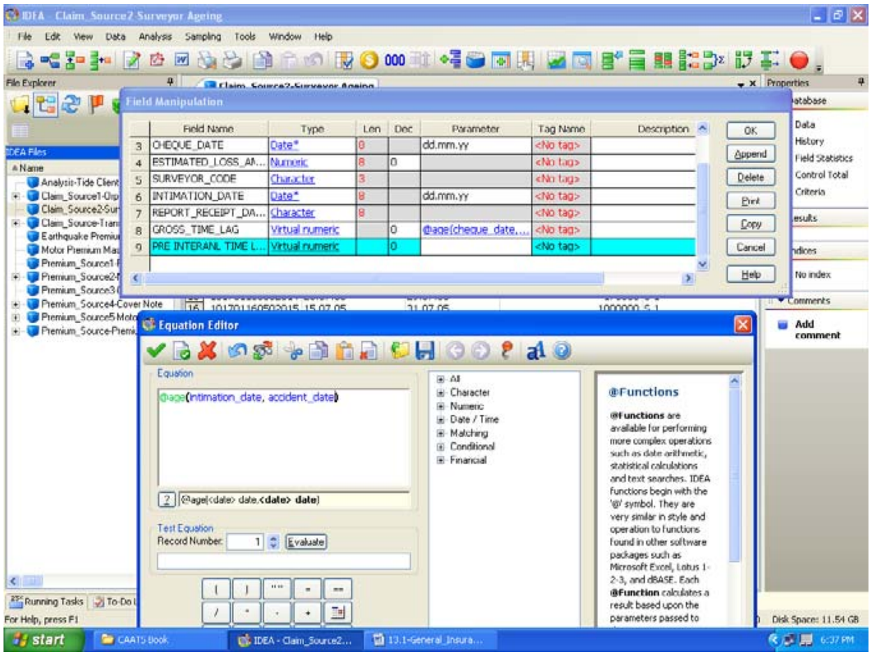
Parameter: @age (Cheque_Date, Accident_Date)



Field Name: Pre – Internal Time Lag

Type:Virtual Numeric

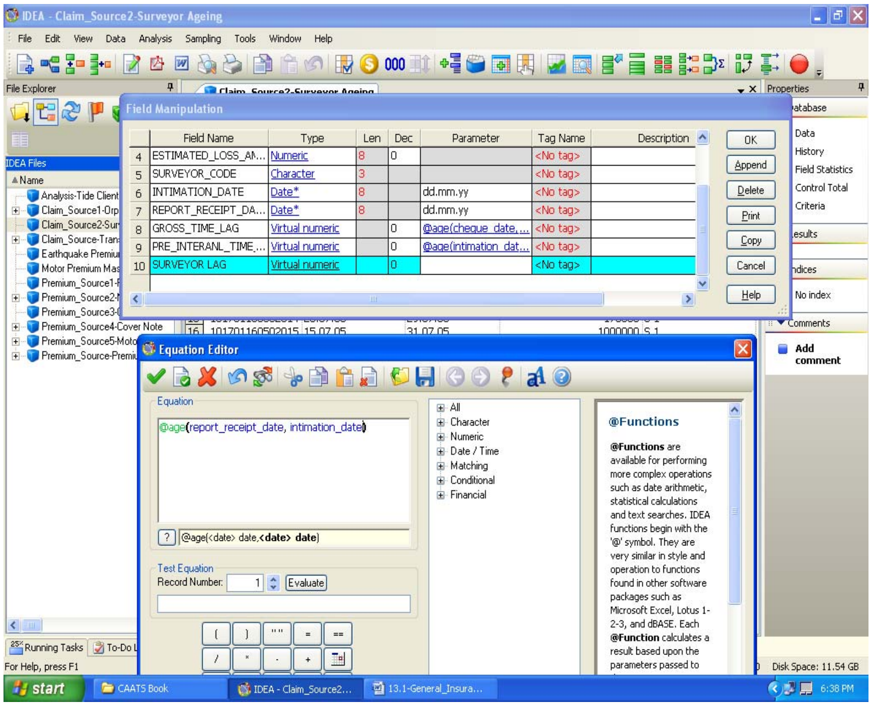
Parameter: @age (Intimation_Date, Accident_Date)



Field Name: Surveyor Lag

Type: Virtual Numeric

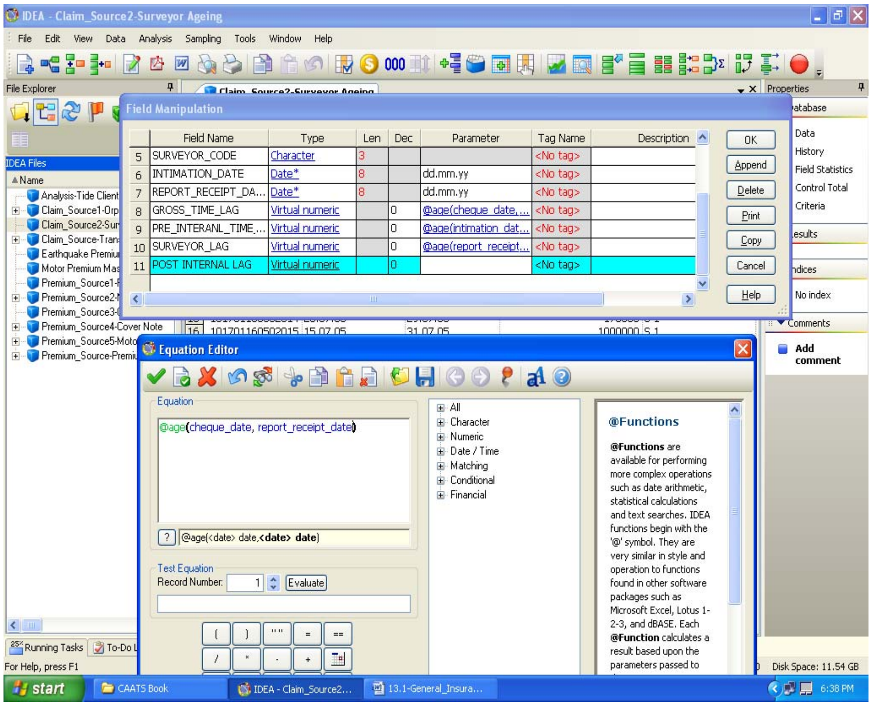
Parameter: @age (Report_Receipt_Date, Intimation_Date)



Field Name: Post – Internal Lag

Type: Virtual Numeric

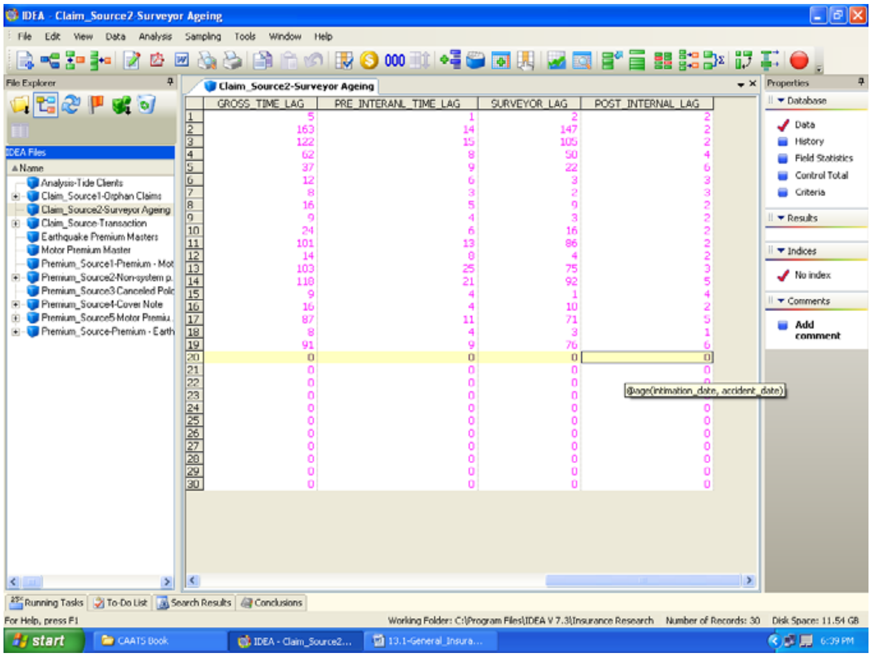
Parameter - @age (Cheque Date, Report_Receipt_Date)



APPLY QUICK FIELD SUMMARIZATION

The Claim Source-Surveyor Activity. IMD file appears, as can be seen below, after appending the virtual fields.

Navigate to Quick Field Summarization in the menu tool bar.



This opens the Quick (Unindexed) Field Summarization dialog box.

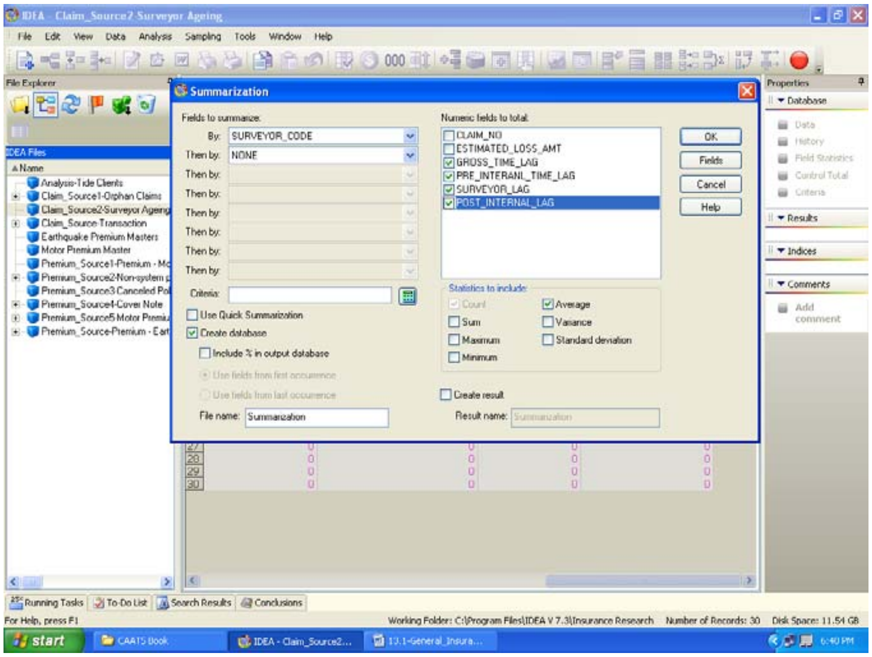
Select the SURVEYOR_CODE as the Summarization Field.

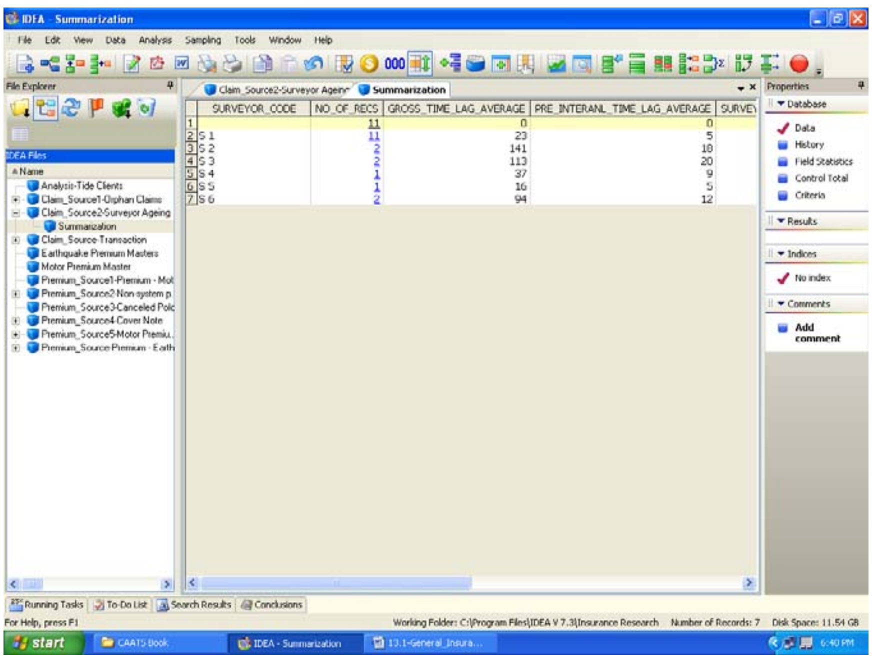
Select the Gross Time Lag, Pre Internal Time Lag, Surveyor Lag and Post Internal Time Lag as Fields to Summarize.

Click OK

This step will result in a ready summary of the internal stage wise time lags in the claim settlement process.

This summary will be available Surveyor wise.





The screen shows that Surveyor 'S2' has the highest absolute Surveyor Lag Average of 120 days for only 2 surveys.

CONCLUSION

Monitoring the Surveyor Lag can help in streamlining and disciplining the Surveyor's attempt to achieve efficient survey turnaround and output.

LEARNING

The same exercise can be carried out on the Pre-Internal and Post-internal Lag to isolate cases where claim officers have not issued intimation letters to surveyors and not acted upon survey reports within 7 days per claim case.

The @age function can help to identify delays, bottlenecks in specific functions, handled by specific individuals.

This can help in removing bottlenecks and improving claim throughput.

6.0 RE -VERIFICATION OF AGENCY COMMISSION ON BUSINESS SOLICITED

OBJECTIVE

To re-verify commission paid to agents on policy business solicited.

EXERCISE DESCRIPTION

In this exercise, we will run a data search on the agency commission transaction file to identify business solicited from Corporate Clients.

Then we will perform a data extraction to identify incorrect 'Individual' category Codes placed for corporate business.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Use the Search Function
- Extract data as per Logical Criterion
- Join Files

BACKGROUND

Agency commission varies based on the category of the insured, viz., corporate or individual.

DATA IMPORT

Import the file Premium Source-Commission. xls, Premium Source-Agency Commission. xls and Commission Master. dbf into IDEA as per the steps in IMPORT OF DATA FILES. (check?)

USE THE SEARCH FUNCTION

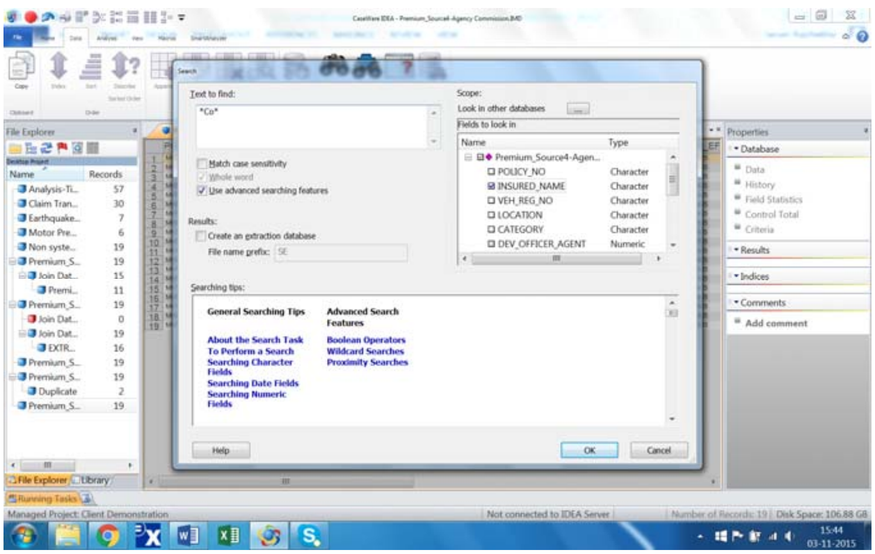
The file Premium Source-Agency Commission. IMD has a field CATEGORY. This field could either reflect Individual or Corporate, based on the status of the client.

The first test is to identify incorrect CATEGORY. That is, Individual category given to corporate clients.

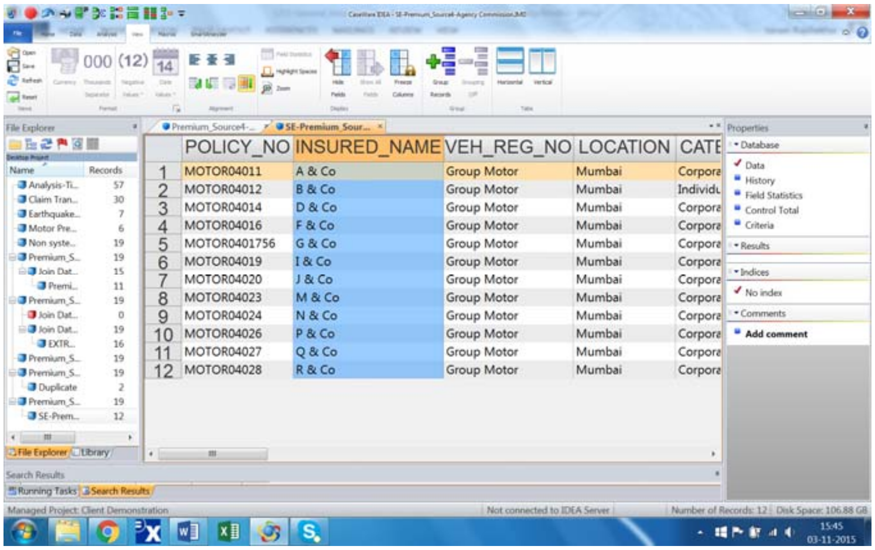
Locate all transactions where business has been solicited from Corporate Clients.

Do the same by writing a search command in the ‘Look for the following text box’ in the Search dialog box.

The search command is * Co



The resultant file is a file containing business solicited from corporate clients, as is evident from the ‘& Co’ notation in the Insured Name field.

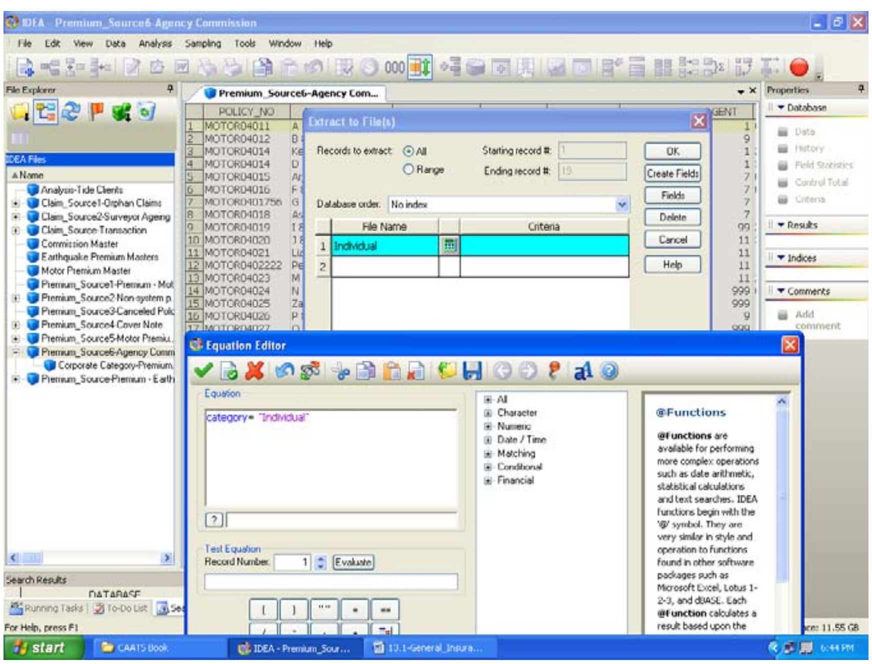


EXTRACT DATA AS PER LOGICAL CRITERIA

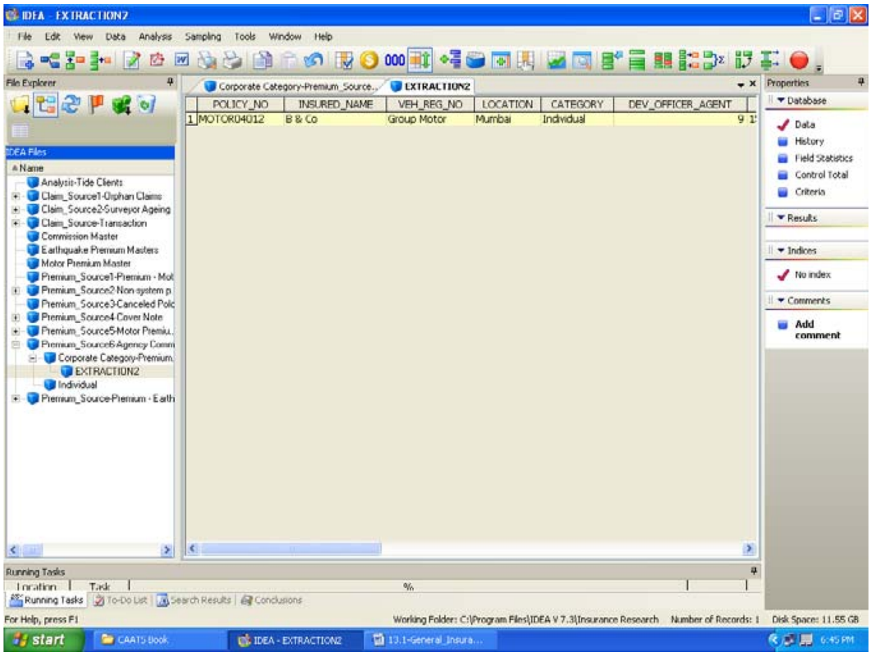
Extract data to a separate file through an equation

Category = ‘Individual’

This is to identify incorrect Individual category given to Corporate Clients.



One case is shown where an Individual appears in the CATEGORY for corporate clients.



In this particular case, represented by Policy No MOTOR04012, 3% commission (Individual) has been paid to the agent in place of 5% (Corporate). (Source Commission Master. IMD)

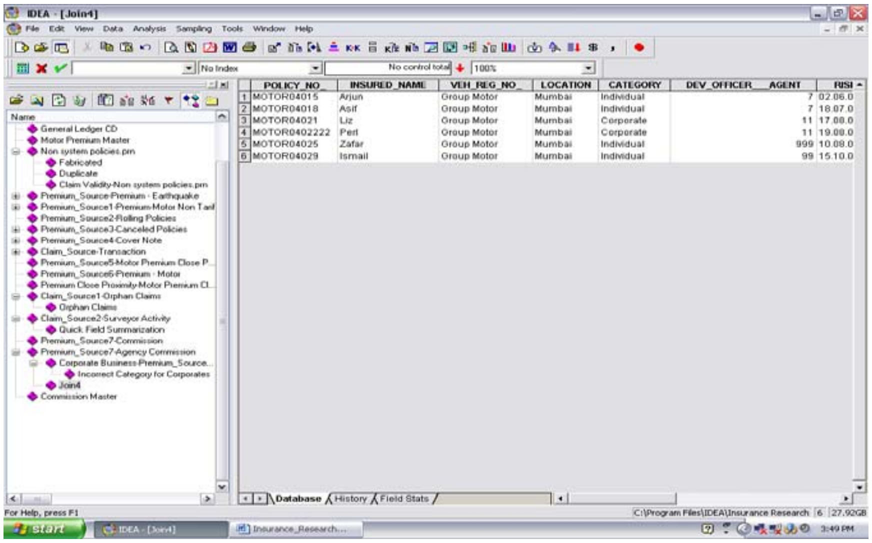
JOIN FILES

Designate Premium_Source – Agency Commission. IMD as the Primary Database and Corporate Business-Premium Source. IMD as the Secondary Database.

Select the Policy_No field from both the Databases as the matching field.

Check the option Records with no secondary match to identify and isolate all business solicited other than corporate from the combined file of corporate and individual.

The resultant file should contain only INDIVIDUAL category transactions.



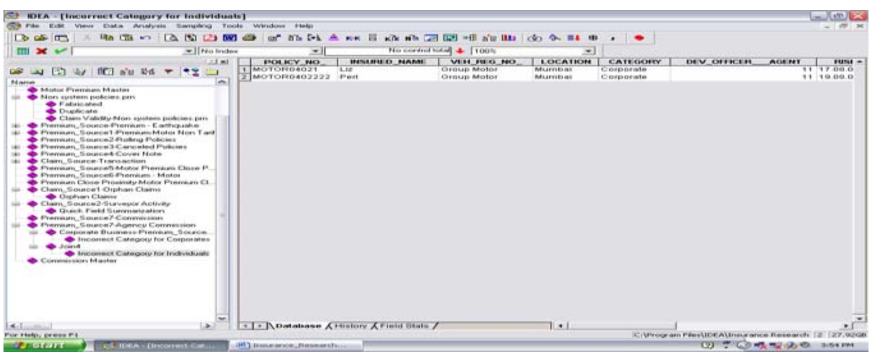
EXTRACT DATA AS PER LOGICAL CRITERIA

Extract data to a separate file through an equation

Category = 'Corporate'

This is to identify incorrect **Corporate** category given to individual clients.

The extraction gives 2 resultant exceptions to the logical expression.



In the specific cases represented by Policy No MOTOR04021 and MOTOR0402222, 5% commission (Corporate) has been paid to the agent in place of 3% (Individual).

CONCLUSION

The Category field determines the agency commission.

The Category field can be altered to reflect a category not in line with the insured status as can be seen above.

Hence the tests for agency commission have to be run on the Insured field to gather the true status of the insured.

7.0 IDENTIFICATION OF MOTOR INSURANCE BUSINESS CLAIMS SETTLED ON POLICIES PERTAINING TO STOLEN VEHICLES

OBJECTIVE

To identify if claims have been settled on policies, insuring stolen vehicles as per the National Crime Records Bureau (NCRB) alert file of stolen vehicles

EXERCISE DESCRIPTION

In this exercise, we will join the claim transaction file with the NCRB list of stolen vehicles on the basis of the vehicle engine number.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

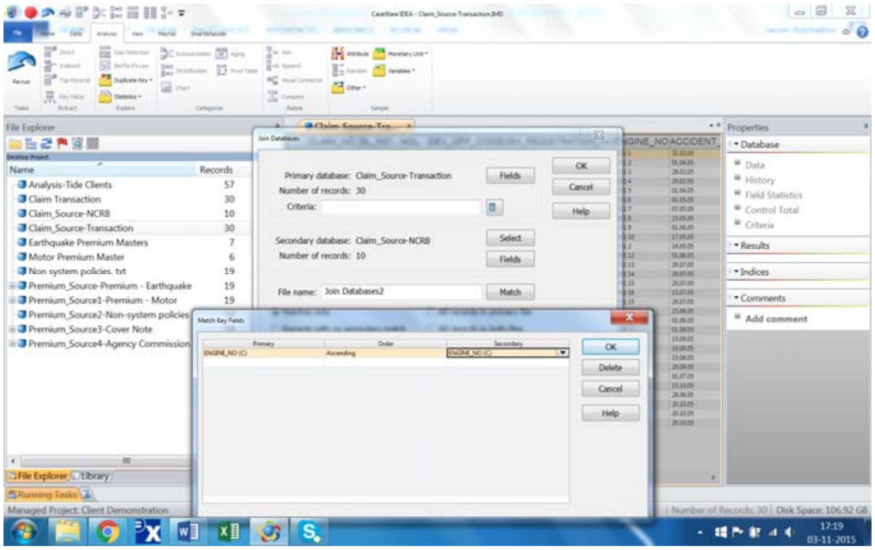
- Join Files

DATA IMPORT

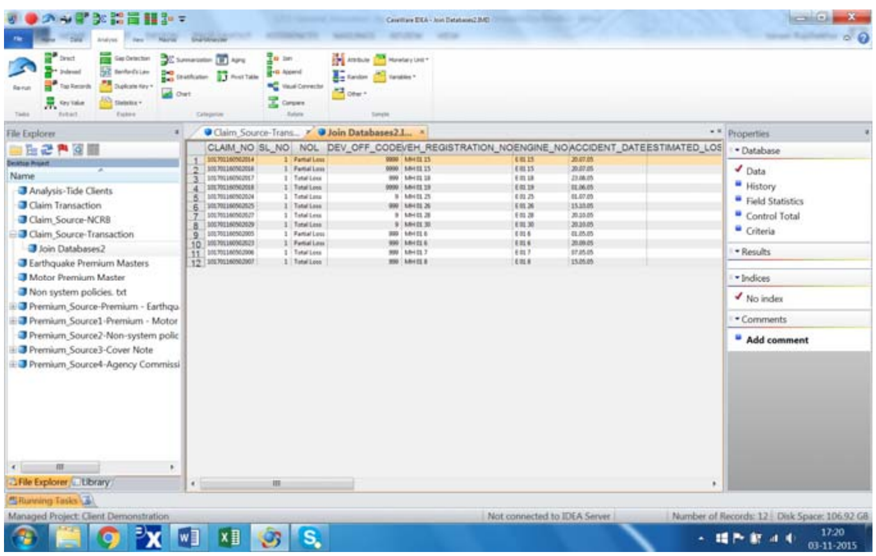
Import the file Stole Vehicles. prn into IDEA as per the steps demonstrated in the session on IMPORT OF DATA FILES.

JOIN FILES

The file Claim Source-Transaction. IMD (Primary File) is joined to the file Claim Source – NCRB. IMD (Secondary File) through File > Join Databases. When a vehicle is stolen the vehicle registration number undergoes a change but the Engine_no does not change, hence the matching key is Engine_No. The process is seen below.



After joining the two files we get a list of 12 policies, pertaining to 12 stolen vehicles against which claims have been admitted, processed and settled. Details about these can be seen below. The value of such claims (Paid Loss Amount) from the control totals amounts to Rs. 87,29,812.50



CONCLUSION

The development officer and agents should be provided with the current and dynamic list of stolen vehicles every fortnight from the NCRB.

Preventive checks should be run by way of Vehicle Engine No. to ensure that no stolen vehicles are being insured by the branch.

13.1.8 Business Analysis Case Studies

1.0 ANALYSIS BY CLIENT FOR DIFFERENT CLASSES OF RISKS INSURED

OBJECTIVE

To analyze the premiums collected, claims paid and losses sustained per insured for different classes of risk insured.

EXERCISE DESCRIPTION

In this exercise, we will create a Pivot Table for two variables: Class of Risk and Customer ID.

These variables will be analyzed for the main heads premium, claims and losses incurred.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Create a Pivot Table

DATA IMPORT

Import the file Analysis Business. xls into IDEA

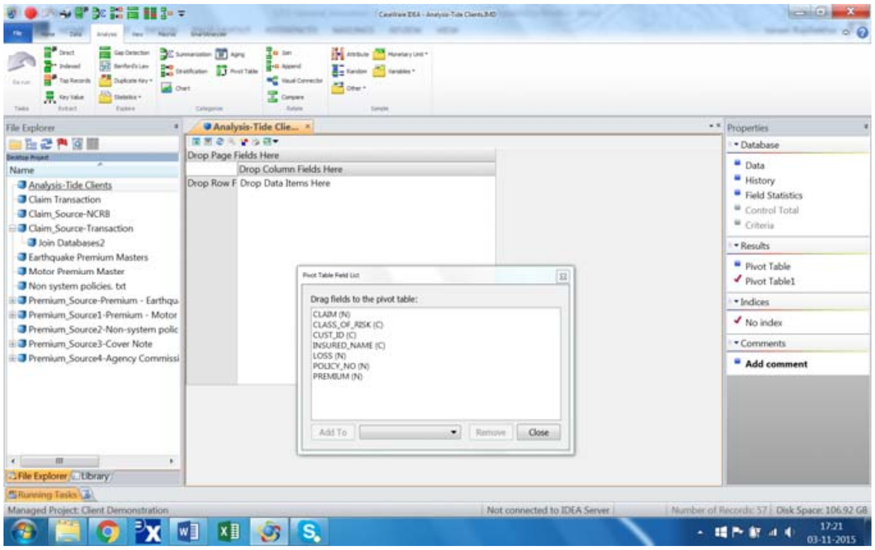
CREATE A PIVOT TABLE

In this exercise, capture two key variables Cust_ID and Class_of_Risk.

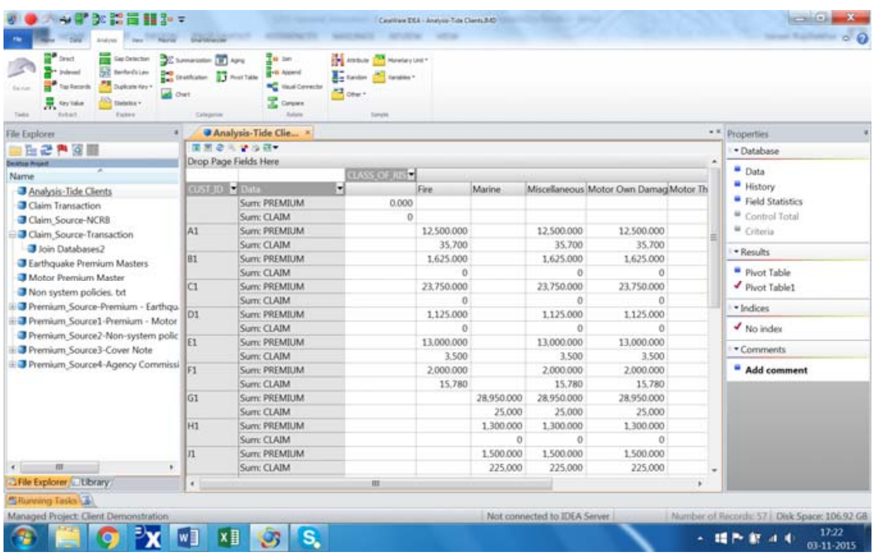
Proceed on the basis of the Insured_Name because there may not be consistency in the naming convention for the Insured_Name across Class_of_Risk.

The variables need to be analyzed across numeric fields Premium, Claim and Loss. Normally multi-key analysis is done through the Pivot Table feature in IDEA 9.

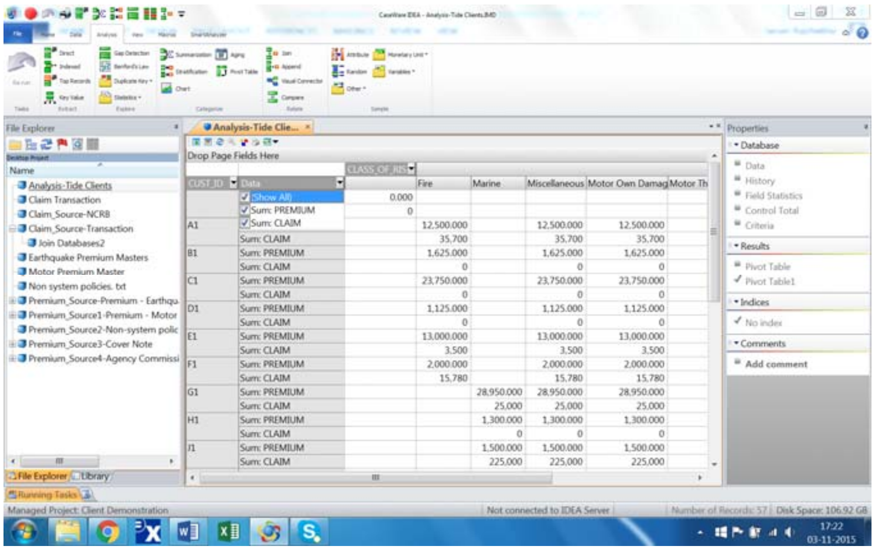
The Pivot Table is invoked through the Analysis option on the menu tool-bar. Once invoked, it appears like what is seen below



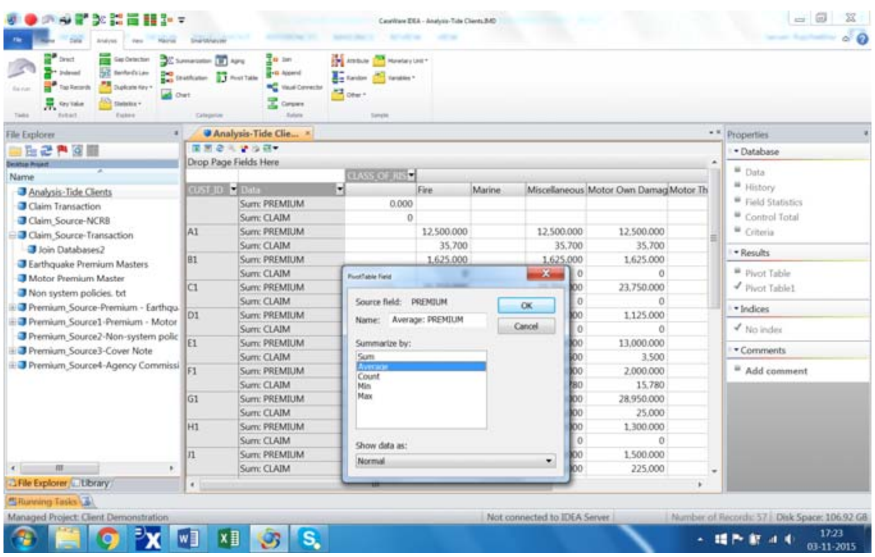
The Cust_ID from the Pivot Table Field List is picked and dropped in the Row section. The Class_of_Risk from the Pivot Table Field List is picked and dropped in the Column section. Finally the numeric fields – Claim, Premium and Loss are picked one by one and dropped in the Data Grid section of the Pivot Table. The result is seen below as a tab in the file Analysis-Tide



By placing the cursor of the mouse over the Row Header 'Data' and right clicking it, it invokes the drop down list of Numeric fields, which can be checked or unchecked, as is seen below, to change the view. We uncheck Premium and Claim to just reflect the Loss position.

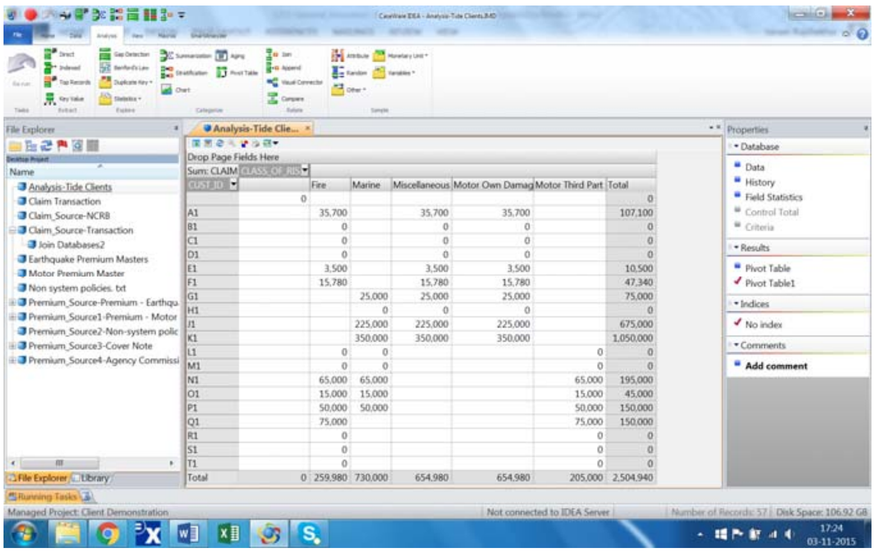


By placing the cursor of the mouse over the actual Data Grid and right clicking on any data cell, it invokes the drop down list of Pivot Table field list containing significant arithmetical functions like sum, min, max, count and average. A single function can be selected, as is seen below, to change the view. We select Average to reflect the average loss sustained per Cust_ID per Class_of_Risk.



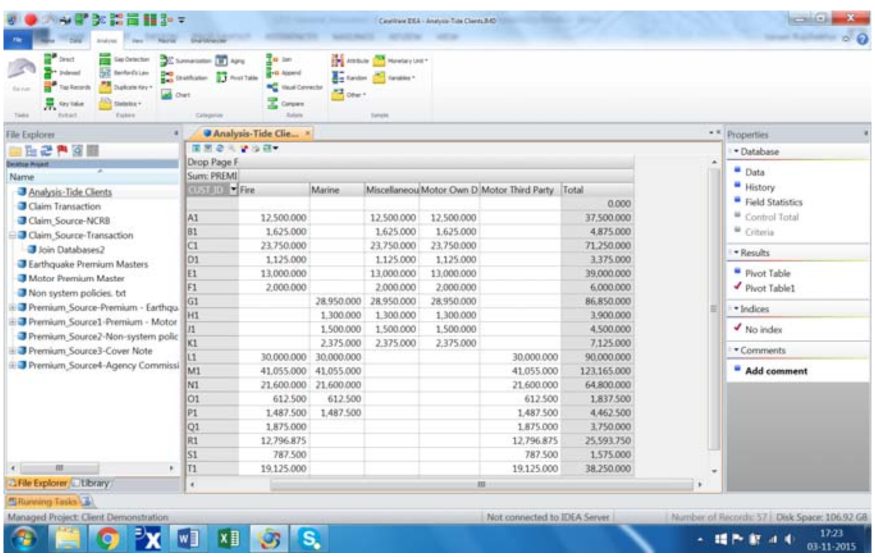
In the Fire risk category, Cust_ID Q1 seems to have caused the highest damage to the branch of Rs. 73,125.

The branch has taken a significant hit on Cust_ID K1 for three categories of risks Marine, Miscellaneous and Owned Motor of Rs. 347625, 347625 and 347625 respectively. The Owned Motor risk category seems exceptionally high in comparison with the other Cust_ID's.



CUST_ID		Fire	Marine	Miscellaneous	Motor Own Damag	Motor Third Part	Total
	0						0
A1		35,700		35,700	35,700		107,100
B1		0		0	0		0
C1		0		0	0		0
D1		0		0	0		0
E1		3,500		3,500	3,500		10,500
F1		15,780		15,780	15,780		47,340
G1			25,000	25,000	25,000		75,000
H1			0	0	0		0
J1			225,000	225,000	225,000		675,000
K1			350,000	350,000	350,000		1,050,000
L1		0	0			0	0
M1		0	0			0	0
N1		65,000	65,000			65,000	195,000
O1		15,000	15,000			15,000	45,000
P1		50,000	50,000			50,000	150,000
Q1		75,000				75,000	150,000
R1		0				0	0
S1		0				0	0
T1		0				0	0
Total	0	259,980	730,000	654,980	654,980	205,000	2,504,940

Upon re-viewing the Pivot Table for summary of Premiums across Cust_ID and Class_of_Risk, we get the position seen below.



CUST_ID	Fire	Marine	Miscellaneou	Motor Own D	Motor Third Party	Total
						0.000
A1	12,500.000		12,500.000	12,500.000		37,500.000
B1	1,625.000		1,625.000	1,625.000		4,875.000
C1	23,750.000		23,750.000	23,750.000		71,250.000
D1	1,125.000		1,125.000	1,125.000		3,375.000
E1	13,000.000		13,000.000	13,000.000		39,000.000
F1	2,000.000		2,000.000	2,000.000		6,000.000
G1		28,950.000	28,950.000	28,950.000		86,850.000
H1		1,300.000	1,300.000	1,300.000		3,900.000
J1		1,500.000	1,500.000	1,500.000		4,500.000
K1		2,375.000	2,375.000	2,375.000		7,125.000
L1	30,000.000	30,000.000			30,000.000	90,000.000
M1	41,055.000	41,055.000			41,055.000	123,165.000
N1	21,600.000	21,600.000			21,600.000	64,800.000
O1	612.500	612.500			612.500	1,837.500
P1	1,487.500	1,487.500			1,487.500	4,462.500
Q1	1,875.000				1,875.000	3,750.000
R1	12,796.875				12,796.875	25,593.750
S1	787.500				787.500	1,575.000
T1	19,125.000				19,125.000	38,250.000

M1, N1, L1, G1, and C1 seem to be the Tide Clients drawing in cumulative premium business in excess of Rs. 15000 per annum per class_of_risk

Of the above Tide Clients a careful examination reveals that N1, seems to be causing heavy losses to the branch despite contributing handsome premiums to the Tide Client kitty.

CONCLUSION

The branch should inform this case to the Region and take a concerted decision to raise the risk rating on N1, thereby hiking up the premium rates substantially on future business accepted from N1.

A careful examination of J1 and K1 reveals that these customers are contributing insignificant premium to the branch (average of Rs. 1500 and 2375 respectively per class), but are invoking heavy back breaking losses on the branch (average of Rs. 223500, and 347625 per class).

This dis-proportionate premium to loss result is placing an enormous burden on the branch.

2.0 ANALYSIS BY DEVELOPMENT OFFICER AND / OR AGENT OF INSURED LOSSES

OBJECTIVE

To analyze the losses sustained in motor class risk for policies underwritten by different development officers and / or agents.

EXERCISE DESCRIPTION

In this exercise, we will join the agent-wise incurred loss file with the file for tide clients to capture the policy number, agent code and incurred loss together.

This file will then be analyzed through a Pivot Table for the business soliciting patterns of agents that burden the branch with heavy claims and losses.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

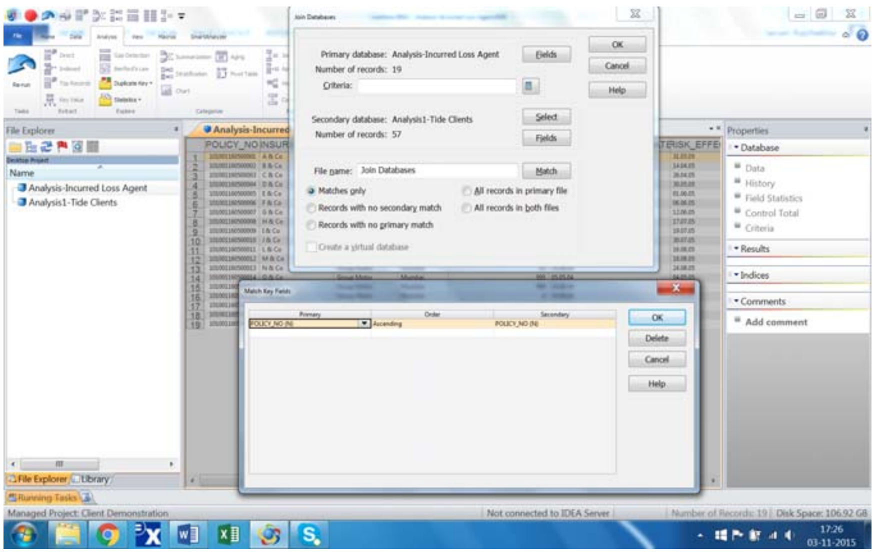
- Join Files
- Create a Pivot Table

DATA IMPORT

Import the file Analysis Incurred Loss Agent. xls into IDEA as in IMPORT OF DATA FILES.(Check)

JOIN FILES

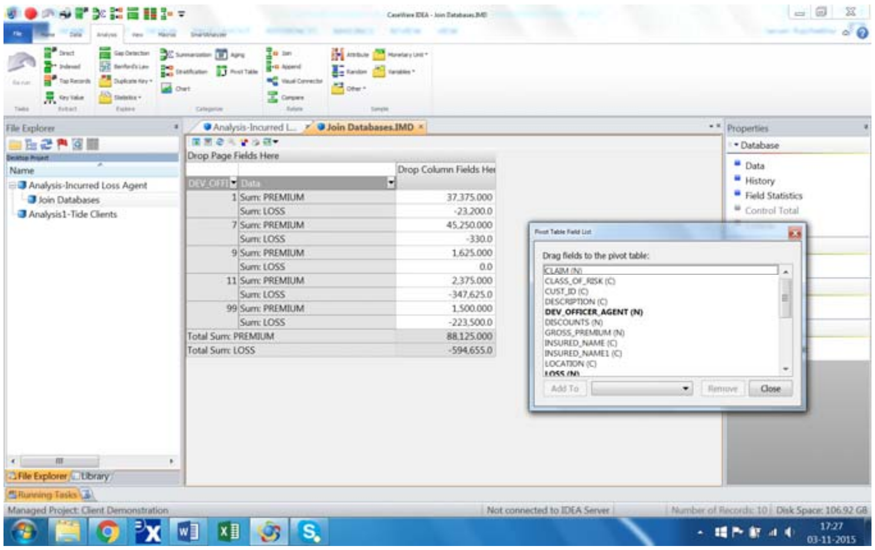
Join the files Analysis Incurred Loss Agent. IMD (primary file) with the file Analysis Business. IMD (secondary file) on the key field POLICY NO, with 'Matches only' as the joining option.



CREATE A PIVOT TABLE

Run a Pivot Table on the joined file generated.

The Development Officer / Agent Code is captured in the rows section and the numeric fields premium, claim and loss incurred are captured in the data grid.



From the above it can be seen that Dev Officer 99 and Dev Officer 11 have brought in policies that have saddled the branch with enormous losses.

Dev Officer 99 has brought in a motor policy of Rs. 1500 premium for which the loss incurred in the year is Rs. 223500.

Dev Officer 11 has brought in a motor policy of Rs. 2375 premium for which the loss incurred in the year is Rs. 347625.

CONCLUSION

These cases should be specifically reported and taken up for close scrutiny.

The past trend for these policies should also be viewed carefully to examine history of repetitive losses. It is quite possible that the current losses were one-time abrupt losses due to bad luck. .

Based on the findings, the development officer should be informed and asked to defend his decision to solicit policies experiencing repetitive claims.

In dire circumstances the development officer may have to be asked to step down from his post because his activities are blatantly hard hitting on the finances of the branch.

3.0 IDENTIFICATION OF PREMIUM ENHANCEMENT CASES FOR LOSS MAKING AND HIGH-RISK POLICIES

OBJECTIVE

To identify trends in policies experiencing repetitive losses for over 3 years.

The evaluation of such cases for premium enhancement.

EXERCISE DESCRIPTION

In this exercise, we will learn to use Visual Connect to join more than two files at the same time.

The files will be joined based on the key 'Insured Name'.

Trends in incurred losses will be identified and acted upon.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to

- Use Visual Connect

DATA IMPORT

Import the file Analysis Incurred Loss 2005. xls, Analysis Incurred Loss 2004. xls and Analysis Incurred Loss 2003. xls into IDEA

USE VISUAL CONNECT

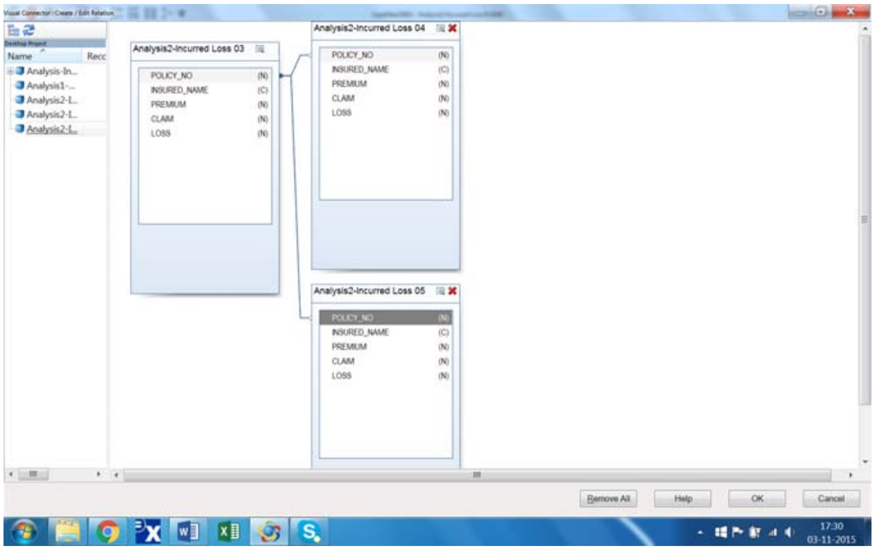
Use the Visual Connect feature in IDEA 9 to join the files on the basis of the Key 'Insured_Name'.

The files reflect renewal policies from 2003 to 2005, so it can be taken as given that the Insured_Name field will stay the same over the three years.

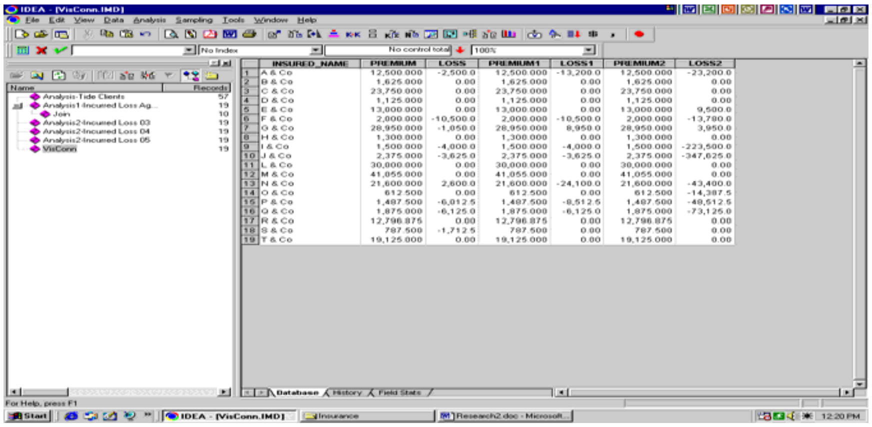
Invoke the Visual Connect feature from Analysis > Visual Connect.

Open the three files consecutively in the Visual Connect dialog box.

Join the three files together based on the Insured_Name field by linking the output node from the Analysis-Incurred Loss 03 file to the input node from the Analysis-Incurred Loss 04 file, and the output node from the Analysis-Incurred Loss 04 file to the input node from the Analysis-Incurred Loss 05 file.

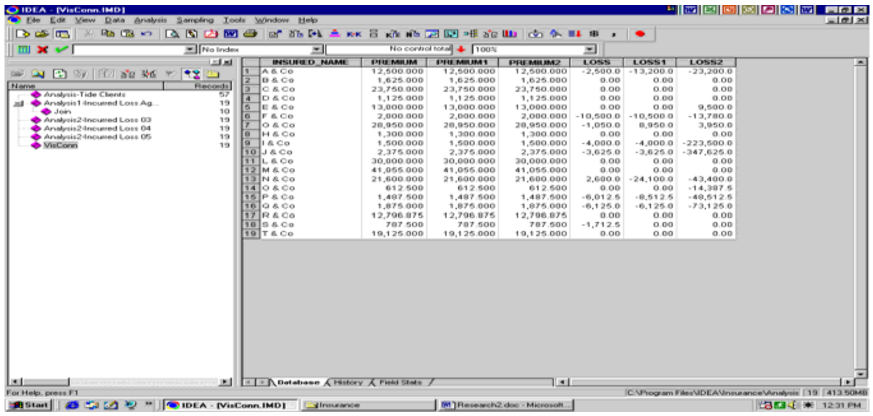


After Visual Connect the joined file appears as below –



	INSURED_NAME	PREMIUM	LOSS	PREMIUM1	LOSS1	PREMIUM2	LOSS2
1	A & Co	12,500.000	-2,500.0	12,500.000	-13,200.0	12,500.000	-23,200.0
2	B & Co	1,625.000	0.00	1,625.000	0.00	1,625.000	0.00
3	C & Co	23,750.000	0.00	23,750.000	0.00	23,750.000	0.00
4	D & Co	1,125.000	0.00	1,125.000	0.00	1,125.000	0.00
5	E & Co	13,000.000	0.00	13,000.000	0.00	13,000.000	9,500.0
6	F & Co	2,000.000	-10,500.0	2,000.000	-10,500.0	2,000.000	-13,780.0
7	G & Co	28,950.000	-1,050.0	28,950.000	8,950.0	28,950.000	3,950.0
8	H & Co	1,300.000	0.00	1,300.000	0.00	1,300.000	0.00
9	I & Co	1,500.000	-4,000.0	1,500.000	-4,000.0	1,500.000	-223,500.0
10	J & Co	2,375.000	-3,625.0	2,375.000	-3,625.0	2,375.000	-347,625.0
11	L & Co	30,000.000	0.00	30,000.000	0.00	30,000.000	0.00
12	M & Co	41,055.000	0.00	41,055.000	0.00	41,055.000	0.00
13	N & Co	21,600.000	2,600.0	21,600.000	-24,100.0	21,600.000	-43,400.0
14	O & Co	612.500	0.00	612.500	0.00	612.500	-14,387.5
15	P & Co	1,487.500	-6,012.5	1,487.500	-8,512.5	1,487.500	-48,512.5
16	Q & Co	1,875.000	-6,125.0	1,875.000	-6,125.0	1,875.000	-73,125.0
17	R & Co	12,796.875	0.00	12,796.875	0.00	12,796.875	0.00
18	S & Co	787.500	-1,712.5	787.500	0.00	787.500	0.00
19	T & Co	19,125.000	0.00	19,125.000	0.00	19,125.000	0.00

By re-formatting the above file to position the Loss fields and the premium fields together, we get a clear linear view of the regular loss making cases, as can be seen below



Take the first case, for example, of Insured A & Co. There has been a steady increase in the Losses Incurred from Rs. 2500 in 2003 to Rs. 13,200 in 2004 and Rs. 23,200 in 2005.

Despite this steady increase in losses due to higher and repetitive claims over the years, the premium has not been appropriately enhanced to reflect the high-risk profile of the Insured.

The same premium of Rs. 12,500 has been maintained from 2003 to 2005.

The same is true of N & Co, P & Co and Q & Co. These cases falling under the non-tariff segment of motor insurance can be treated appropriately as sensitive risk profiles and the premiums on such cases can be suitably hiked up to justify the high risk nature of the Insured.

CONCLUSION

By making a running comparison of Insured Losses for 3-5 years, the audit department can arrive at a well-based intelligence on specific policies, loss assets and claim patterns.

Given this intelligence, the audit department can strongly recommend such cases for premium enhancement in the best interest of the branch (thereby delivery its role as a value added function this part is not clear).

13.1.9 Finance & Accounts Case Studies

1.0 RECONCILIATION OF SUB LEDGER AND GENERAL LEDGER FOR PRINCIPAL CONTROL ACCOUNTS

OBJECTIVE

To reconcile the Cash Deposit Control Account in the General ledger with the Cash Deposit Sub Ledger to establish the accuracy of transactions recorded and posted.

EXERCISE DESCRIPTION

In this exercise, we will learn to add a computed virtual numeric field in the sub-ledger file to arrive at the month for each transaction undertaken.

This computed field giving monthly results would be based on the transaction date field.

Then we will summarize the sub-ledger on the basis of the month field to arrive at gross transaction details-debit and credit per month.

This summarization will finally be compared with the General Ledger monthly summary.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Tile Files
- Append a Field
- Use Equation Editor
- Write @ Functions
- Perform a Quick Field Summarization

DATA IMPORT

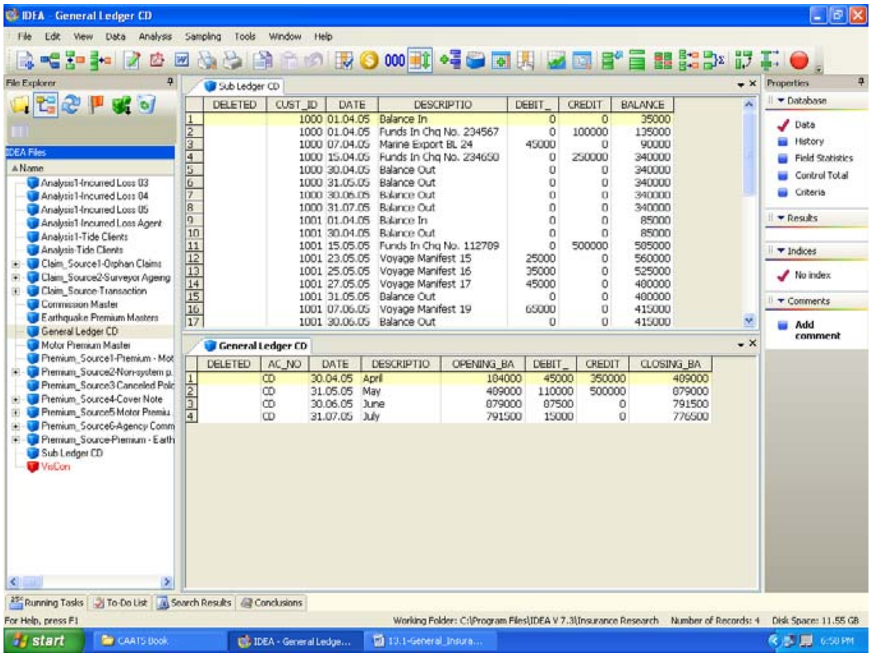
Import the file Analysis Sub Ledger CD. Dbf and General Ledger CD. dbf into IDEA as in IMPORT OF DATA FILES.(check?)

TILE FILES

Import the files as specified.

Navigate to window and select TILE from the windows menu

Tiling allows the files to be viewed concurrently on the screen.



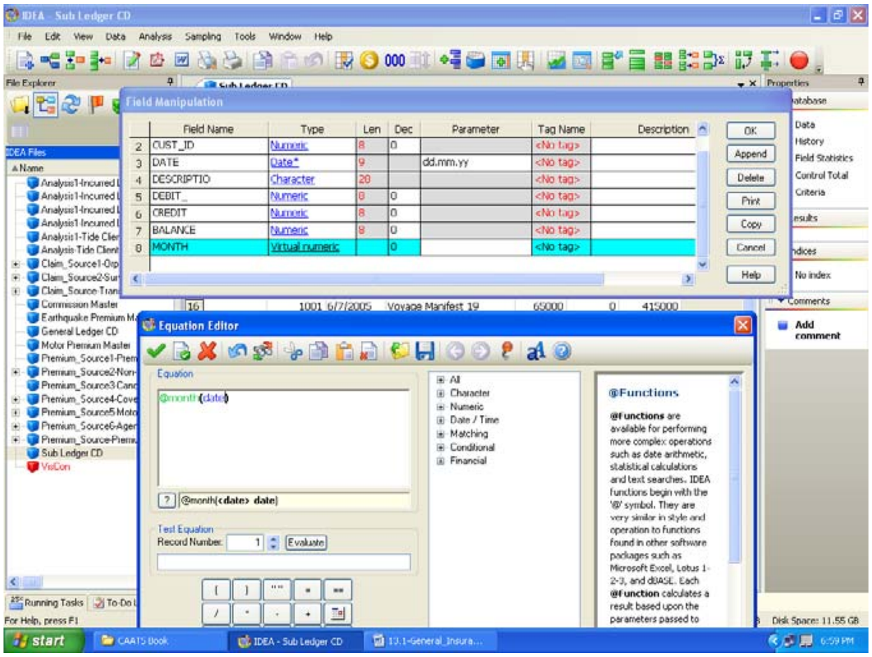
We are required to reconcile the two files on the monthly totals.

APPEND A VIRTUAL FIELD, USE THE EQUATION EDITOR AND WRITE AN @ FUCNTION

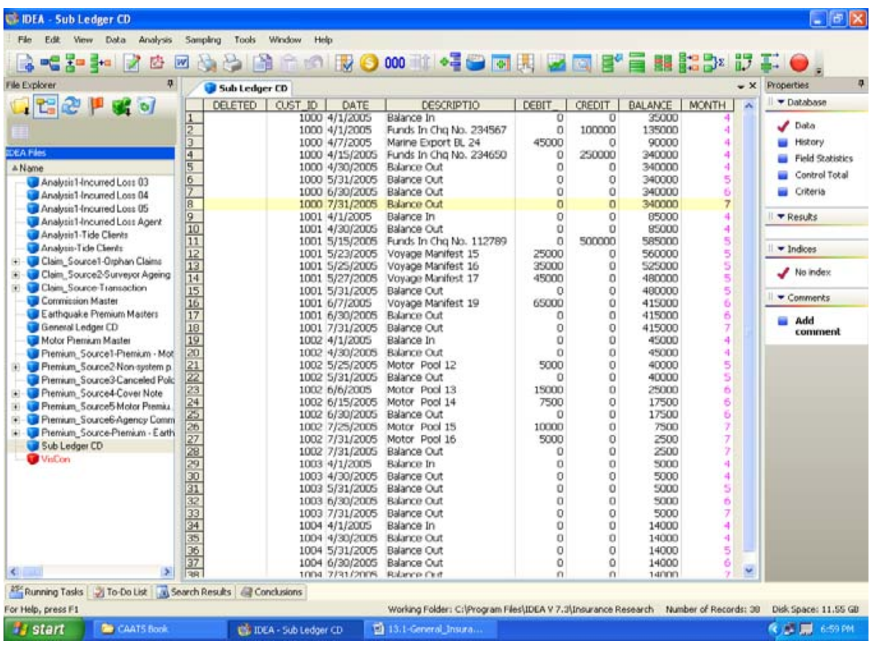
Add a virtual numeric field called ‘MONTH’ to the file Sub Ledger CD. IMD.

In this field write the equation @month (date) to convert all the transaction dates into month details.

This is with the objective of summarization of balances, debit and credit fields month- wise.



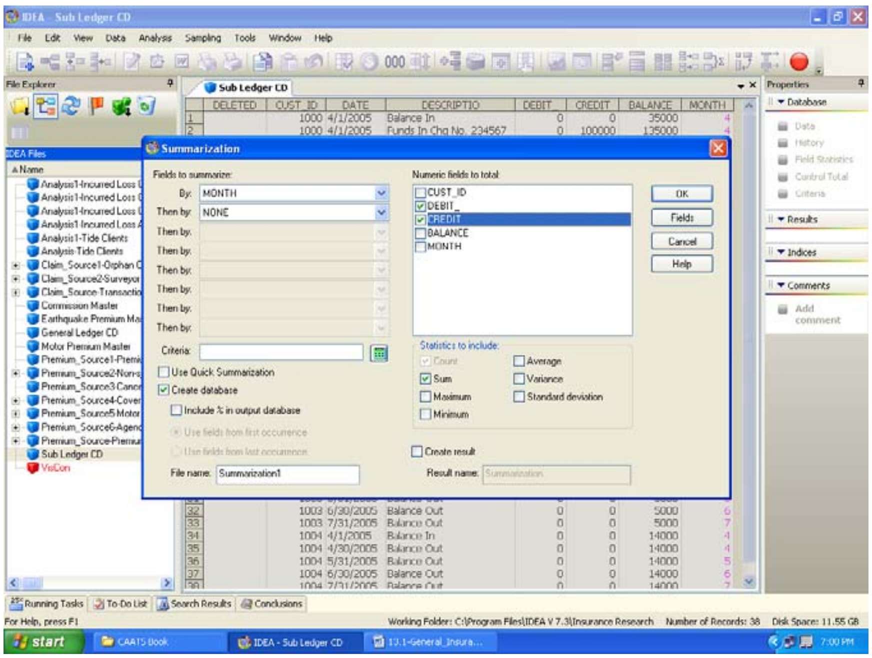
After applying the virtual numeric field we get the following file with month-wise details in the virtual numeric field, as can be seen below.



QUICK FIELD SUMMARIZATION

Perform a quick field summarization on the above file through Analysis > Field Summarization > Quick.

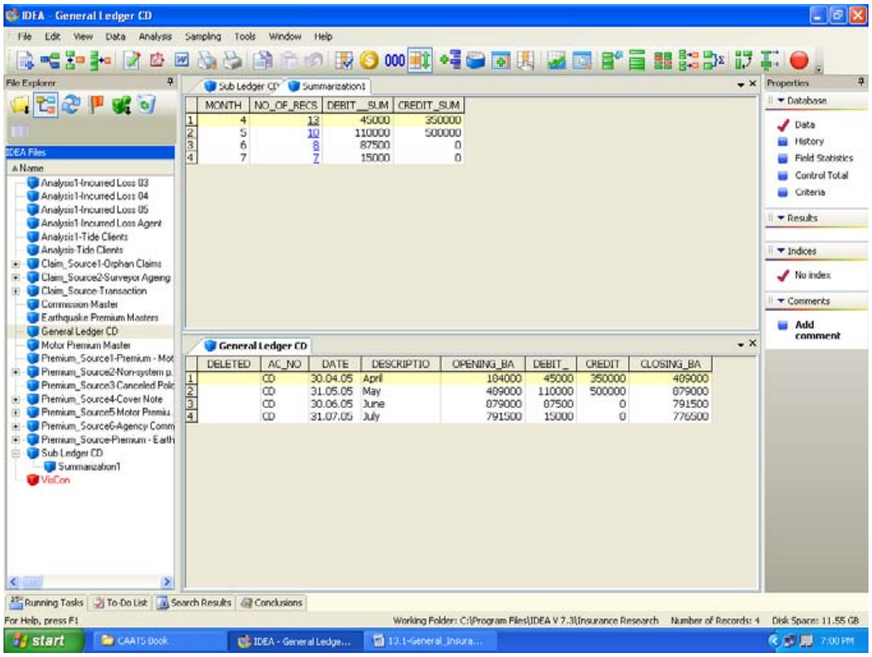
In the Quick Field Summarization dialog box select 'MONTH' as the Summarization Field and 'DEBIT', 'CREDIT' as the Fields to summarize. The process can be seen below



After the quick field summarization, we get the result Quick Field Summarization. IMD as can be seen below.

Simultaneously open the file General Ledger CD. IMD, and tile the two files together through Windows in the menu tool bar.

The comparison of the DEBIT and CREDIT fields from the two files shows that the values match uniformly.



CONCLUSION

This is representative of the arithmetical accuracy of the sub-ledger postings into the Control accounts.

2.0 IDENTIFICATION OF UN ADJUSTED CREDITS IN CASH DEPOSIT ACCOUNT

OBJECTIVE

To identify unadjusted credits in the Cash Deposit Account from the Sub Ledger.

EXERCISE DESCRIPTION

In this exercise, we will create a pivot table by capturing the Customer ID in the row section, month in the column and balance field in the data grid.

Thereafter we will be in a position to identify unadjusted credits based on the pivot table result.

IDEA FUNCTIONALITY COVERED

This exercise will show us how to

- Create a Pivot Table

Use the file Sub Ledger CD. IMD created in the previous exercise.

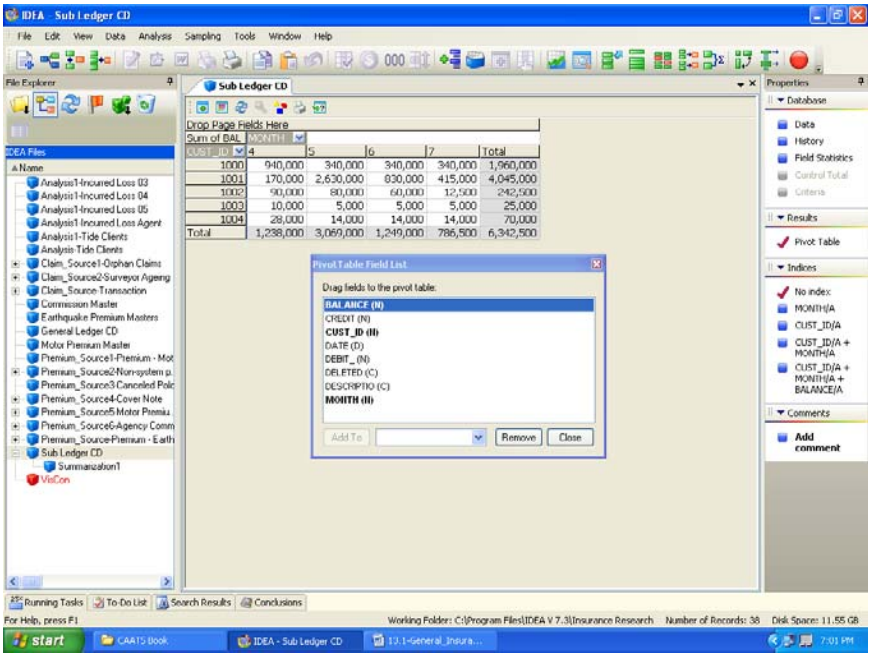
BACKGROUND

Cash deposits are advance sums of money placed by corporate insured with insurance companies, wherefrom regular policy premiums are appropriated towards running policies.

CREATE A PIVOT TABLE

Create a Pivot table on this file.

Capture the CUST_ID in the row section, the MONTH in the column section and the field BALANCE in the data grid.



A closer examination of CUST_ID 1003 and 1004 reveals that the balance of Rs. 5000 and Rs. 14000 respectively has been lying dormant and in credit since May 2005 (Ageing over 3 Months).

CONCLUSION

These cases reflect old outstanding advance receipts, which have not been appropriated towards any specific policy.

Reminders should be sent out to the Insured, informing them of the same and seeking their confirmation on the balances held.

13.1.10 Healthcare Frauds

1.0 EXCESSIVE PROCEDURE BILLING OF SAME DIAGNOSIS, SAME PROCEDURES

OBJECTIVE

To identify instances of excessive medical procedure billing for the same diagnosis and same medical procedure.

EXERCISE DESCRIPTION

In this exercise, we will learn to join the Healthcare Claims transaction file with the master file on the basis of the Diagnosis Code.

Append a computed virtual numeric field to arrive at instances where excessive procedural charges have been claimed by the insured, in comparison to the current master charge list.

Extract cases where the difference exceeds 15% (Acceptable variance norm across hospitals)

IDEA FUNCTIONALITY COVERED

This exercise will show us how to:

- Join Files
- Append a Virtual Field
- Use the Equation Editor
- Apply A Direct Data Extraction

DATA IMPORT

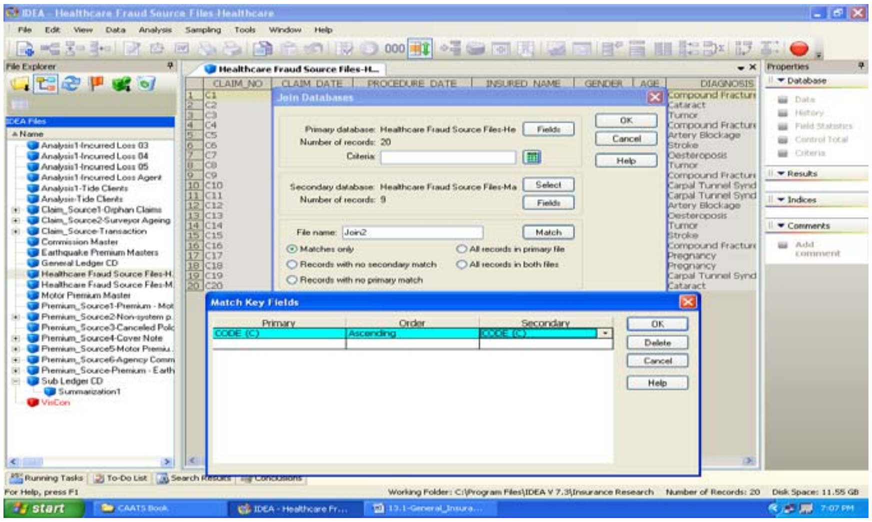
Import the file Healthcare Fraud Source Files. xls

JOIN FILES

Designate Healthcare Fraud Source Files - Healthcare. IMD as the Primary Database and Healthcare Fraud Source Files - Masters. IMD as the Secondary Database.

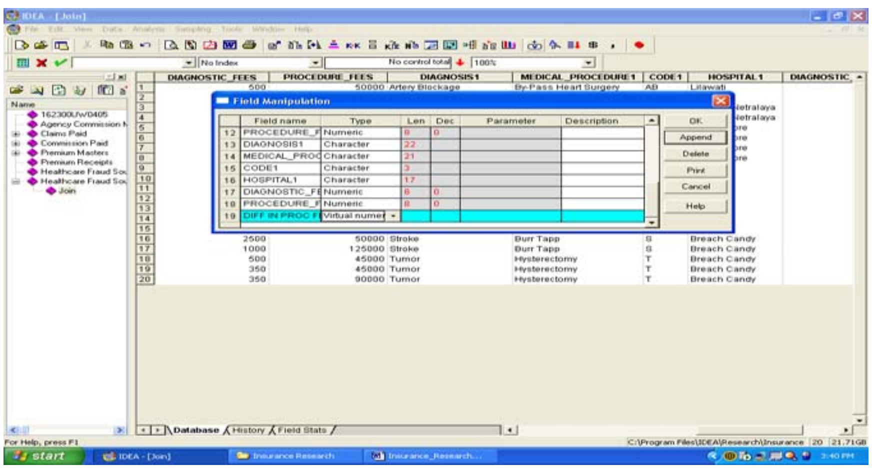
Join the two databases on the basis of ‘Code’ from the Primary Database and ‘Code’ from the Secondary Database. The ‘Code’ represents the Diagnostic Code.

Use the option ‘Matches Only’



APPEND A VIRTUAL FIELD

Append a virtual numeric field by the name ‘Diff in Proc Fees’.



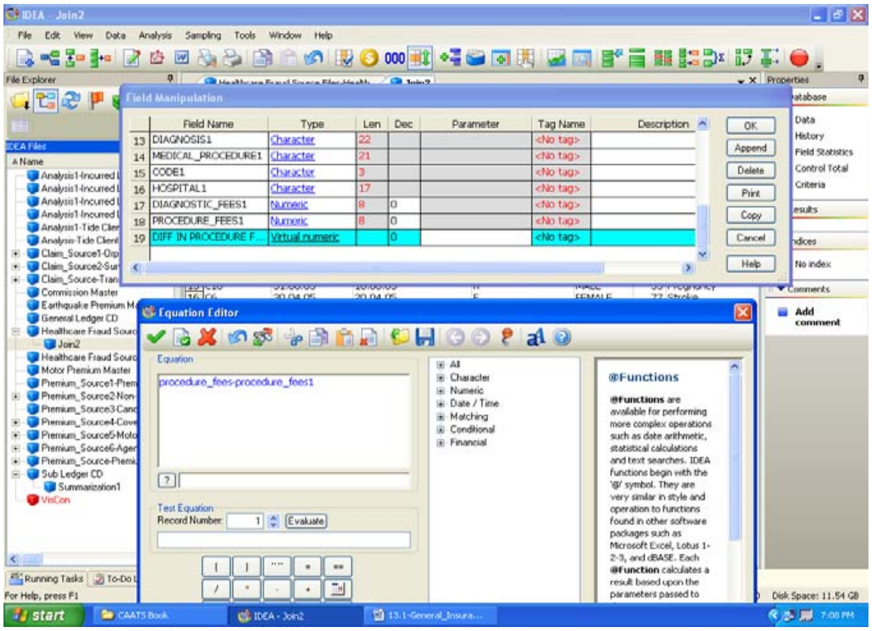
USE THE EQUATION EDITOR

After adding the Virtual Numeric field, click parameter.

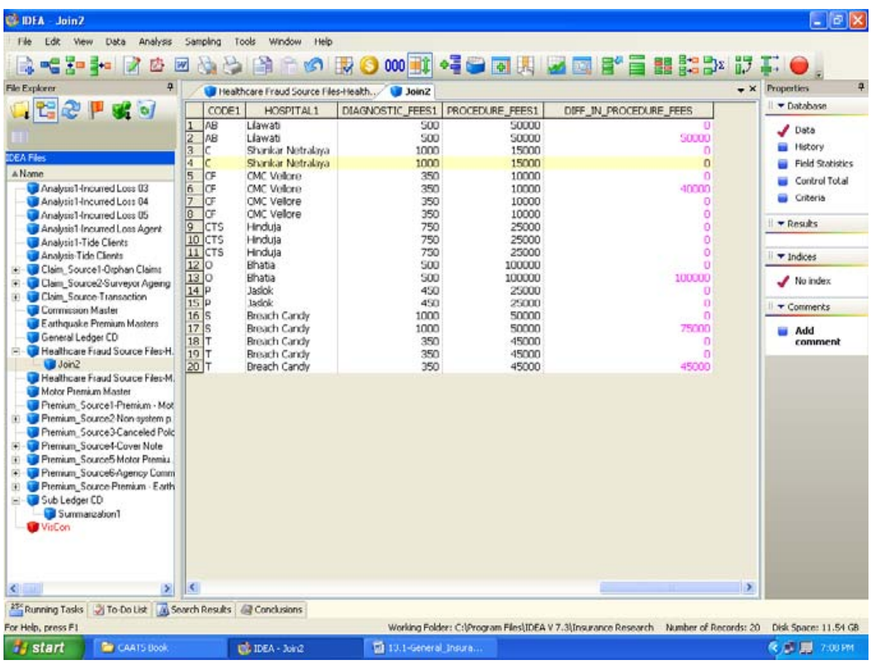
This invokes the Equation Editor.

Enter the equation procedure_fees-procedure_fees1 where procedure_ fees represents the fees claimed and procedure_ fees1 represents the master fees.

Validate and exit the Equation Editor.



The List of exceptions are the non zero values seen in the table below under the field DIFF IN PROCEDURE FEES



APPLYING A DIRECT DATA EXTRACTION

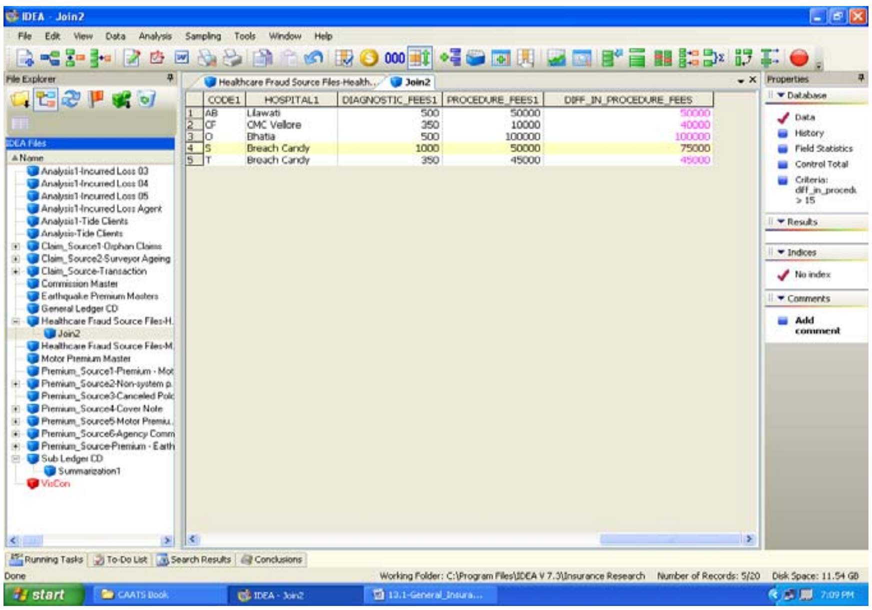
The normal acceptable variance between procedure fees across hospitals is pegged at 15%.

Hence proceed to identify exceptions beyond 15% of procedure_fees claimed.

This is done by writing an equation in Direct Data Extraction.

The extraction gives a child listing of 5 cases, as can be seen below.

These cases should be identified, flagged off and taken up for special scrutiny along with supporting documents, inquiries and investigations.



Other IDEA Tests which can be Run to Identify Healthcare Frauds

- Identify excessive number of procedures per day or place of service per day / per patient
- Identify multiple billings for same procedures, same date of service
- Analyze for mismatched services to diagnosis codes
- Review diagnostic fees in excess of masters
- Review procedure fees in excess of masters
- Identify and report diagnosis and / or treatment that is inconsistent with a patient's age and / or gender

13.2 Application of CAATs to Banks

Banks:Application of CAATs

Introduction

IDEA is useful in most areas of banking, including current accounts, personal and corporate lending, treasury functions and investment accounts. The usual guidelines on volume and depth apply. However, in some banking systems the volume of transactions may be too heavy to download to a PC. It may be appropriate to download a segment of the data or a selected number of fields. Alternatively, a data link may be setup with the Application Server to access raw data in Back-end Tables directly.

IDEA is time-tested, robust and stable. It has been proven globally as an efficient Audit Tool with all the Banking Application Systems:Core Banking Systems, Total Branch Automation Applications, and specialized Banking Domain applications like Applications for Treasury Management and Control.

The following tests cover a variety of account types:

Calculation

- Cast (or total) ledgers.
- Cast (or total) account balances.
- Cast (or total) accrued interest.
- Perform interest calculations.
- Check calculation of bank charges.
- Reprove arrear calculations and reporting (loans).
- Prove interest capitalization

Analysis

- Cast (or total) ledgers.
- Cast (or total) account balances.
- Cast (or total) accrued interest.
- Perform interest calculations.
- Check calculation of bank charges.

- Reprove arrear calculations and reporting (loans).
- Prove interest capitalization

Exception tests

- Identify accounts with missing standing data.
- Identify invalid or unusual standing data.
- Identify dormant accounts and transactions therein
- Identify accounts with statement suppression
- Check charges are being raised where appropriate
- Identify staff loans
- Identify large loans
- Test for unusual interest rates
- Identify loan balances greater than original advance
- Identify negative balances
- Identify forward dated transactions
- Provide totals of forward dated transactions
- Identify overdue maturities
- Identify customers over their overdraft limit or customers with expired limits
- Perform currency conversions
- Provide details of currency exposure

Other useful tests include stratifying interest rates against each risk rating. Generally higher risk loans should bear a higher rate of interest. The analysis followed by exception testing can identify anomalies or branches/individuals that are either risky or particularly prudent in their lending.

Matches and Compares

- Multiple loans to the same address by setting up a key based on the address and using duplicates test
- Match addresses of loans to employee addresses
- Compare balances at different periods to identify movements.

Revenue Assurance for the Bank using IDEA

CASE

Verification of accuracy of Demand Draft charges applied by the Branch Software

OBJECTIVE

To verify the correctness of Demand Draft charges levied.

EXERCISE DESCRIPTION

In this exercise, an equation is written that expresses logical criteria of DD charges applicable to DD amount as per the DD charges handbook issued by the Head Office of the Bank to the Branch for Retail Banking.

IDEA FUNCTIONALITY COVERED

This exercise includes the following steps -

- Append a computed Virtual Numeric Field
- Use the Equation Editor

METHOD

The Equation is composed as follows and can also be saved for future use

@compif((@between(dd_amount, 0,5000),25, @between (dd_amount, 5001, 25000), 40, @between(dd_amount, 25001, 50000), 75, @between(dd_amount, 50001,150000), 150, dd_amount>150001, 250+2.5*(dd_amount-150001)/100))

Multiple Approaches to Data Collation and Interrogation within IDEA for Banks

Approach 1

1. Requesting IT for Data

Case Study 1.1: Electronic KYC Audit

The team selected an objective to test KYC norms on Current account customer master data.

To test this objective the team issued a Data Request to IT in the following format:

- Data required: Current account customer master information.

- Period: As of the date of Audit (period cannot be a date. Rethink).
- Fields of reference: Branch ID, Customer ID, Account ID, First Holder & Joint Holder/s Name, Address, PAN No., Mobile No., Residence No., Office No., Mode of Operation and Clear Balance.
- Format of Data: Text form.

IT in turn ran a SQL query on the production database and generated a text file dump, which was saved by IT in a secure folder with special access to the Audit team only.

The Audit team imported the text file, using the Text Report import option within the GAS.

Post import, the team used the Duplicate Key test within the GAS to identify fictitious accounts opened with similar PAN No., or Mobile No., or Address, or Office No., or Residence No., but different Customer ID.

50 cases out of 65,000 were identified where key account opening fields like PAN No., Mobile No. were same for different Customer ID's. These cases have been taken up for further checking with the account opening forms from the respective Branch to ascertain the validity of the accounts opened.

Additionally, the Team employed a SOUNDEX test on the Customer Name field. The SOUNDEX test generates a unique alpha-numeric code in a new computed field. This code depends on the mnemonic of the Name rather than the spelling. For example, Two Customer Accounts with names Prasanna Iyer and P A Iyer will have the same SOUNDEX code. After arriving at this code, duplicate tests can be run on the code rather than the Name. The advantage here is that the Duplicate on Code will generate similar sounding names (Essence of De-Dup Tests Globally), whereas a pure Duplicate will perform an exact match.

The Team also performed De-Dup Tests to identify Customer's having multiple Savings Accounts within the same Branch of the Bank, which is a potential red flag.

Case Study 1.2: Data Migration Audit

The team then decided to check the integrity of loan data migrated from the Legacy TBA application to the CBS.

To test this objective the team issued a Data Request to IT in the following format:

- Data required: Cash Credit master information for large scale branch X.
- Period: Data immediately post-migration.
- Fields of reference: Customer ID, Sanction Limit, Drawing Power, and Rate of Interest.
- Format of Data:Text form.

IT in turn ran a SQL query on the production database and generated a text file dump which was saved by IT in a secure folder with special access to the Audit team only. The corresponding data from the TBA legacy system (immediately pre-migration? Not clear) was available with the Migration Team. IT sourced the Text data from the Migration Team through a formal email request and placed the data in the secure folder.

The Audit team imported both the text files using the Text Report import option within the GAS.

Post import, the Team linked the pre-migration and post-migration data through the JOIN function in the GAS. The two data files were linked based on the Customer ID available in both the files. Post join, three new fields were created by the Team containing the differences in the Sanction Limit, Drawing Power and Rate of Interest in each field. The Team finally queried each of the fields for non-zero data.

Accounts that showed a difference in the masters migrated (non-zero data) were identified by using the procedure given above. These accounts were taken up for further testing with the legacy system data and the loan appraisal forms to ascertain their accuracy.

Approach 2

2. **Using existing End of the Day (EOD) reports or existing Business Warehouse (BW) solution reports.**

Case Study 2.1: Audit of movement on in-operative accounts

In the second data retrieval approach, the Team decided to audit the movement on inoperative accounts over a period of one financial year.

To test this objective, the Team identified the Report on Inoperative accounts as on 1st April 2014 and 31st March 2015 respectively. These reports were a part of the EOD suite of reports. These reports were generated as a part of the end of the day routine at Branch X and by default saved in a compressed print report format.

The Team uncompressed the reports using WIN-ZIP. Then both the print report files were imported using the Text Report import option within the GAS.

Post import, the Team linked the Report on Inoperative accounts as on 1st April 2014 and 31st March 2015 using the COMPARE function within the GAS. The two data files were linked based on the Customer ID available in both the files. Post compare, a new file was created with differences in the Clear Inoperative balances. The Team finally queried the difference field for non-zero data.

Accounts in which there was a reduction in the inoperative account balances (non-zero data) were identified through the above approach. These accounts were taken up for further testing with EOD Exception reports to ensure that the movement on the inoperative account was authorized by the Branch Manager.

Case Study 2.2: Audit of Loans and Advances MIS from Business Warehouse (BW) for approval verification

In this case, the Team decided to audit the MIS report on Loans and Advances generated from the Business Warehouse (BW) for approval verification. The MIS report is a comprehensive listing of Account ID, Type of Loan, Type of Security, Type of Industry, Sanction Limit, Drawing Power, Rate of Interest, Due Date of Loan, and Approval Officer.

To complete this test, the Team sought assistance from the Merchant Banking wing of the Bank. The Loans officer for Branch X generated the MIS report for the period 1st April 2014 to 31st March 2015. This report was saved as a Microsoft Excel file and provided to the Team on a CD.

The Team imported the Microsoft Excel file using the MS-Excel option for import within the GAS.

Post import, the Team entered a conditional criteria/query on the file, identifying non-compliances with the Sanction Limit-Approval Officer table. The CM, DGM, AGM-HO and AGM-ZO of the Bank were all vested with specific approval Sanction limits. The Team tested the file for cases where loans sanctioned by officers were not in accordance with their financial Sanction Limit powers.

Some cases were identified in which the loans approved did not fall within the Approval Officer’s sanctioning limit. These cases were noted and taken up for review with the respective officer and the Branch Manager.

Approach 3

3. Access to Disaster Recovery (DR) site databases or mirror databases

The Team, with the help of the IT wing, set up connectivity between the GAS and the DR site server. This connectivity was the third and final mode of data retrieval, which gave the Team direct access to raw data tables residing on the DR site server.

Audit Tools like IDEA can work optimally on very large databases. For example, the Team connected to the DR Site Server during peak business hours imported 6 Tables from the Core Banking System Server. The Tables had a cumulative record size of 5 crore records, and the imports were completed within 45-50 minutes cumulatively (Key Performance Metrics). The Teams having specialized skill-sets in SQL Query Building, performed consecutive ODBC links on the 6 Tables. For each link, simple SQL queries (with written?), to slice and dice the data down to exact requirement:. for example, Data for Branch A and Branch B from 2050 Branches. In the Transaction Tables, the Team performed additional SQL Queries for transactions with value dates between 1st April 2014 and 31st August 2014.

The Teams also automated the import links to run as scheduled tasks during non-peak hours. This was done by converting the import tasks into MACROS and MACROS into Executable files. These executable files were programmed to run through IDEA by Windows Tasks Scheduler. The import tasks for the 6 Tables ran automatically at 7 PM in the evening, gave results similar to the peak hour imports with a throughput of 38 minutes vis-à-vis 50 minutes during business hours.

The successful Automation and Scheduling of tasks gave the Team the confidence to deploy Continuous Control Monitoring on a regular basis in the Bank. The Team graduated to running complex imports and queries on weekends, holidays and night-time slots to arrive at real time results at the start of work on the next business working day.

Case Study 3.1: Identifying Term Deposit accounts where the deposit was accepted for a period greater than 120 months.

As per the bank-rules for acceptance of Term Deposits and guidelines of the regulator, the Bank cannot solicit Term Deposits in excess of a period of 120 months. This test was undertaken to identify non-compliances with the rule.

The Team connected to the DR site server using the GAS and identified the data table containing the Term Deposit account masters. At the import stage, the Team selected three fields – Account ID, Account Name and Deposit Period in Months from the table containing 15 different fields. The Team also entered an SQL query in the GAS, filtering the data for Branch X. The import of data of 1 Lac rows/lines was initiated and duly completed within 1 minute. This was faster in comparison to writing an SQL query, which would take 3-4 hours to run in the past.

Post import, the Team queried the data to identify accounts where the Deposit Period in Months was greater than 120.

The result was exported from the GAS back to MS-Excel, printed and taken up for discussion with the Branch Manager and the Head of Term Deposits.

Case Study 3.2: **Isolating loan accounts where the Drawing Power was greater than the Sanction Limit.**

The Team connected to the DR site server by using the GAS, and identified the data table containing the account masters. At the import stage, the Team selected four fields – Account ID, Account Name, Sanction Limit, and Drawing Power from the table containing 35 different fields. The Team also entered an SQL query in the GAS, filtering the data for Branch X.

Post import, the Team applied criteria to the file identifying instances where the Drawing Power was greater than the Sanction Limit.

The result was exported from the GAS back to MS-Excel, printed and taken up for discussion with the Branch Manager and the Head of Loans & Advances.

13.3. Application of CAATs to Retail Sector

Retail Malls – Application of CAATs

Introduction

In the current business environment, the burgeoning growth of Retail Malls/Chains across India and their internal controls need to be controlled and monitored by Auditors for their continued adequacy and effectiveness.

Retail Malls are highly tech-centric with online point of sale billing applications, automatic inventory updation and Electronic Data Interchange re-ordering. There is widespread use of bar coding and bar scanning for billing.

Given the high proliferation of technology, it is best to deploy technology in the form of CAATs to control Retail Technology.

Our case-study looks at Point of Sale Billing Systems, the import of data from such systems and the interrogation of data generated through such systems.

Point of Sale Systems from the Retail Industry – Malls

The Point of Sale Applications deployed in Malls generate comprehensive Sales Reports. These can be conveniently saved in MS-Excel Data formats. This is an alternative to File Print options.

These Sales Reports contain fields like

- Date of Sale
- Time of Sale
- Transaction Number
- Cashier Name
- Cashier ID
- Product Sold
- Quantity
- Rate
- Gross Value
- Taxes
- Scheme Discounts
- Net Value
- Scheme ID
- Scheme Details
- Collections in Cash
- Collections by Card
- Dues

Import into IDEA

These MS-Excel Data Files once saved on the workstation containing IDEA or on the Local Area Network in a shared audit data folder can be accessed through IDEA's Import Assistant, Excel component. The process of import is simple and easy to apply, since Excel file Record Definitions are readily recognized by IDEA.

Interrogation within IDEA

Case Study 1: Reconciliation of Net Sales with Cash Takings, and Card Receipts

The fields of reference relevant to the objective being tested are

- Net Sales
- Cash Collected
- Card Receipts

The process of interrogation followed in IDEA is:

- Import the Sales Report for a given period through IDEA's Import Assistant – MS Excel.
- Navigate to the Field Statistics in the Database Toolbar.
- View the numeric control totals for the Net Sales, Cash Collected and Card Receipts fields respectively.
- Normally the Net Sales should be arithmetically balanced by Cash Collections and Card receipts.
- In the case under review we notice a high percentage of Unpaid Bills, almost 25% of the period's Net Sales.
- An overview of the Unpaid Bill cases through Field Summarization reveals that the due amounts are significantly concentrated on Cashier A and Cashier D.

These cases may be specially looked into by interrogating the concerned Cashier's, to ascertain their motive.

Case Study 2: Inconsistent scheme discount rates offered by Cashier's to different customers against the same Scheme ID

The fields of reference relevant to the objective being tested are

- Cashier ID
- Scheme ID
- Scheme Discounts
- Gross Value

The process of interrogation followed in IDEA is:

- Navigating to Data in the Menu Tool Bar and selecting Field Manipulation
- In Field Manipulation, appending a computed Virtual Numeric Field Discount % with the Criteria (Scheme Discounts*100 / Gross Value), rounded off to the nearest integer.
- Navigating to Analysis in the Menu Tool Bar and selecting Duplicate Key Exclusion
- In Duplicate Key Exclusion, we identify different Discount % values for the same Scheme ID.
- We get a list of cases where varying Discount % have been applied for the same Scheme ID.
- Some cases are extremely glaring, with the Discountbeing as high as 45% though the Scheme ID warrants a Discountof 15% only.

These cases may be specially looked into by interrogating the concerned Cashier, to ascertain his motive.

Case Study 3: Identifying the most efficient Cashier across all the Malls – Operational Throughput

The fields of reference relevant to the objective being tested are

- Cashier ID
- Cashier Name
- Start Time
- End Time
- Quantity

The process of interrogation followed in IDEA is:

- Create a new computed numeric field in the imported Sales File with the difference between the Start Time and the End Time using the criteria '**@agetime(End Time, Start Time)'**. This new field will give us the time taken by each cashier in seconds to scan, bag and bill all the items against that transaction.
- Create another computed numeric field with the criteria '**Difference in Time/ Quantity'** to arrive at the Time taken to scan each item.

- Now perform Field Summarization on the Cashier ID and Cashier Name with regard to the numeric field containing the Time taken to scan each unit.
- In the Field Summarization also include additional statistics like Count and Average along with Min and Max. These statistics will give us the number of scans by a single Cashier in a given period, the minimum time for scan, the maximum time for scan and the average time for scan.
- In the Summarization result, sort the Time to Scan (Average) on a Descending basis.
- The Cashier with the best scan rate appears right on top.

The Management of the Mall Chain can select the Top 10 Cashiers from the above exercise and place them at front desks on crucial days like weekends or public holidays. This test is a concrete measure of customer service.

Case Study 4: Detecting transactions Out of Office hours

The fields of reference relevant to the objective being tested are

- Start Time
- End Time
- Cashier ID
- Cashier Name
- Net Sales

The process of interrogation followed in IDEA f is:

- Perform a Data Extraction on the imported Sales File.
- Build a criterion using the function .NOT. @betweenagetime(Start Time, “10:00:00”, “22:00:00”) .OR. .NOT. @betweenagetime(End Time, “10:00:00”, “22:00:00”)
- This criterion will isolate all transactions out of the normal Mall working hours of 10 AM to 10 PM. Here we trap both Start Time and End Time.
- The Direct Extraction function within IDEA is workable on large databases, say upwards of 1 Crore Transactions. The function first sorts the entire database and then runs the equation through the sorted database. Hence the results arrive faster than running an ordinary command on an unsorted database.

The case reveals around 50 out of 1 Lac transactions where the Start Time and End Time is after office Hours at 10:30 PM. An explanation from the Mall Incharge reveals that all these transactions pertained to Public Holidays, when schemes were launched and working hours were extended by two hours with permission from the local administration.

Case Study 5: Demand Study of New Products introduced into Pilot Mall Outlets across India

Malls introduce new products into Pilot outlets to study the Customer Behavioral patterns, spending patterns, loyalty to existing products rather than new substitutes. .

In this Business Case Study the following fields are considered

- Transaction Number
- Product Number
- Quantity
- Net Sales

The process of interrogation followed in IDEA is:

- The Transaction Number contains a combination of the Mall Outlet ID, the financial year and the transaction ticket number. The first 3 digits of the Transaction number represent the Mall Outlet ID.
- We append a new computed character field with the aim of getting the Mall Outlet ID into this field. This is performed through a criterion /equation @left(Transaction ID, 3). This function removes the first 3 digits from the Transaction Number and places it in a separate computed field.
- Field Summarization is performed on the imported Sales file on Mall Outlet ID and Product Number/Code with respect to Quantity and Net Sales.
- In the Summarization result, a direct extraction is performed on the Pilot Product Malls through the Equation @ list(Mall Outlet ID, “003”, “005”…)
- A Top Records Extraction is performed with the Key field being the Mall Outlet ID, and the top 5 products in each of the 10 pilot malls are identified.

A detailed review of the final result broadly confirms the Management’s expectation. All the new products have fared well in the 10 pilot malls, save one product, which has not been preferred over its existing competitor. This exercise has armed the Management with factual historical data from a truly representative sample of Mall Outlets. Now the Management is in a position to slowly roll out the most liked products to the remaining Mall outlets over the next month.

'Retail Performance Management' enables decision-making for retailers of all sizes and segments by empowering them with comprehensive relevant 'enterprise business intelligence', across technology platforms.

With CAAT, users can jumpstart their analytic journey, and enjoy improved margins, better customer retention, inventory efficiency and promotion effectiveness.

CAATs :

(1) helps accelerate retail organisation's analytic maturity, taking the user one step closer to achieving excellence.

(2) creates benefits by delivering enhanced usability by anticipating the evolving needs of decision-makers and ensuring a faster adoption rate.

(3) through simple screen-guided analytics, empowers every decision-maker in every role in the retail organisation. It takes the load off the IT group by being easily extendable and maintainable.

(4) avoid latency, cost and project management challenges associated with traditional business intelligence methods and enjoy unparalleled speed to benefits.

(5) transforms business intelligence from being a 'decision support system' to a 'decision-making system'.

(6) makes business intelligence pervasive across the retail business by impacting both the top-line and bottom-line performance.

(7) take on a whole new revolutionary role in 'Retail analytics' where the tool is used for continuous monitoring by 'process owners' rather than the erstwhile traditional continuous auditing by internal auditors.

However, the significance of CAAT is greatly accentuated by the understanding of the underlying business process by the 'Process Owner'. 'Retail Analytics' can thus deliver immeasurable business benefits.

Merchandising and assortment :

It's a well-known fact that shoppers prefer to visit places that offer them the maximum options, deals and not to forget a good shopping experience. As a retailer, one does everything to attract and retain customers by providing excellent customer service in terms of variety of products, price and promotions to achieve 'customer satisfaction'.

Here is where 'merchandise and assortment planning' comes in. CAAT provide merchandisers an analytic framework to plan and analyse business activities related to 'merchandise and assortment'. It :

(1) compares historical, planned or forecasted data against actual data to define and optimise merchandise plans

(2) analyses merchandise hierarchy across departments, categories, product lines and 'Stock Keeping Units' — SKUs

(3) analyses the performance of new products

(4) determines customers' seasonal requirements/ preferences.

In short, CAAT provides data of when and what customer needs and all this leads to customer satisfaction which results in customer loyalty.

Loss prevention :

In the retail industry, it is well known that losses due to fraudulent transactions, theft, pilferage, excessive stocking, wastage, shoplifting, internal theft, refunds, exchanges and excessive discounting are inevitable. While one can't do away with these problems, retailers are always on the lookout for ways to minimise losses while keeping costs minimum.

Loss prevention analytics helps the retailer diagnose the root cause of the problem by identify exceptions and taking corrective measures. CAAT substantiates its analyses with historical, geographic, and demographic trends. It can identify :

(1) incorrect refunds which could be fraudulent

(2) spoilage, damage and write-offs

(3) price overrides and improper discounting

(4) supplier or warehousing issues

(5) administrative errors

(6) sales to customers with dubious shopping records — such sales indicate fraud.

(7) erroneous entries for product returns.

Supplier performance :

This is an area that tends to get neglected. However, focusing on this area can help you bring down operation costs because CAAT also enables to forecast optimal levels of inventory, optimise lead time, manage orders, improve fill rates, negotiate trade promotions, manage risks and improve supply chain efficiency — these are just a few challenges faced by retailers when it comes to managing supplier performance.

CAAT provides a decision-making framework that enables the retailer to identify new areas of synergy and avenues for bringing about operational excellence. It enables :

(1) improved inventory management by tracking slow and fast-moving goods, helps avoid 'out-of-stock' situations which lead to avoid loss of sales and profit. In other words, it enables the retailer gain better visibility into stock movement from warehouse to stores.

(2) improve vendor performance

(3) assist in rewarding performing vendors

(4) remove non-performing vendors

(5) negotiate supplier involvement in product trade promotions for better deals

(6) negotiate longer credit periods and shorter delivery cycles

(7) manage supply-related risks and take corrective measures proactively.

Store performance :

CAAT provides retail operation managers a framework to analyse 'store performance' to increase 'store productivity'.

- reclassify stores by local demographics, competitive density, store locations, size and age
- reclassify merchandising categories based on the relationship between the customer, product and store — merchandise assortment required at the store.

- measure contribution and competence of store employees by monitoring their contribution to total sales
- benchmark, compare and rank peer groups based on metrics like yield per square area and average price per item sold.

Customers :

Customer data has long been touted as a key determinant in making better merchandising decisions; however it is an area/asset most retailers have struggled to use to its maximum potential. CAAT provides the retailer with the critical platform he needs to leverage customer loyalty data, sales transaction data, and store data to improve merchandise planning and sales tactics.

CAAT unveils hidden relationships with the customer, because it helps in identifying what the customer buys, how he pays and with what frequency he buys. This enables the manager to :

(1) identify which store needs what and when – thus reducing waste and ensuring optimum use of the shelfspace.

(2) evaluate performance of promotion schemes.

(3) measure campaign effectiveness – what media can be used to achieve better results by having customer feedback.

(4) identify regular — high-value customers who can be suitably rewarded

(5) identify what products customers buy together — for example — floor cleaners are bought alongwith utensil and crockery cleaners and vice-a-versa. In retail trade this is known as ‘product affinity’ or ‘market basket’. This enables the retailer to bunch products in promotional schemes and by placing them close to each other on the store-shelf, thereby increasing sales.

(6) identify seasonal trends – for example – increase in chocolate sales during festive season – what brand and in what ‘pack size’.

(7) identify ‘impulse buying’ generated by a new product or new packing.

(8) weigh loyalty schemes — consider whether the customer should be given more of what he normally buys or offer him/her a new product and/or offer him/her better service — for example — having a dedicated counter.

Conclusion :

CAAT creates an environment where the process owners — the retailer — can make informed and real-time decisions on :

- Which customer segments are the most profitable ?
- Which customers should a campaign target ?
- When and how should he communicate with a customer ?
- Which customers should he spend resources on retaining ?
- Which products should be 'cross-sold' ?

We are at the dawn of mature retail analytics for the discerning retail customer. It is an 'idea whose time has come'.

The use of CAAT by the internal auditor on a sample basis on 'customer analysis' motivates the retailer client to increase greater use of CAAT. The tool introduced by the internal audit led to increased 'customer satisfaction' and increased profits.

13.4 Application of CAATs to Utility Services

Electricity Utilities – Application of CAATs

Introduction

Revenue Audits take on a pivotal role in the audit of Electricity Companies. These companies have specialized Billing applications to record the distribution of power to households and industry. These sophisticated Billing systems are able to generate Billing Sheets and Customer Personal Ledgers under normal and exceptional circumstances like temporary meter disconnection, new consumers not yet metered and the like

These application systems are designed to generate exceptional reports like Zero Value Billing, Customer Overdue, Security Deposit Recoverable, and Ageing of Faulty Meters. .

Majority of these applications run on Oracle as the back-end database due to its stability, performance and security. All the billing related information is stored in a single billing data table.

The user can integrate IDEA with the Oracle database through an ODBC link, and replicate the billing data onto his system. Alternatively, the user can request the IT Cell of the Electricity Company to provide the Billing Data

Table in a flat file data format like (Delimited for instance). These file formats can also be conveniently imported into IDEA.

Bird's Eye View of Functions which can be tested using IDEA for Electricity Companies

1. Recalculation of Revenue Heads

Electricity Companies levy a host of charges to both household and industry customers. These vary from fixed processing cycle charges like Meter Rent to variable charges like Energy Charges. The Company also buys power from state private players to meet the power deficit. Charges levied by the private player to the electricity company are passed on to the final consumer. These Additional Supply Charges or Fuel Escalation Charges vary on an average from month to month.

The objective, description, rationale and authority to raise these charges are stated in a Power Tariff Manual, which is issued in the common interest of operators and customers for transparency. The manual contains the formulae for the calculation of each charge very clearly.

The user can get the manual and identify the charges to be recalculated along with the formulae. Nowadays, because of the Right to Information Act, the Tariff Manuals are also available on the website of the Energy Major.

The user can then create a new computed numeric field (Virtual Numeric Field) within IDEA and replicate the tariff manual formulae in IDEA's Equation Editor.

Illustration – (Units_Consumed * 0.45) - ASC

ASC – Additional Supply Charges stands for purchase of power from private players. The rate of 0.45 per unit as seen in the formulae above is announced by the State Electricity Regulatory Commission on a month-on-month basis.

In this equation we recalculate the ASC and arrive at possible differences between the recalculated ASC and the ASC levied by the billing system.

2. **Verification of Subsidies provided by the State Government to defray the levy of Full Energy Charges to the Final Customer**

In many States in India, the energy charges to be recovered by the Electricity Company are distributed between the final customer and the State itself. Here the State steps in and contributes to the Energy charges by providing Subsidy.

The State Subsidy is invariably a percentage of the Net Billed Amount.

The objective of the test is to ensure that the right amount is being recovered from the State as Subsidy.

The IDEA user can accomplish this task easily by creating a new computed numeric field with the formulae

(Net_Amt * 0.40) – Subsidy

Here we assume that the State supports the Energy Bill up to 40%.

Subsidies are normally provided to households and critical industry units. Hence the IDEA user has to obtain a listing of Customer Code Categories, to know who are entitled to Subsidy and who have to bear the entire Energy Bill.

With this information n the User can make use of the Display or Extraction function in IDEA to identify eligible and ineligible Customer Categories.

3. **Monitoring of Faulty Meters**

Next to power theft, which ranges between 10%-30% in India, faulty meters are the high-risk area for an Energy Major.

Billing Data Files invariably contain a field for Faulty Meters. If the meter is Faulty a flag ‘Y’ appears against the concerned Customer. If the meter is running, the flag ‘N’ appears in the respective field and cell. There is also an additional field available which states the date when the meter became faulty and the average units consumed. The average units consumed is updated on the basis of past usage and history of each user. This field is invariably updated by the Billing clerk manually. In the absence of automatic system generation of average units consumed, this is a ripe area for mismanagement of revenues.

With the help of IDEA, the user can link the faulty meters, the average units consumed and the last consumption prior to the meter going faulty through the Visual Connector.

The user can then create a new computed numeric field where the average units being billed fall short of the last active consumption reading by say 20% (norms can be decided on a case-to-case basis depending on the Customer Class, Geographical Region and the like). These cases can be taken up for review, discussion and scrutiny with the respective Regional Chief Electricity Engineers.

4. Duplicate Tests on Billing Information

The Billing file contains the Customer Number, Meter Number, Bill Number and Receipt Number where the Bill has been duly paid.

As a part of the standard financial and revenue integrity testing, the user can employ the Duplicate Key test within IDEA to look for possible duplicates in the combination of the following fields

Customer Number

Meter Number

Bill Number and

Receipt Number

The existence of duplicates could indicate an irregularity which needs to be investigated further in the Billing Application System.

Chapter 14

Accounts Payable and IDEA

14.1 IDEA and the Accounts Payables Process

14.1.0 Introduction

It is vital to establish that controls in Accounts Payable processes are in place and working. For example, to make sure that the client is procuring the right products, at the right price, at the right time and from the right vendor.

CAATs – IDEA helps us to identify areas where the controls are failing because of actual data errors. These factual cases serve well in building up a strong defense and back-up for the Auditor to express opinions on the processes.

IDEA can be applied to Accounts Payable processes from an Internal Audit, Statutory Audit and Forensic Review perspective. In this chapter, we deal with IDEA applied to Accounts Payable from a Sarbanes Oxley perspective.

Most of the tests within IDEA can be automated and scheduled, which is the heart of Continuous Control Monitoring and Pre-Audit of Accounts Payable.

14.1.1 Potential Risks

The following table identifies the key risks, explains the business and audit implications of each risk and the audit objectives that can be realized by audit tests.

S No	Risk	Implications	Audit Objectives
1	Payments are made to unauthorized suppliers.	Unauthorized suppliers could represent former suppliers that supplied goods or services of unacceptable quality and should have been removed from the list of suppliers; or they could be fictitious suppliers set up by dishonest personnel to	Existence, Validity

S No	Risk	Implications	Audit Objectives
		receive automated payments. Payments made to unauthorized suppliers could therefore represent either error or fraud.	
2	Payments are made to individuals or employees.	Payments made to individuals or employees could represent a diversion of company payments, indicating fraud.	Existence, Validity
3	Unauthorized premiums are given to suppliers.	Unauthorized premiums may represent overpayments to suppliers in return for kickbacks.	Existence, Validity
4	Invoices are paid late.	Delays in processing Accounts Payable approvals can result in a loss of available discounts for timely remittances and understatement of liabilities in a particular period.	Cut-off, Completeness
5	Invoices are paid as per irregular or inconsistent credit period schedules	Irregular payments may reflect processing errors or fraud.	Existence, Validity
6	Invoices are processed twice.	Duplicate payments can result from the failure to cancel documents to prevent re-use or processing errors in Accounts Payable such as restoring a backup file twice.	Existence, Validity
7	Payments are made in a way	Perpetrators of fraud may arrange payments to avoid	Existence, Validity

S No	Risk	Implications	Audit Objectives
	that these are not detected by audits.	detection. For example, large amounts may be split into several smaller payments to coincide with a perpetrator's transaction approval limits or to avoid limit checks on large payments.	
8	Items (e.g., P.O.s, checks) are missing.	Since Accounts Payable are often not authorized for payment until there is a three-way match of purchase order, receiving document and supplier invoice, missing documents could result in Accounts Payable being understated.	Completeness

14.1.2 Potential Tests

The following audit tests are suggested for auditing an Accounts Payable system. However, the exact tests to be carried out for a particular client will depend upon the system used and the data available. Common tests include:

Mechanical Accuracy and Valuation

1. Total the file. It often pays to separate debits and credits.
2. Revalue foreign payables, if applicable.
3. Check transaction totals to the balance on each account.

Analysis

1. Stratify the size of payments and extract any exceptionally high payments.
2. Analyze payment days and identify suppliers with favorable payment terms.
3. If the computer system captures the approving authority for a transaction, examine the value distribution for each manager.

Exception Tests: Existence and Validity

1. Identify payments made to unauthorized suppliers by matching the payments and authorized suppliers list.
2. Search payments file for payees without “Inc”, "Ltd”, or "Co" in their name to identify payments to individuals.
3. Test for large discounts.
4. Test for duplicated invoices using value and supplier code as the key fields for one test and purchase order number for another. The second processing of invoices can be used to establish a value on the P/L to make a fraudulent payment. This will also pick up accidental duplication.
5. Identify payments made on Sundays or other days/dates that are not valid.
6. To find if amounts are being approved at or just below break points in authority level by a value distribution across the whole ledger. If approval authority is not directly available, perform subsidiary analysis by types of supplier or approving department (i.e. marketing).
7. Look for split invoices to enable approval to be kept by an individual. Extract all invoices within 90% of an approved limit (preferably for a suspected manager or department) and search for all invoices from that supplier. Sort by approving manager, department and date to identify possible split invoices or summarize payments by invoice number to determine how many partial payments have been made for each invoice.
8. Test for sudden increase or decrease in supplier payments between two time periods like two years.
9. Test for large one-off payments to suppliers.
10. Using the first five or six characters of the name, match supplier names against a list of employee surnames from a payroll or personnel file.
11. Test for suspected duplicate supplier names.
12. Test for incomplete or unusual supplier details.

Gaps and Duplicates

Test for missing items or gaps in the check number sequence.

14.1.3 Accounts Payable Fraud Tests

Employees as Vendors

An employee sets up his or her own company and then funnels purchases to that company. Variations include a "ghost" approach where invoices are sent from the employee's company but no actual goods or services are provided. In other instances, actual goods may be shipped.

To detect this type of fraud, ask operational managers to review new vendors as they come aboard. A quick phone call to the vendor may reveal suspicious activity. Make sure, however, that the person doing the review is not the perpetrator. With IDEA, we can use sampling techniques to generate a list of vendors to verify.

Favorable Treatment of Vendors

Look closely at vendors who have a relationship with an employee – a spouse, friend, social partner/ buddy, etc.

Pivot tables in IDEA are effective in detecting this type of fraud.

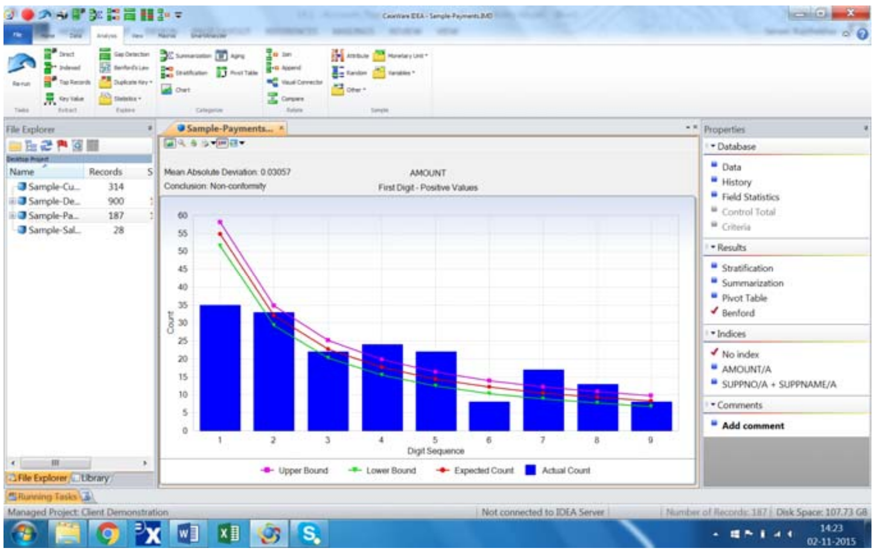
Once created, the table shows the percentage of purchases by each vendor. Always work with an operational expert to interpret the results. It may be normal, that 60% of purchases are from a single vendor, if there are only two or three vendors who can provide the item.

Transactions at or Near Spending Authorities

Spending limits are often referred to as the "trigger price". As soon as a pre-set limit is reached, the system automatically triggers some form of action or examination.

The perpetrator's goal is simply to avoid the threshold approval price. For example, it may be known to almost everybody that the spending authority limit is Rs. 50,000. The fraud is designed to stay under the threshold approval price.

This can be detected by employing Benford's Law of Digital Analysis, which detects instances where prices are just below the threshold approval price. The Benford Curve will reveal sharp, striking indications of actual frequency occurrences in excess of expected and upper tolerance frequency occurrences.



Another common scenario involves paying for large purchases over several invoices. To close a sale, the vendor may accommodate this request. Look for transactions with the same vendor that are close to the threshold approval price. Comparing inventory records is also an effective way to detect this. If three invoices are used to count for the full amount, an inventory check will reveal only one set of goods.

14.1.4 Accounts Payable – Sarbanes Oxley Testing

Sarbanes-Oxley Section	Requirement	IDEA Test
Section 302: Corporate Responsibility for Financial Reports	Review of Financial Transactions of Significant Accounts	Use IDEA to summarize all detailed financial transactions for Accounts Payable and agree to General Ledger
	Financial Transaction Analysis	Use IDEA to re-age Accounts Payable, use the “Field Statistics” feature to identify unusual financial

Sarbanes-Oxley Section	Requirement	IDEA Test
		transactions i.e. outliers.
Section 404: Management Assessment of Internal Controls	Matching Controls	Use IDEA in Accounts Payable to verify the matching of Purchase Orders, Invoices and Receivers.
	Cut - off Controls	Use IDEA to test Expense booking by extracting all Creditors for Expenses transactions before and after cut-off dates to verify inclusions and exclusion in General Ledger for the accounting period.
	Limit Controls	Use IDEA to extract all checks (vendor payments, paychecks, commission checks, bonuses, employee reimbursements) above threshold limits for dual approval authorizations.
Section 806: Whistleblower Protection	Protection of Employees who provide evidence of fraud.	Use IDEA to select a sample of phone calls and incidents from Hotline database and test the effectiveness of the Anti-Fraud Hotline program.

14.2 Accounts Receivable and IDEA

IDEA and the Accounts Receivable Process

14.2.0 Introduction

It is vital to establish that controls in Accounts Receivable processes are in place and working as intended. For example, to find out –

- Whether the client is monitoring the overdue Receivables over 180 Days?
- Whether sales are being affected in excess of the Customer's Credit Limit?
- Whether excessive credit notes are being offered to specific customers towards returns, rebates, incentives.

CAATs –IDEA provide us instances to identify where controls are failing because of actual data errors. These factual cases serve well in building up a strong defense and back-up for the Auditor to expresso pinions on the processes.

IDEA can be applied to Accounts Receivable processes from an Internal Audit, Statutory Audit and Forensic Review perspective. This chapter shows how IDEA can be applied to Accounts Receivable from a Sarbanes Oxley perspective.

Most of the tests within IDEA can be automated and scheduled, which is the heart of Continuous Control Monitoring and Pre-Audit of Accounts Receivables and Collections Monitoring.

14.2.1 Potential Risks

The following table identifies the key risks, explains the business and audit implication of each risk and the audit objectives that could be realized by audit tests.

S No	Risk	Implications	Audit Objectives
1	The file is incorrectly consolidated or	Items could be omitted, or the listed items may not be included in the totals reported	Completeness, Accuracy

S No	Risk	Implications	Audit Objectives
	summed.	in the financial statements. The Accounts Receivable could be overstated or understated depending on the direction of error.	
2	Foreign currency transactions are not translated correctly.	Management may not be aware of the impact of transactions in foreign currency and may fail to take steps to manage currency risks. The Accounts Receivable could be overstated or understated depending on the direction of error.	Accuracy
3	Credit is granted to customers that are likely to default.	The business will sell goods to parties from which they will not be able to collect cash. This has potential implications for liquidity and bad debt expenses.	Valuation
4	Customers are double billed.	Double billing can negatively affect customer satisfaction. Also, revenues and receivables would be overstated.	Existence, Validity, Valuation
5	Accounts Receivable are invalid or incorrectly stated.	Accounts could be entirely or partly invalid. Partially invalid accounts may be the result of delays in processing transactions or errors in applying credits and payments to accounts. Fictitious accounts could be due to fraud.	Existence, Validity, Accuracy
6	Improper allocation of credits and payments	As mentioned earlier, improper allocation of payments to accounts could affect the aging of the Accounts Receivable and this would affect management's ability to determine an effective	Existence, Validity, Accuracy

S No	Risk	Implications	Audit Objectives
		course of action for handling the customer's account. For example, sales to customers could be blocked or the customer may be sent to a collection agency; (even though the customer is current on their accounts Does not make sense. Improper allocation of payments to accounts could also be an indicator of fraud.	
7	Accounts Receivable is not properly aged.	If the aging of the Accounts Receivable is not correct, then management may fail to take action on overdue accounts in a timely manner and expose sales to poor credit risks. Also, the calculation of the allowance for doubtful accounts and bad debts expense would be affected.	Valuation
8	A significant percentage of the receiva-bles is concentrated in a few customers.	The business could be exposed to a combination of credit and liquidity risks if these large customers do not pay their debts in time. Also, the company may be deemed to be economically dependent on the identified customers, and this may need to be noted in the financial statements.	Presentation
9	Improper classification of amounts.	If a credit balance is classified as an AR instead of an AP, then it could distort the current ratio which could be part of a debt covenant.	Presentation

14.2.2 Potential Tests

The following audit tests are suggested for auditing an Accounts Receivable system. However, the exact tests to be carried out for a particular client will depend upon the system used and the data available. Common tests include:

Mechanical Accuracy and Valuation

1. Total the file. It often pays to separate debits and credits.
2. Revalue foreign debts, if applicable.
3. Check transaction totals to the balance on each account.

Analysis

1. Profile debtors using **Stratification** to see the number of large debts and what proportion of value is in the larger items.
2. Produce an aged debt analysis. Consider how to deal with unallocated cash and credit notes. IDEA, by default, ages these on their date rather than allocating against the oldest item or any other treatment. It is often worthwhile to split the file into invoices, unallocated cash, and so on using multiple extractions, and then to age the individual files.

Exception Tests:- Existence and Valuation

1. Identify old items (i.e., more than three months old).
2. Identify large balances either in their own right or compared to turnover.
3. Select accounts for which no movements have been recorded in a set time.
4. Report credit balances.
5. Identify unmatched cash or credits.
6. Compare balances with credit limits and report exceptions (i.e., accounts with balances in excess of their credit limits or accounts with no credit limits etc.).
7. Test for items with invoice dates or numbers outside the expected range.
8. Identify partial payments of debts.
9. Identify invalid transaction types.

10. Identify customer addresses that are “care of” or flagged not to be sent out.

Gaps and Duplicates

1. Test for duplicate invoices (both invoice number and customer/value).
2. Use duplicate exception testing for less obvious input errors, such as the same customer ID assigned to two different customer names, the same customer name assigned to two different customer IDs, receipt against the same invoice number and amount to two different customer, and so on.

Matching and Comparison Tests

1. Compare the balance on an account with its turnover.
2. Match the sales transactions to the customer master information to identify sales to new or unauthorized customers and those with exceeded credit limits.
3. Compare Accounts Receivable to Accounts Payable for possible contra accounts.

Sampling

Select samples (random or specific) for functional testing and confirmation (and produce confirmation letters).

14.2.3 Accounts Receivable Fraud Tests

Teeming and Lading

In Teeming and Lading, (especially in Cash Collection centric Business or Dunning Centres), the first lot of collections are defalcated by the perpetrator. To avoid clear obvious notice of overdue, cash collections from the subsequent customer are applied to the first customer. In this systematic scheme of events, old invoices are always shown/made up as clear/settled/paid and new invoices for latest customers are always shown as overdue, even though they have been paid.

In such cases, IDEA can be used to extract current bills for current sales and current customers based on Stratified Random Sampling (Materiality of the Sale). These customers can be called by the Auditor (telephonically) to ascertain the balance in his books for Debtor Confirmation. Any foul-play can be identified at once through such a scheme of events unless there is a systematic collusion between the Vendor and Customer.

Forced Fictitious Sales to meet Sales Targets

We all have encountered enormous flurry of activity in the Sales Cell of an Entity towards each month end, quarter end, and year end. It is a known fact that Sales are pushed over phone to secure orders. These book orders are eventually accompanied by stock movements within a few days, hence the order is accrued at the period end and the targets are met.

However, we need to take care to identify unusual sales patterns at period ends which are counter-balanced by reversals in ensuing period beginnings. These unusual trends can be identified through the Field Statistics – Date Statistics in India. High number of sales transactions / records which almost resemble one another between say March 2015 and April 2015 need to be investigated, especially when the transactions in April 2015 pertain to reversals of March 2015 sales.

Transactions at or Near Selling Limits

Selling limits are often referred to as the "trigger price". As soon as a pre-set limit is reached, the system automatically triggers some form of action or examination.

The perpetrator's goal is simply to avoid the threshold approval price – trigger price. For example, it may be known to almost everybody that the spending authority limit is Rs. 50,000. The fraud is designed to stay under the threshold approval price.

This can be detected by employing Benford's Law of Digital Analysis, which detects instances where the sale price is just below the trigger price. The Benford Curve will reveal sharp, striking indications of actual frequency occurrences in excess of expected and upper tolerance frequency occurrences.

Another common scenario involves receipts for large sales over several invoices. To close a sale, the customer may accommodate this request. Look for transactions with the same customer that are close to the trigger price. Comparing inventory records is also an effective way to detect this. If three invoices are used to account for the full amount, an inventory check will reveal only one set of goods.

14.2.4 Accounts Receivable: Sarbanes Oxley Testing

Sarbanes-Oxley Section	Requirement	IDEA Test
Section 302: Corporate Responsibility for Financial Reports	Review of Financial Transactions of Significant Accounts	Use IDEA to summarize all detailed financial transactions for Accounts Receivable and agree to General Ledger
	Financial Transaction Analysis	Use IDEA to re-age Accounts Receivable, use the "Field Statistics" feature to identify unusual financial transactions.
Section 404: Management Assessment of Internal Controls	Matching Controls	Use IDEA in Accounts Receivable to verify the matching of Sales Invoices / Orders and Stock Movements.
	Cutoff Controls	Use IDEA to test Sales Booking by extracting all Debtors created before and after cutoff dates to verify inclusions and exclusion in General Ledger for the accounting period.
	Limit Controls	Use IDEA to extract all receipts (sales, customer payments, incentives, Selling Agent commission checks, Annual Target Bonuses) above threshold limits for dual approval

Sarbanes-Oxley Section	Requirement	IDEA Test
		authorizations.
Section 806: Whistleblower Protection	Protection of Employees who provide evidence of fraud.	Use IDEA to select a sample of phone calls and incidents from Hotline database and test the effectiveness of the Anti-Fraud Hotline program.

14.3 Inventory and IDEA

IDEA and the Inventory Process

14.3.0 Introduction

It is vital to establish that controls in Inventory processes are in place and working. To see, for example, whether the client is stocking material in excess of the maximum stocking limit, whether obsolete inventory is being monitored and provisioned, and whether inventory is being valued at the correct price.

CAATs IDEA identify instances where controls are failing because of actual data errors. . These factual cases serve well to build a strong defense and back-up for the auditor to express opinions on the processes.

IDEA can be applied to Inventory processes from an Internal Audit, Statutory Audit and Forensic Review perspective. In this chapter we show how IDEA can be applied to Accounts Payable from a Sarbanes Oxley perspective.

Most of the tests within IDEA can be automated and scheduled, which is the heart of Continuous Control Monitoring and Pre-Audit of Inventory.

14.3.1 Potential Risks

The following table identifies the key risks, explains the business and audit implication of each risk and the audit objectives that could be addressed by audit tests.

S No	Risk	Implications	Audit Objectives
1	Inventory is not correctly recorded.	Management will not have accurate information to manage inventory (e.g., ordering) effectively. Inventory could be materially misstated on the financial statements	Accuracy, Existence, Validity
2	Inventory management reports have inadequate supporting information.	Management may identify issues, but may not be able to "drill down" and identify the root cause of issues. This, in turn, hampers management's ability to make effective decisions.	Existence, Validity, Valuation
3	Quantity of inventory is not maintained within the specified range.	If too much inventory is maintained, then there is an increased risk of obsolescence and additional storage costs (i.e., for keeping the inventory). If too little inventory is maintained, then the company will not be able to meet the customer demand.	Valuation
4	Obsolete inventory items are not identified.	Management will not be able to make proper decisions on getting rid of obsolete items. From an audit point of view, obsolete items may need to be re-valued if the market price is less than their cost.	Valuation
5	Differences between the physical inventory and the inventory on the system are	Management will not have reliable information to make inventory management decisions. Also, it may have a problem identifying shrinkages. From an audit perspective, the	Existence, Validity

S No	Risk	Implications	Audit Objectives
	not identified.	inventory could be materially misstated on the financial statements.	
6	Items are not recorded in the correct period.	Financial statements, i.e., inventory and cost of goods sold, could be materially misstated.	Cut-off
7	Gaps in sequentially numbered documents are not accounted for.	Inventory tags, transaction numbers and other information may be missing and not accounted for. This would undermine the control that is provided by inventory tags.	Completeness

14.3.2 Potential Tests

The following tests are to be undertaken for analyzing an inventory system. However, the exact tests carried out for a particular client will depend upon the system used and the data available.

Common tests include:

Mechanical Accuracy and Valuation

1. Total the file, providing sub-totals of the categories of inventory.
2. Re-perform any calculations involved in arriving at the final stock quantities and values.
3. Re-perform material and labor cost calculations on assembled items.

Analysis

1. Age inventory by date of receipt.
2. Compute the number of months each inventory item is held, based on either sales or purchases. Produce a summary of this information.
3. Stratify balances by value bands.
4. Analyze gross profit.
5. Analyze price adjustment transactions.

Exception Tests: Existence and Valuation

1. Identify and total inventory held in excess of maximum and minimum inventory levels.
2. Identify and total obsolete or damaged inventory (identified as such in the database).
3. Identify balances in excess of a reasonable usage period that are probably obsolete.
4. Identify items past their shelf life (if a sell by date or bought date is present on the system).
5. Identify any items with excessive or negligible selling or cost prices.
6. Identify differences arising from physical stock counts.
7. Test for movements with dates or reference numbers not in the correct period (cut-off).
8. Identify balances that include unusual items (i.e., manual adjustments).
9. Identify work in progress that has been open for an unreasonable period.
10. Identify inventory acquired from group companies.

Gaps and Duplicates

1. Test for missing inventory ticket numbers.
2. Test for missing transaction numbers.
3. Identify duplicate inventory items.

Matching and Comparison Tests

1. Compare files at two dates to identify new or deleted inventory lines or significant fluctuations in cost or selling price.
2. Compare cost and selling price and identify items where cost exceeds net realizable value.
3. Compare holdings and inventory turnover per product between stores.

14.3.3 Inventory Fraud Tests

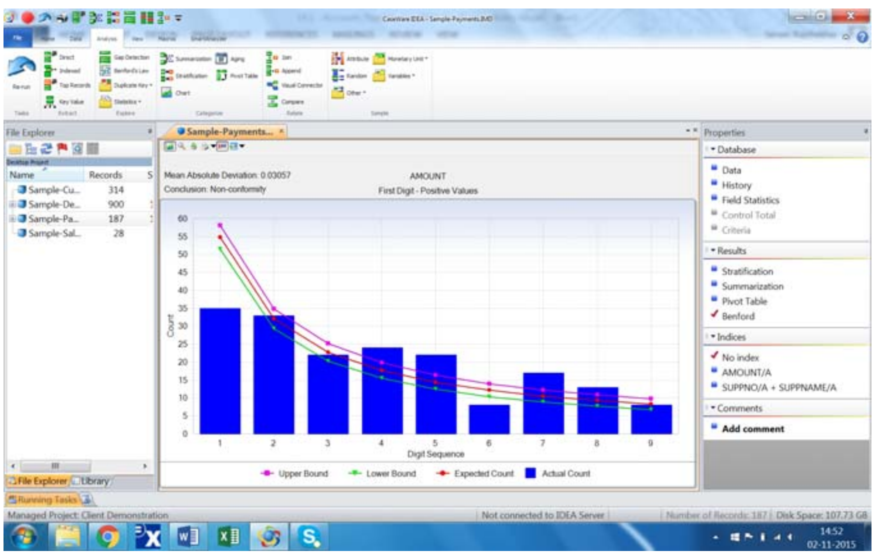
Fraudulent inventory valuations

In certain industries, Finished Goods inventory can be divided into two main classes: For Sale and For Samples / Gifts / Testing.

The inventory held for sale is correctly valued at net realizable value. Fraudsters often inflate inventory valuations for inventory held for Samples / Gifts / Testing.

As a common practice inventory held for Samples / Gifts / Testing, are valued at minimum system recognizable value like INR 1.00 to extend a notional value to the inventory in the accounting system.

The antithesis of the Benford Law may be applied here to test unexpected frequency counts at Digits other than 1 for inventory held for Samples / Gifts / Testing. If the entity is deliberately overstating, such inventory will show up on the Benford Curve for further examination and interrogation.



14.3.4 Inventory: Sarbanes Oxley Testing

Sarbanes-Oxley Section	Requirement	IDEA Test
Section 302: Corporate Responsibility for	Review of Financial Transactions of Significant Accounts	Use IDEA to summarize all detailed financial transactions for Inventory

Financial Reports		and agree to General Ledger
	Financial Transaction Analysis	Use IDEA to re-age Inventory, use the "Field Statistics" feature to identify and drill-down to unusual financial transactions.
Section 404: Management Assessment of Internal Controls	Matching Controls	Use IDEA in Inventory to verify the matching of Purchase Orders, Invoices and Receivers.
	Cut off Controls	Use IDEA to test inventory booking by extracting all receipt transactions before and after cut off dates to verify inclusions and exclusion in General Ledger for the accounting period.
	Limit Controls	Use IDEA to extract all inventory holdings above inventory control limits for dual approval authorizations.
Section 806: Whistleblower Protection	Protection of Employees who provide evidence of Fraud.	Use IDEA to select a sample of phone calls and incidents from Hotline database to ascertain the effectiveness of the Anti-Fraud Hotline program.

14.4. Payroll and IDEA

IDEA and the Payroll Process

14.4.0 Introduction

It is vital to establish that controls in Payroll processes are in place and working. For example, to find whether an employee is entitled to a specific location allowance, whether the head-count analysis for a month-on-month basis matches, and whether different employees have the same bank account number for salary credit.

CAATs –IDEA help us identify instances where controls are failing because of data errors. These factual cases serve well in building a strong defense and back-up for the auditor to express opinions on the processes.

IDEA can be applied to Payroll processes from an Internal Audit, and Forensic Review perspective. In this chapter we show how IDEA can be applied to Payroll from a Sarbanes Oxley perspective.

Most of the tests within IDEA can be automated and scheduled, which is the heart of Continuous Control Monitoring and Pre-Audit of Payroll. Cyclical audits for the same division payroll or for the same company across multiple locations can be automated.

Payroll tests essentially focus on ELIGIBILITY, ACCURACY, ADMISSIBILITY and CONSISTENCY of payment of allowances and credit of deductions.

14.4.1 Payroll – Control Risk Perspective

Payroll auditing is an excellent application of IDEA. The main objective is validity and accuracy by testing the existence of employees and their correct pay. There are many regulations and taxes associated with payroll and compliance with these can be checked. Privacy concerns may limit our testing though.

Analysis

- Summarize and stratify salaries by department/grade, etc
- Profile employee ages/years of service to assist in forward planning
- Analyze costs for special pay, overtime, premiums, etc

- Summarize payroll distribution for reconciliation to general ledger
- Summarize and compare costs for special pay, overtime, premium, etc

Calculations

- Total gross pay, net pay, deductions and any other value fields
- Check calculation of gross pay
- Check calculation of net pay

Exception Tests

- Extract all payroll checks where the gross amount exceeds a stipulated amount

Reasonableness of:

- Tax rates
- Pay/grade comparison
- Hours worked
- Overtime claimed
- Leave consumed
- Date of birth (under 18, over 60 years of age)
- Identify bonuses and other allowances
- Report activity on records for new or terminated employees
- Find changes in key payroll data, such as gross pay, hourly rates, salary amounts, exemptions, etc
- Identify records with missing information (National Insurance number/Social Security number, tax code, employee number etc)

Gaps and Duplicates

- Duplicate employees (Social Insurance, National Insurance, Social Security numbers, Employee numbers, addresses) on payroll file
- Duplicate bank account details
- Duplicate names and dates of birth

Matching and Comparing

- Comparison of payroll file at two dates to determine recorded starters and leavers, (hires and terminations) and changes in pay, etc, are as expected
- Join attendance file, to payroll master to determine if there are "ghost" employees on the payroll
- Compare time-card entries and pay to payroll and indicate variances
- Compare vendor addresses/phone numbers and employee addresses/phone numbers to identify conflict of interests (e.g. postcodes, phone numbers)

Sampling

- Most sampling options apply

14.4.2 Payroll Fraud Tests

A representation of typical fraud tests with regard to Payroll is as follows:

- Employees having the same First Name and Last Name
- Employees having the same First Name, Last Name and Bank Account Number.
- Employees having a different First Name, Last Name but having the same Bank Account for Salary Credit
- Employees having similar sounding Names (De-Dup Tests using Soundex Functions)
- Payments to Employees after they have left the Entity
- Payments to Employees who are not on the Employees Master List
- Overtime payments to Employees when normal hours have been worked.
- Payments of Location Allowances to Employees who are not entitled to them.
- Payment of Grade Allowances to Employees whose Grade does not permit it .

- Payment of both Asset Maintenance Expenses and Asset Maintenance Fixed Allowance like Vehicle Allowances when the Entity Policy allows only one.

14.4.3 Payroll : Sarbanes Oxley Testing

Sarbanes-Oxley Section	Requirement	IDEA Test
Section 302: Corporate Responsibility for Financial Reports	Review of Financial Transactions of Significant Accounts	Use IDEA to summarize all detailed financial transactions for Payroll Heads of Allowances and Deductions from the Pay Ledgers and agree to General Ledger
	Financial Transaction Analysis	Use IDEA to analyse Pay Ledgers by Age, Salary, and Demographics for cases of attrition. Use the "Field Statistics" feature to identify and drill-down to unusual payroll indicators.
Section 404: Management Assessment of Internal Controls	Matching Controls	Use IDEA in Payroll to match Pay Ledger with Payroll masters to test for unauthorised payments.
	Cutoff Controls	Use IDEA to test for unentitled arrear payments or leave without pay returns to employees out of due periods for payment.
	Limit Controls	Use IDEA to test for Net Pay out of Salary Ranges in line with Employee Grades.

Sarbanes-Oxley Section	Requirement	IDEA Test
Section 806: Whistleblower Protection	Protection of Employees who provide evidence of Fraud.	Use IDEA to select a sample of phone calls and incidents from Hotline database to ascertain the effectiveness of the Anti-Fraud Hotline program.

14.5 Computer IT Security and IDEA

IDEA and the Computer IT Security Process

14.5.0 Introduction

It is vital to establish that controls in the IT/IS processes are in place and working. For example, to find out whether out of office hour internet access is possible, prohibited web-site internet access is possible, data back-ups are taken as per the scheduled frequency, virus updates are received regularly, and scans are conducted periodically.

CAATs –IDEA identifies instances where controls are failing because of data errors. . These factual cases serve well in building a strong defense and back-up for the auditor to express opinions on the processes.

IDEA can be applied to the Computer IT Security process from a Systems Audit perspective of testing IT General and Environmental Controls at any location of the entity.

Most of the tests within IDEA can be automated and scheduled, which is the heart of Continuous Control Monitoring and Pre-Audit of IT Security.

The main tenet of tests within IDEA for IT Security is availability, confidentiality, compliance and integrity of IT processes. IDEA can be used as a vital facilitator for detecting non-compliance with the Information Technology Act in India.

14.5.1 Potential Tests

System Logs

When auditing system logs, we should check:

- List:
    - o Accesses outside standard office hours or during holiday/sick leave
    - o All users with their normal computers
    - o All computers with their normal users
    - o Users on unusual computers
- Identify users, particularly those with supervisory rights, who are logged in for a long period of time.
- Analyze by user identify those with higher use than might reasonably be expected.
- Summarize by network address to identify.
- Summarize charges by user to determine resource utilization.
- Analyze utilization by period, such as daily, weekly, and monthly, to show historical trends.

File Lists

When performing auditing tests in regards to computer security, we may :

- List duplicate names (both software for multiple copies and data where there is a risk of accidental deletion).
- Identify old files.
- Analyze by directory.
- Analyze file size by owner.
- Identify last access dates for old files.
- Analyze file type (by file name extension).
- Identify all files without an owner, such as where user accounts have been removed from the system.
- Test for .com, .exe or .bat files in areas where there should not be programs. DOS/Windows systems.

Access Rights

In regards to access rights, we must check:

- Lists of:

  - Accounts with passwords not set or not required for access
  - Group memberships
- Accounts with:
  - Short access passwords (less than the recommended six characters)
  - No activity in the last six months
  - Access to key directories
  - Supervisor status
  - Equivalence to users with high level access, such as supervisory equivalence
- Aging of password changes

E-mail Logs

E-mail logs generally contain information such as the sender and recipient address, subject title, date and time of transmission, size of file, service provider etc. Ensure the organization has a published policy related to employee use of e-mail before undertaking any of these tests.

Common tests include:

- Total length of time spent on e-mails (receiving and responding) by organization as a whole, by individuals, by month
- Analysis of internal and external e-mails
- Summarizing by service providers
- Summarizing number of e-mails by an employee and sorting to capture top users.
- Isolating, summarizing and examining personal e-mail usage
- Stratifying time and examining any unusual activity e.g. lunchtime, weekends, bank holidays
- Stratifying by size of files
- Analyzing file attachments, by size, by type
- Analyzing incoming e-mails, identify common domain addresses

- Calculating length of time individuals spent on e-mail in a given time period, sort in order
- Matching with the list of employees and extract any e-mails that are sent by invalid employees or terminated employees
- Analyzing any dormant accounts
- Identifying non-work related e-mails by searching for specific words in the subject title e.g. weekend Auditing E-mail Logs

Firewalls

Many organizations implement a range of controls, including installing sophisticated firewalls to eliminate the risk of unauthorized access to their networks, especially via the Internet.

Firewall logs record all incoming and outgoing transmissions on a network and it is not unusual to record hundreds of thousands of activities in a single day. IDEA can be used to analyze the logs, identifying trends and exceptional items to follow up.

Firewalls generally contain information like the source and destination IP address, date and time of admission, action by the firewall on receipt of transmission, the service type and the service port accessed.

Common tests include:

- Summarizing the type of service being requested or being used
- Identifying the most common IP addresses attempting access to the network
- Summarizing actions upon connection, i.e. control, accept or drop
- Analyzing trends to determine the most common access times and identifying requests at unusual times
- Extracting all dropped transmissions
- Identifying potential attacks by looking for a pre-defined sequence of port scans e.g. SATAN, ISS attacks or searches for ports which can be used.

Chapter 15

Tally Accounting System and IDEA

IDEA and the Accounts Payables Process

15.0 Introduction

IDEA, like every other CAAT, can easily connect to an open data source. In fact, the precondition for the use of CAATs like IDEA is that there is an open unencrypted data source available for interaction.

In Tally Accounting System the data source is housed in the Tally Vault and hence protected. Direct interaction between IDEA and Tally Data Tables has not been established yet. In the given situation, IDEA interacts with reports from Tally. This is an established and well-recognized approach.

Here Account Ledger views (Display) are generated through the Balance Sheet and / or Profit and Loss Account option in Tally. These views are converted into a columnar representation, similar to an analytical Day Book view. Instead of printing out these columnar views, the view is exported to a File Format through the F12 command (for instance). The view can be saved as an ASCII, TXT, CSV, PRN, XML data format based on the need of the user. These formats can then in turn be imported through the relevant option in IDEA's import assistant.

In the case study to follow, we have imported an ASCII data file format.

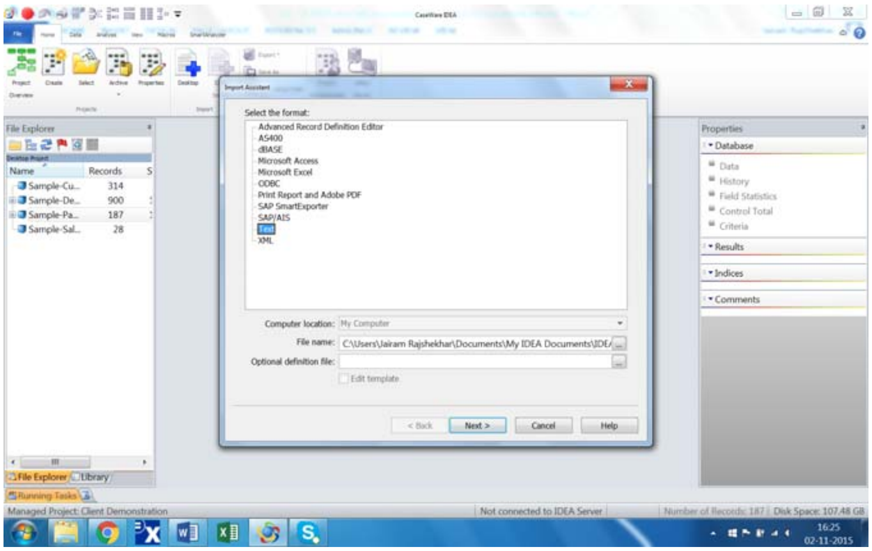
15.1 Import of a Tally 9 ERP Data File

"*The auditor entered the Expense Group of the Profit and Loss Account within Tally 9 for his client. He set the account period using the F2 option within Tally and generated a display of direct and indirect expenses. He then entered into direct expenses and generated a display of payments made towards manufacturing for the period 01.04.14 to 31.03.15. The expenses were displayed in a columnar ledger view with specific flags enabled for narrations and daily balances. Subsequently, the "Print to File" command was employed to capture the ledger display as an ASCII/ SDF file.*

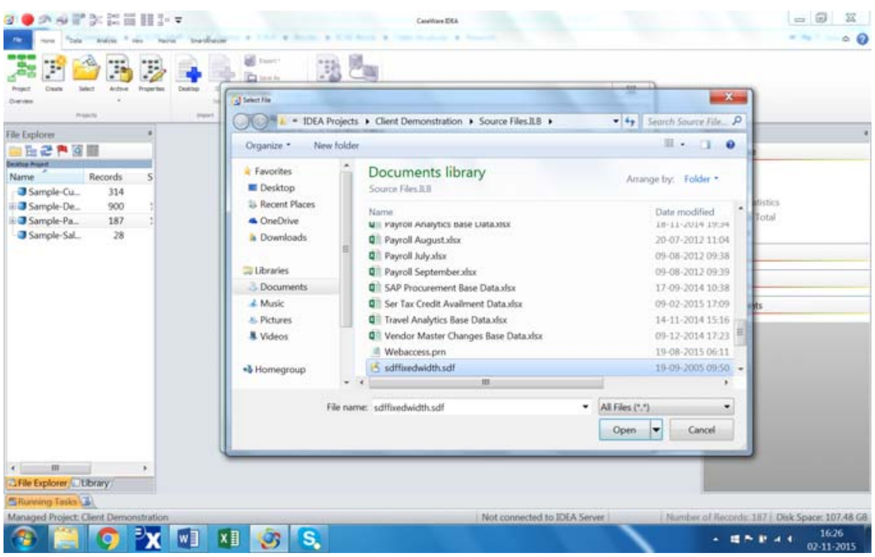
Steps used to import the Tally Data are

15.1.1 Click the Desktop icon under Home in the IDEA menu tool-bar

15.1.2 Select the Text option from the Import Assistant.



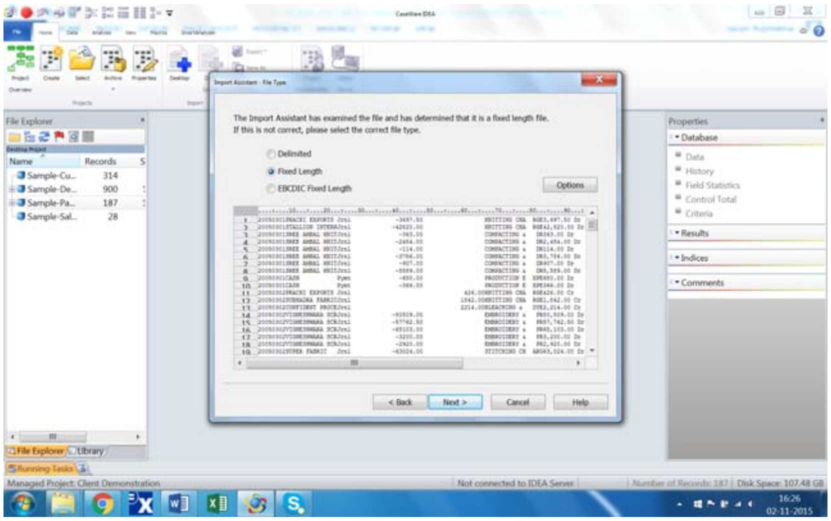
15.1.3 Click the browse button by the side of File name and select the file from the Source Files folder location on the computer.



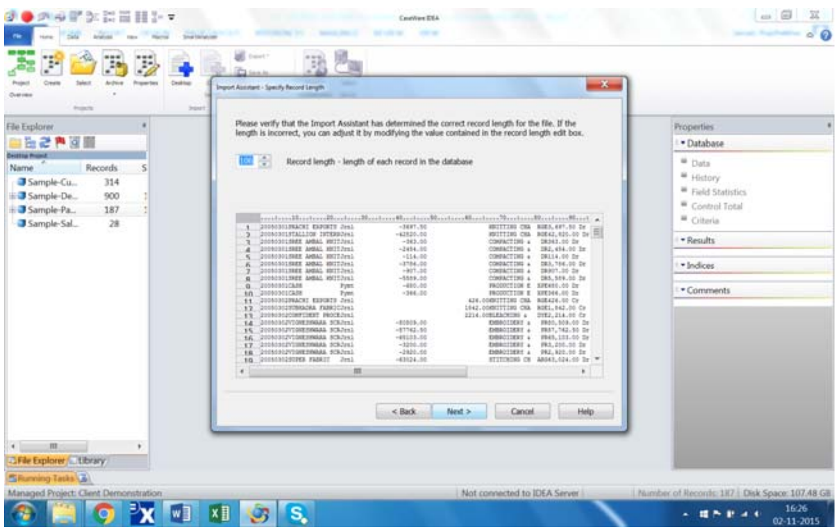
15.1.4 Click Open and then Next.

15.1.5 IDEA scans the tally file and recognizes it to be a Fixed Length ASCII file.

(SDF files are “Fixed length ASCII” files)

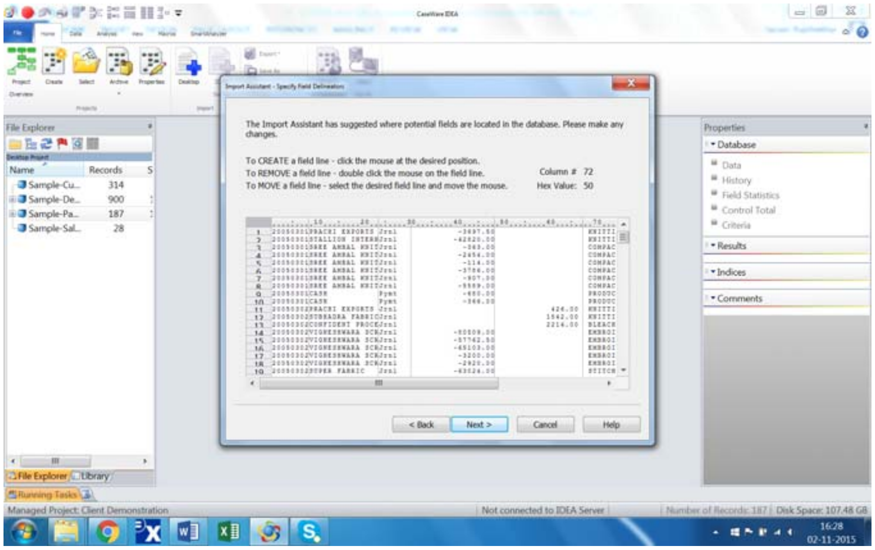


15.1.6 IDEA scans the record length of the source file and, by default, determines the file record length. In the case under demonstration it is 106.



15.1.7 Click Next

15.1.8 IDEA automatically places Field Separator Lines at the points where each logical field ends and the next begins. This at times requires to be changed to ensure that all logical units of data (Fields) get correctly selected/ imported and properly formatted.



15.1.9 Insert field separators at the end of each field by clicking the cursor in the data content screen once at the required point.

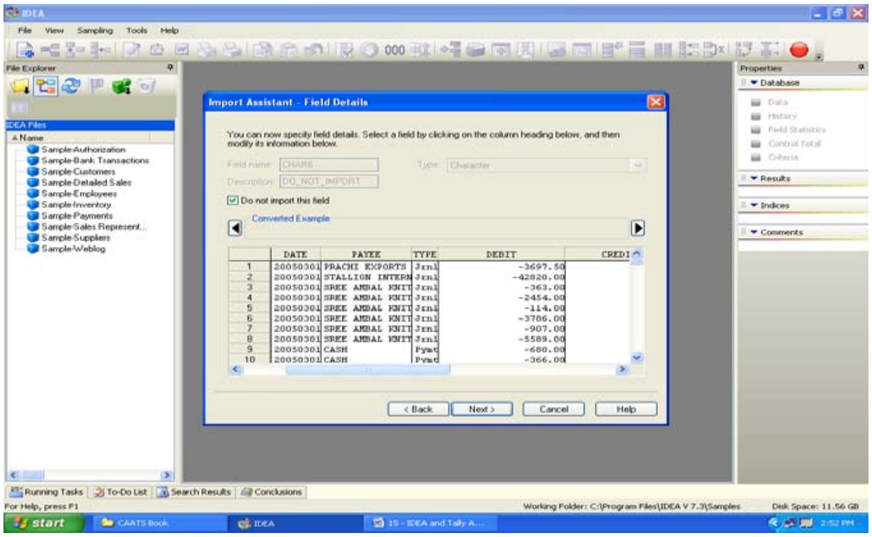
To remove an unwanted field, click the mouse point twice over the unwanted line.

To move a field line place the mouse point on the select line, keeping the left button of the mouse pressed, and drag the line to the point of choice.

15.1.10 After giving the field line breaks, click Next.

15.1.11 Now enter the Field Names.

15.1.12 We have an option of not importing any field. Click Do not import this field in the check box.



15.1.13 Then click each field heading, enter the field name, type and description.

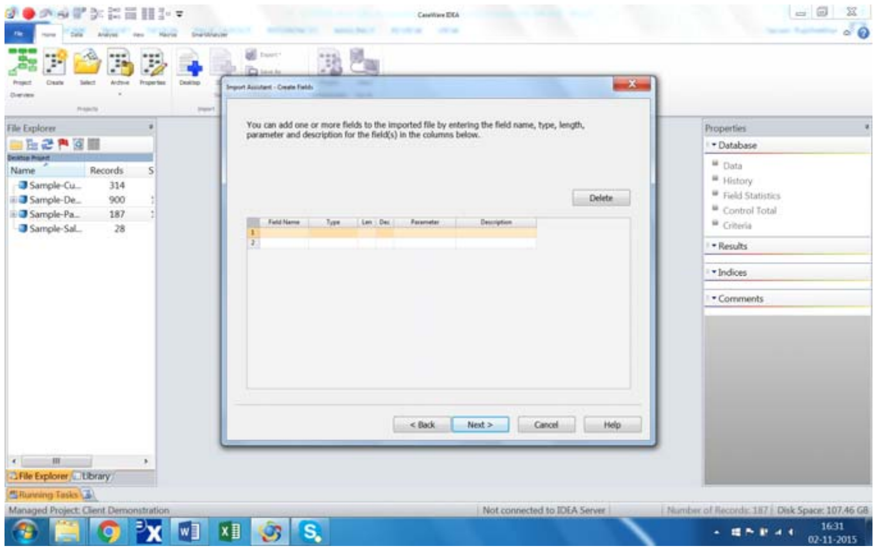
15.1.14 For example, for the first field the name is Date, type is Date and Date Mask is YYYYMMDD, which matches with the actual view of the date data in the data content section. For example 20050301 will necessitate as mask as YYYYMMDD

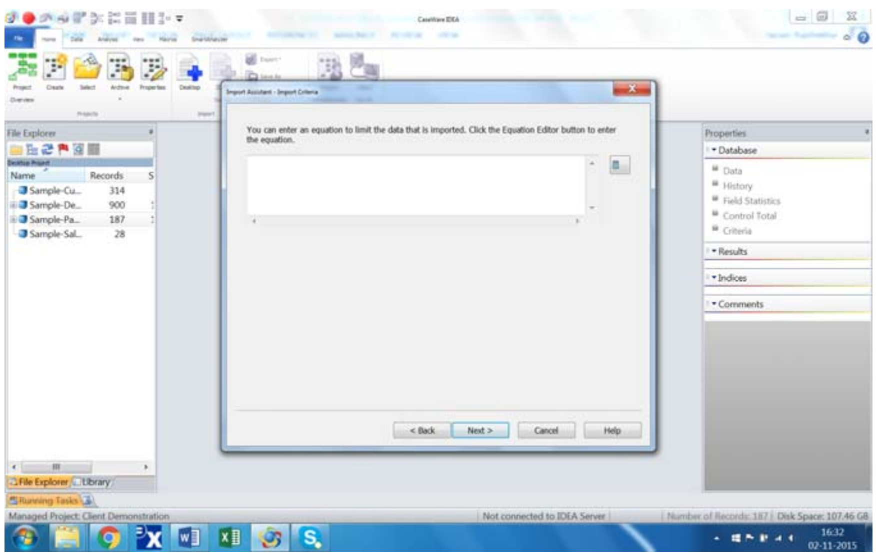
15.1.15 Define all the fields as per the following File Layout

Field Name	Field Type	Remarks
Date	Date	Mask YYYYMMDD
Vendor	Character	
TXN TYPE	Character	
DEBIT	Numeric	
CREDIT	Numeric	
AC HEAD	Character	
BALANCE	Character	

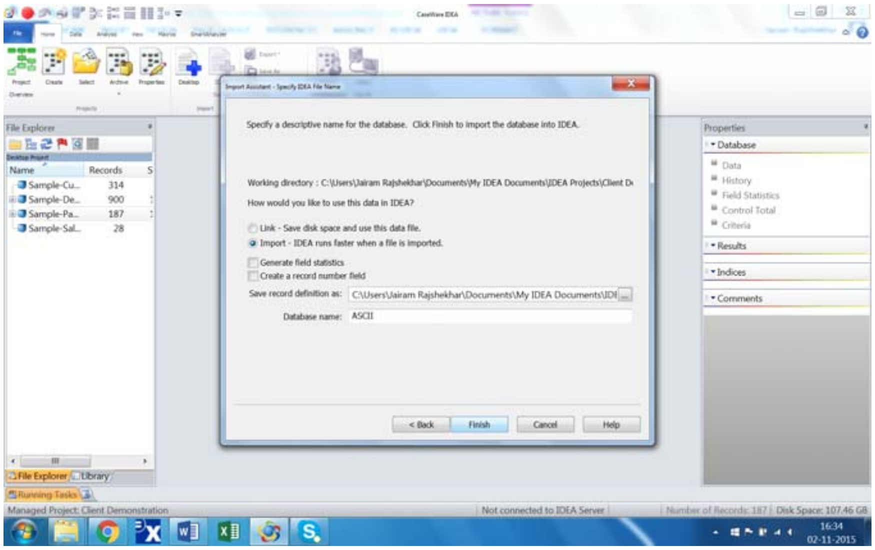
15.1.16 We may create additional fields over and above the existing fields at this stage.

But in this presentation we will not create any additional fields. Just Click Next.

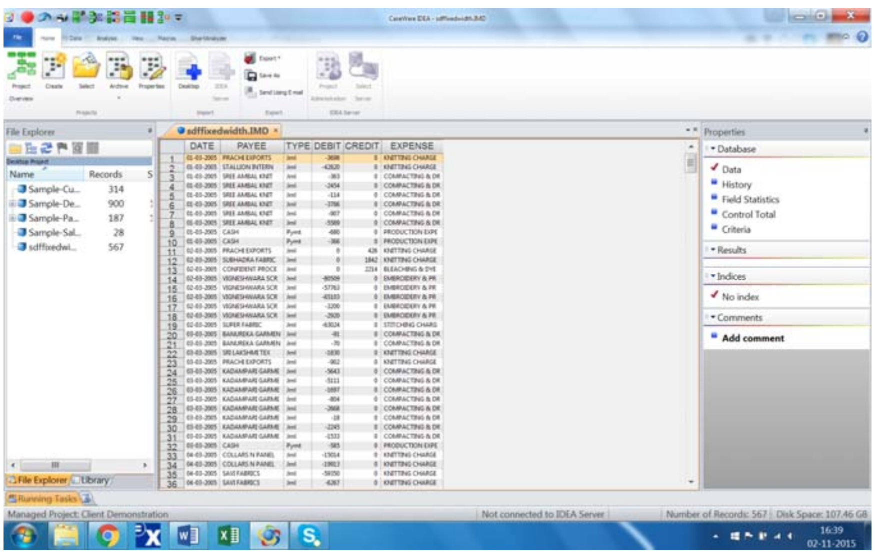




15.1.17 In the final 'Specify IDEA File Name' screen enter the file name ASCII and click Finish



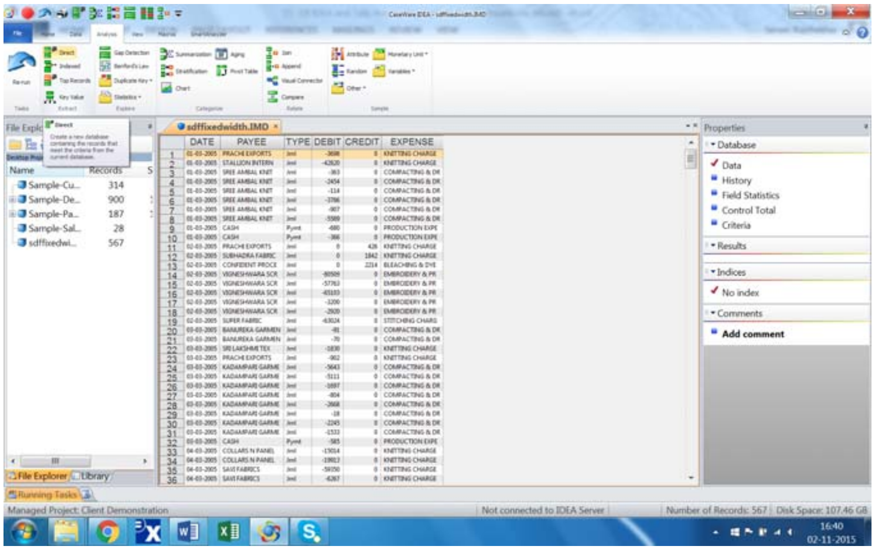
15.1.18 Once imported, the file looks like what is shown below:



15.2 Analysis of Tally 9 ERP Data in IDEA

15.2.1 Extract Cash Payments

Go to Analysis – Extract - Direct



This will open the Direct Extraction dialog box.

Enter the File Name as Cash Payments

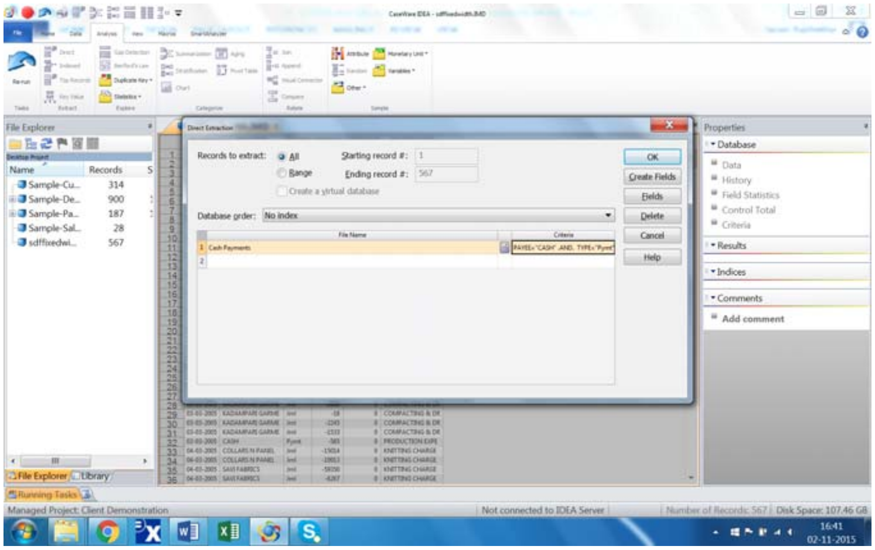
Click the CRITERIA button to the right of the File Name

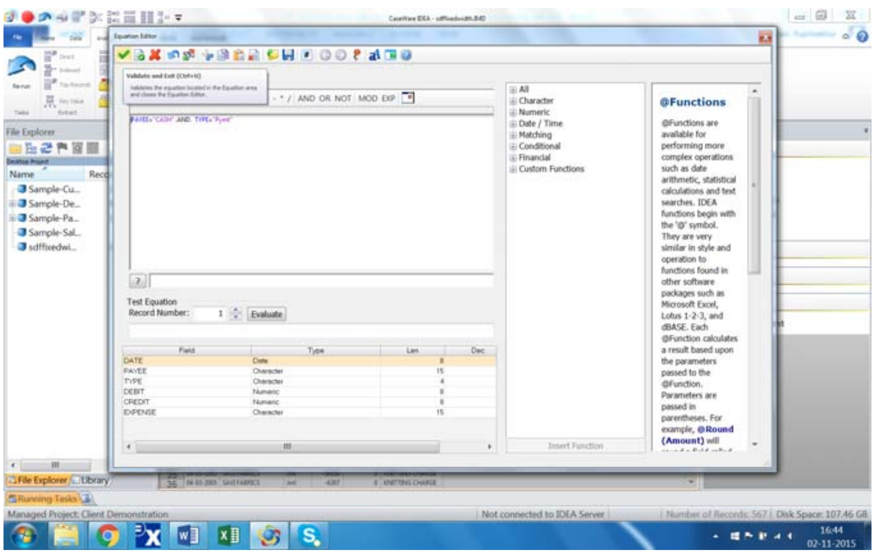
This will open the Equation Editor.

In the Equation Editor, write the Equation

PAYEE=”CASH”. AND. TYPE=”Pymt”

Enter the field names PAYEE and TYPE by entering the first letters of the field names in the Equation Box. The Field names appear automatically by clicking the first letter. Then select the same and press enter. Enter CASH and Pymt in “ “ since they are Character fields. Enter these words by hand. Enter .AND. by clicking the button .AND. in the Equation keypad

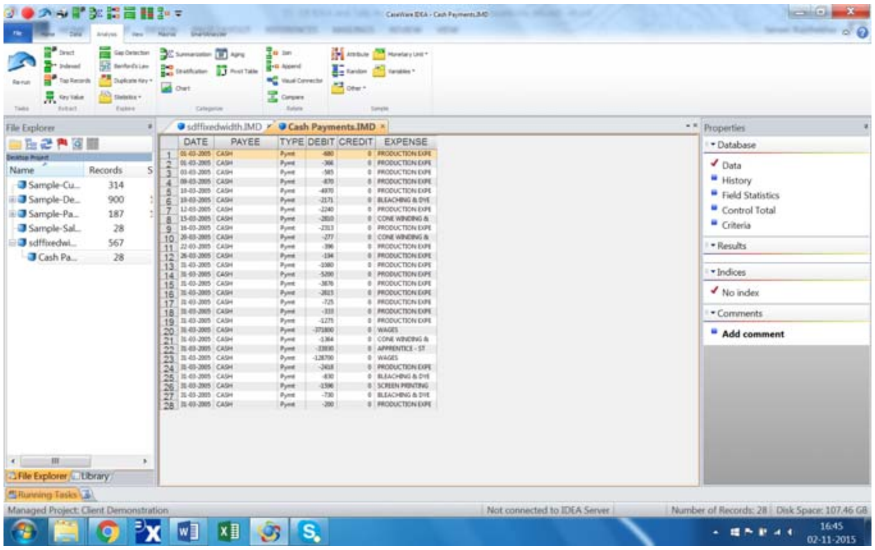




Click the Green Check button to Validate and Exit.

Then Click OK in the Extract to file (s) dialog box.

We will get a list of 28 Cash Payments, as can be seen below, which will serve as the basis for further tests to follow.



15.2.2 Identify Cash Payments made on Sundays which is the Company Weekly Off

Go to Analysis – Extract - Direct

Enter the File Name as Sunday Payments in the Direct Extraction dialog box.

Click the Criteria button to open the Equation Editor.

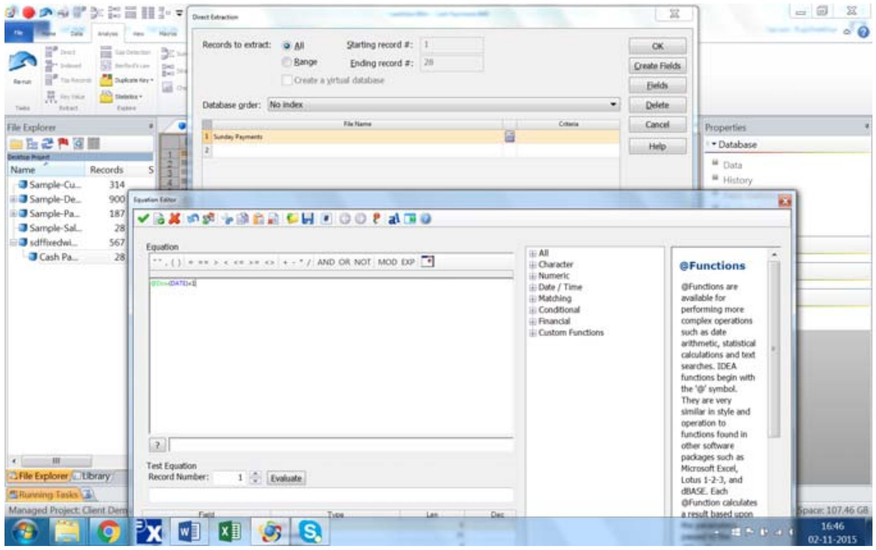
In the Equation Editor write the function

@dow(date)=1

Enter the @ function - @dow by pressing shift @ in the Equation box. This will bring up a list of all @ functions. Now enter dow, and we will get to view the function @ dow.

Press enter, open bracket enter d, select date as the field on the basis of which we have to run the @ dow function, close bracket and enter = 1.

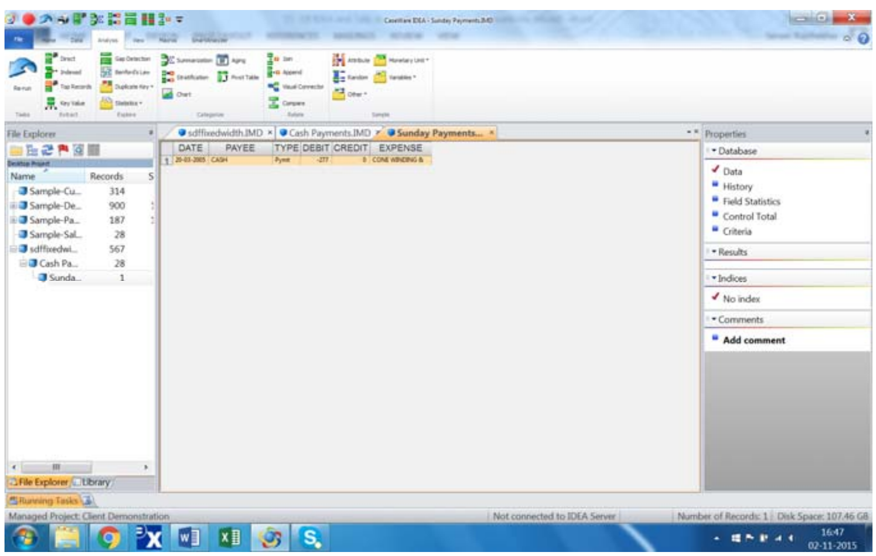
This function will extract all cash payments made on Sundays, since Sunday is 1 within IDEA.



Click he Green Check sign in the Equation Editor to Validate and Exit the Equation Editor.

The function gives one instance of cash payment made on a Sunday.

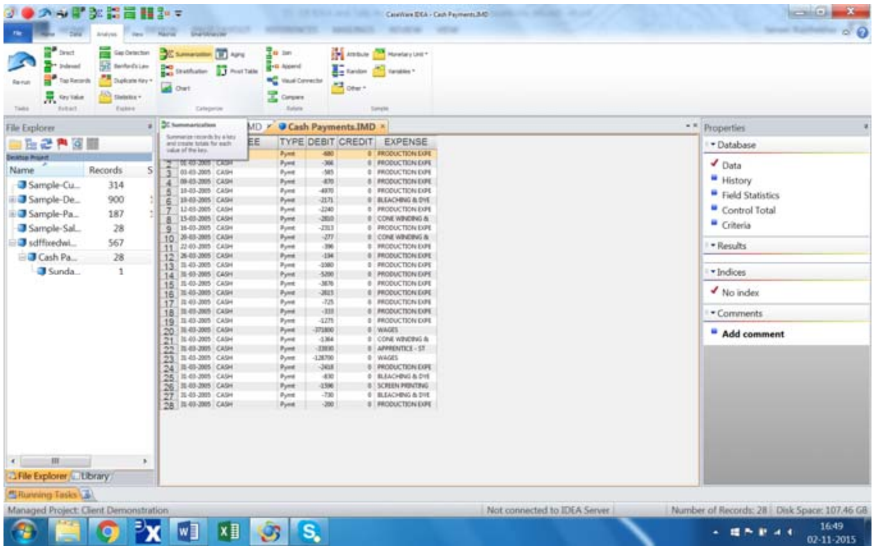
This case needs to be taken up for investigation with the Accounts section.



Close the file Sunday Payments and open Cash Payments from the File Explorer.

15.2.3 Identify the most expensive Cash Payment Expense Head

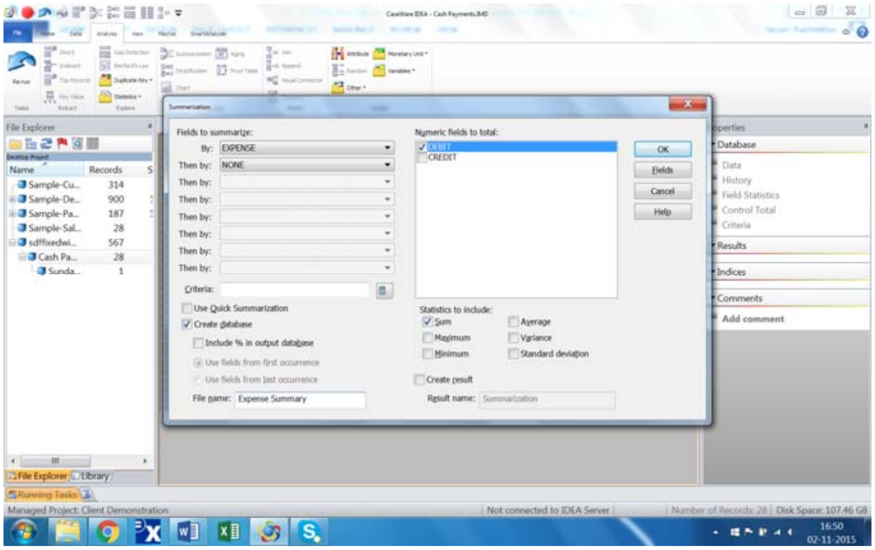
Go to Analysis – Field Summarization – Quick



This will open the Summarization dialog box.

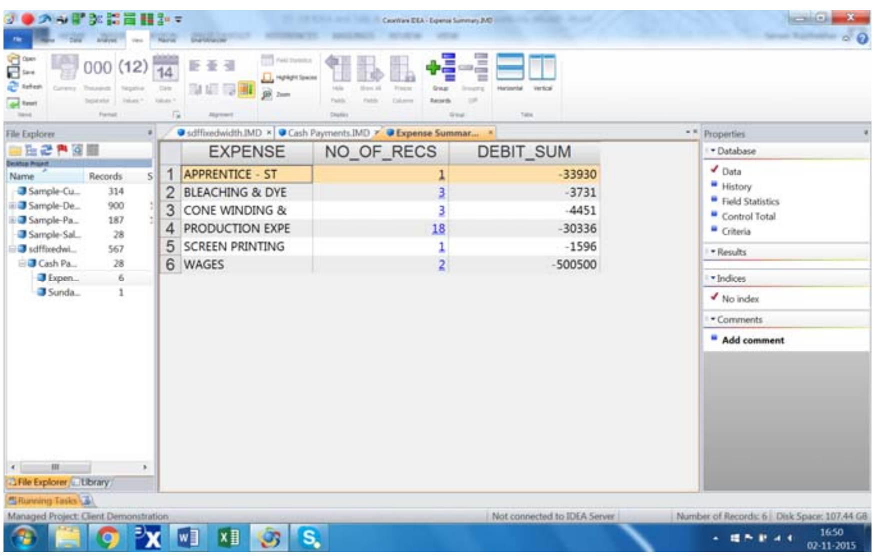
Select the Summarization Field as EXPENSE from the drop down list of active fields

Select the Fields to Summarize as Debit from the drop down list of active fields.

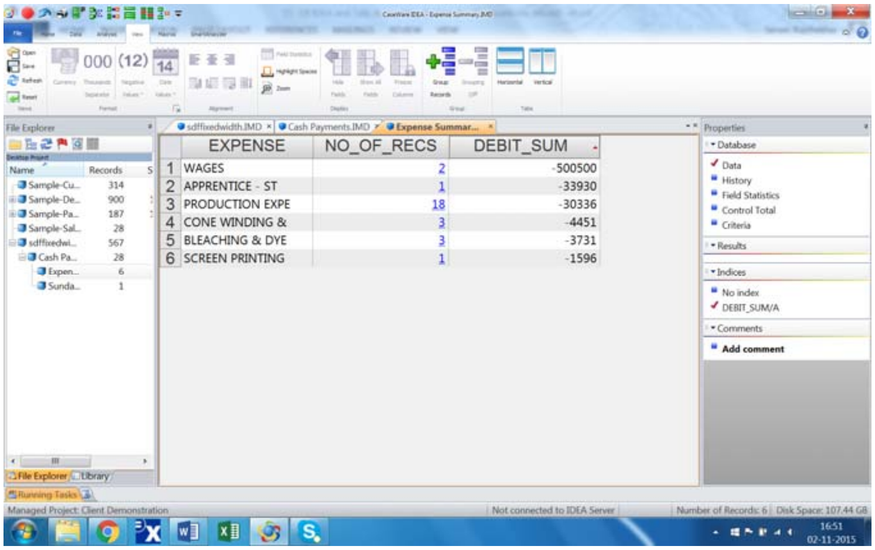


Click OK

The summarization gives a tab result in the file Cash Payments listing the Account Head, number of records for each payment and the sum total value.



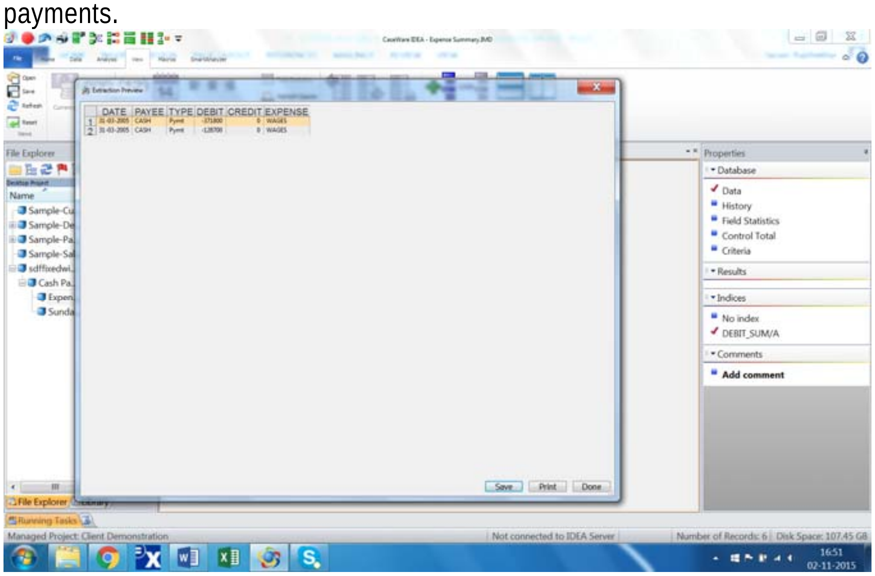
Double click the field-heading DEBIT. The field gets sorted on ascending basis and reveals the most expensive cash payment head: Wages Rs. 500500.



If we click the number of records field cell in NO_OF_RECS for Wages i.e. 2, it will give us a detailed listing of cash payments to workers.

A closer scrutiny reveals that these cash payments exceed Rs. 20000 under the Tax Audit rules. However, these entries are single combined entries.

Ask the Accountant for detailed listing of wage payments in cash to identify inadmissible payments under the Tax Audit Code for single transaction cash payments.



Chapter 16

Fraud Investigation using IDEA

16.0 Introduction

The degree of automation in the typical business environment has led to an increase in the complexity of internal control systems. Technological advances in distributed processing, worldwide networking, and remote access to corporate systems, for example, increase organizations' vulnerability to control breaches and present new challenges for the internal auditor. The impact of such technology on forensic audit practices is especially significant.

A recent study by the Association of Certified Fraud Examiners indicates that financial losses due to fraud in the United States amounted to a staggering $400 billion in recent years. As more business operations use computerized information, more fraud is committed via computer assisted means. The technology that enables this type of fraud, however, also provides auditors with more sophisticated weapons to fight it. In fact, fraud detection is an ideal application for computer assisted audit techniques (CAATs). In recent years, analytical techniques have become not only more powerful but also more widely used by auditors for forensic investigation.

CAATs enable investigators to obtain a quick overview of business operations, develop an understanding of the relationships among various data elements, and drill down into the details of specific areas of interest. A systematic approach to fraud investigation that involves the identification of unusual activity with the aid of CAATs, including the use of digital analysis techniques such as Benford's Law, can help to ensure that misdoings get detected as close to the event as possible.

IDENTIFYING FRAUD WITH CAATS

While it does not "take one to know one" with regard to fraud investigation, auditors seeking to detect fraud would certainly want to know what it looks like. This requires auditors to recognize the tell-tale evidence of fraud and to understand how the data they obtain can be used to verify whether or not a fraudulent act has been committed. Data analysis software can assist by highlighting transactions that contain the characteristics often associated with fraudulent activity. Internal auditors can review millions of transactions, including data from previous years or multiple locations, and probe anomalies by analyzing information across databases. Auditors should begin their analysis by pinpointing hot spots and then searching for patterns among the data.

LOOKING FOR SYMPTOMS Data analysis programs feature many commands that review records for fraud symptoms, such as the existence of duplicate transactions, missing transactions, and other anomalies. Examples of applications for these types of functions include:

- Comparing employee addresses with vendor addresses to identify employees who are also vendors.
- Searching for duplicate check numbers to find photocopies of company checks.
- Scanning the list of vendors to identify those with post office boxes for addresses. These records can be easily extracted from the vendor file for further follow-up to ensure they represent legitimate vendors.
- Analyzing the sequence of all transactions to identify missing checks or invoices.
- Identifying all vendor companies that have more than one vendor code or more than one mailing address. Such listings may represent "phantom" vendors, which exist for the sole purpose of stealing payments from companies.
- Finding several vendors with the same mailing address. These records may also signal phantom vendors.

Investigators can narrow their search for symptoms by forming hypotheses about the data and then testing them with CAATs. For example, an auditor could reasonably assume that each employee in the organization receives only one paycheck per pay period. Using CAATs to search for duplicate records in a single pay period would quickly test the validity of the assumption by highlighting all instances where employees received more than one paycheck. If the assumption is correct, the existence of employees receiving two or more paychecks may indicate fraud. If the assumption is invalid, some employees may receive a separate check for overtime in addition to their regular salary. For instance, the auditor can then form a revised hypothesis that each employee should receive only one paycheck for regular hours per pay period. The database search would then be revised on the basis of this new assumption.

CAATs can also help to reveal fraud symptoms that may be elusive because the evidence is spread across separate databases. For example, reviewing data from the accounts payable file may identify a trend in the expenditures to a particular vendor that seems unusual, but not necessarily indicative of fraud. Combining the accounts payable data with information from the contracting database, however, may reveal that all contracts with the vendor in question were raised by one contracting officer. This type of relationship among the data would indicate the possibility of a fraudulent activity, such as kickbacks received by the officer from the vendor. Conducting a search of this type would be impractical, even impossible with either sampling techniques or other manual methods of investigation.

PATTERNS OF FRAUD Experience and training in electronic fraud investigation enables auditors to recognize and identify data patterns that may be associated with fraud. Patterns such as negative entries in an inventory-received field, voided transactions followed by a "no sale," or a high percentage of returned items may indicate of fraudulent activity.

As investigators become more familiar with the characteristics of suspicious patterns, they can develop sets of predefined criteria, and transactions meeting these criteria can trigger automatic investigations. In other words, auditors can use CAATs to create a "fraud profile," which functions as a template for future audits. The template may also be used to monitor transactions on an ongoing basis, thereby functioning as a continuous tracking device to flag any suspicious activity, as soon as it appears in the database. This technique can serve as an instrumental means of reducing the number of losses attributed to fraudulent acts. For instance, a timely review of credit card transactions can identify unusual spending patterns that might represent the symptom of a stolen credit card. Continuous monitoring for suspicious patterns allows auditors to catch fraud early, before serious damage can occur.

16.1 Benford’s Law of Digital Analysis

16.1.1 Evolution of the Law & its Concept

Frank Benford was a physicist at GE Research Laboratories in the 1920s. He noted that the first parts of the log table books were more worn than the back parts. The first pages contain logs of numbers with low first digits. The first digit is the left-most digit in a number.

Benford collected data from 20 lists of numbers totaling 20,229 observations. He found that the first digit of 1 occurred 31 percent of the time. Using integral calculus, he calculated the expected digit frequencies that are now known as “Benford’s Law”. It took him six years to perform his analysis and develop his law of expected digit frequencies.

The Benford’s Law task in IDEA can provide a valuable reasonableness test for large data sets. IDEA only tests items with numbers over 10.00. Number sets with less than four digits tend to have more skewed distributions and do not conform to Benford’s Law. Positive and negative numbers are analyzed separately. This is because abnormal behavior patterns for positive numbers are very different from those for negative numbers.

16.1.2 Application to Data Mining

The application of Digital Analysis and the Benford Module is also permissible in the framework of Data Mining where certain distinctive facts in a data supply are measured against the personal expectations of the user and interpreted according to them. In this case it is not necessary that the data to be analyzed create a Benford Set, . In fact, it is permissible under these circumstances to analyze the numerical distribution of the leading digits of each data quantity and to interpret it independent of Benford’s Law.

16.1.3 Assumptions of Benford’s Law

Geometrical Series

The mathematical pre-condition for the examination of a data supply based on Benford’s Law is that the data supply is based on a geometrical series (thus, it is presented as a Benford Set). In reality this condition is rarely met. Experience, however, confirms that data must only partially meet this condition, i.e., the constant increase, percentage-wise of an element compared to the predecessor must only be met partially. Otherwise, this would mean that no number may occur twice, which is quite improbable in the case of business data supplies. However, the pre-condition is that there is at least a “geometrical tendency”.

Description of the same object

The data must describe the same phenomenon. Examples are:

- The population of cities
- The surface of lakes
- The height of mountains

- The market value of companies quoted on the NYSE
- The daily sales volume of companies quoted on the Stock Exchange
- The sales figures of companies

Unlimited data space (non-existence of minima and maxima)

The data must not be limited by artificial minima or maxima. A limitation to exclusively positive numbers (excluding 0) is permissible as long as the figures to be analyzed do not move within a certain limited range. This applies, for example, to price data (e.g., the price of a case of beer will generally always range between 15 and 20 dollars) or fluctuations in temperature between night and day.

No systematic data structure

The data must not consist of numbers following a pre-defined system, such as account numbers, telephone numbers, and social security numbers. Such numbers show numerical patterns that refer to the intentions of the producer of the number system rather than to the actual object size, represented by the number (e.g., a telephone number starting with 9 does not mean that this person possesses a bigger telephone).

16.1.4 Statistical Explanations for the Law

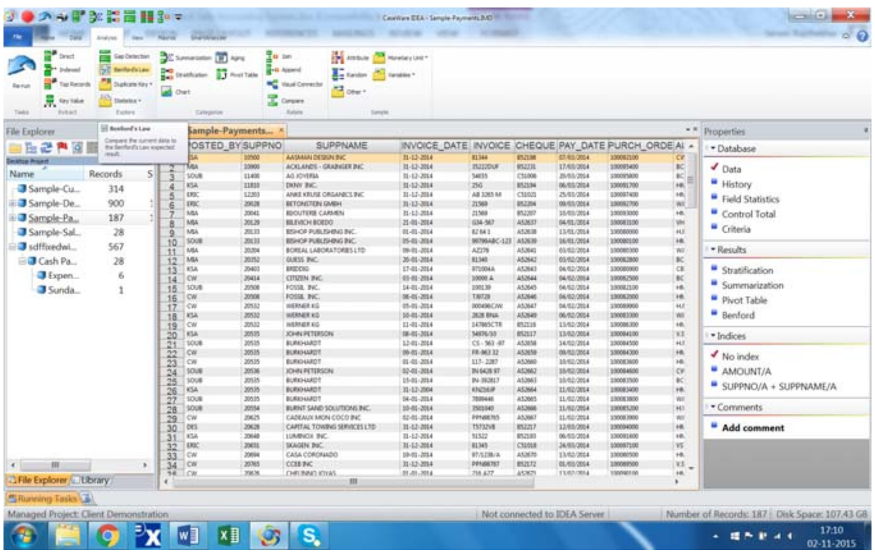
Basically, data complies best with Benford's Law if it meets the rules mentioned above, namely that the data consists of large numbers with up to 4 digits and the analysis is based on a sufficiently large data supply. A large data supply is necessary to come as close to the expected numerical frequencies as possible. For example, the expected frequency of the digit 9 in any data supply is 0.0457. If the data supply consists of only 100 numbers, the numbers which have a 9 as their first digit may be 5% of the data supply. Thus in the case of a small data supply, there may be an over-proportional deviation from Benford's Law. In large data supplies, the numerical distribution is increasingly closer to the expected frequencies.

If the data supply has, or just roughly has, the characteristics mentioned above, it can be analyzed based on Benford's Law. However, the results of the Benford analyses are not interpretable on the basis of Benford's Law. As stated before, the expected frequencies according to Benford's Law often represent, in the practical use, nothing more than a type of benchmark for the observed frequencies. Since the observed frequencies will only be compared with the legality discovered by Benford, not interpreted accordingly, it is not necessary that all conditions mentioned above be met. In fact, the analysis results will help the auditor to interpret the personal expectation of the user, without including the reference value according to Benford in the investigation. If, for example, the personal expectation of the user is that the starting digit 4 must occur twice as often in the analyzed data than the starting digit 2, the results of the analyzed values must not be compared with the expected frequencies according to Benford, but with the individual expectation of the user.

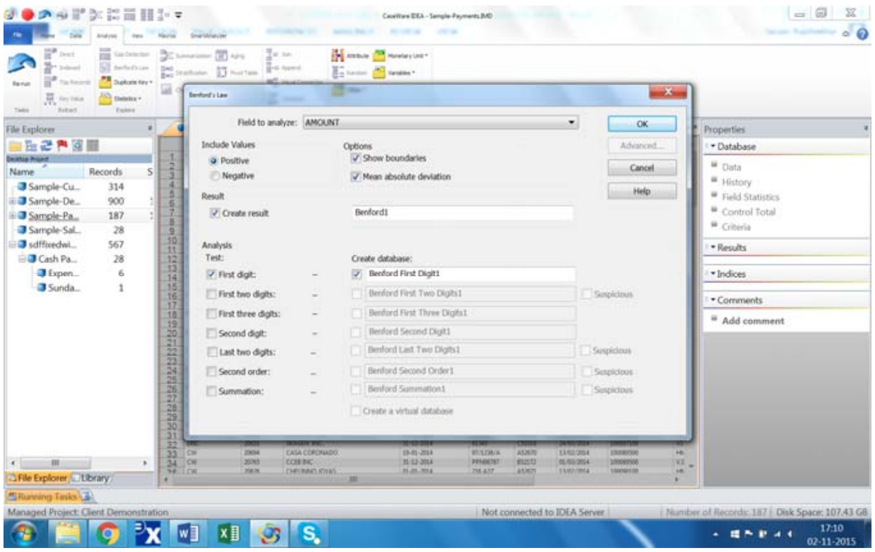
16.1.5 Case Study on Benford's Law within IDEA

To run a **Benford's Law** analysis on the **Accounts Payable** database

1. From the **Analysis** menu, select **Benford's Law.**



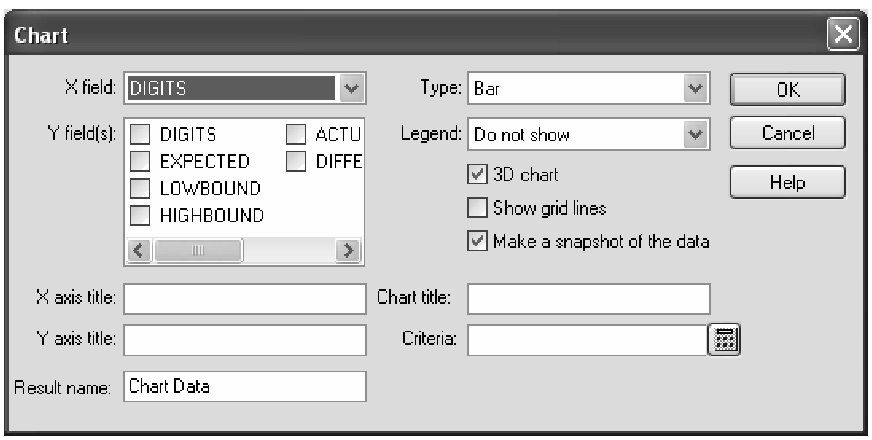
2. In the **Benford's Law** dialog box, select **AMOUNT** as the field to be analyzed. Accept all other default options as displayed in the image below.



Click **OK** to perform the analyses.

The **Benford First Digit** database becomes the active database. Other databases should be opened from the *File Explorer*.

3. To graph the data, select **Chart Data** from the **Data** menu.



In the Y field(s) box, select **ACTUAL**.

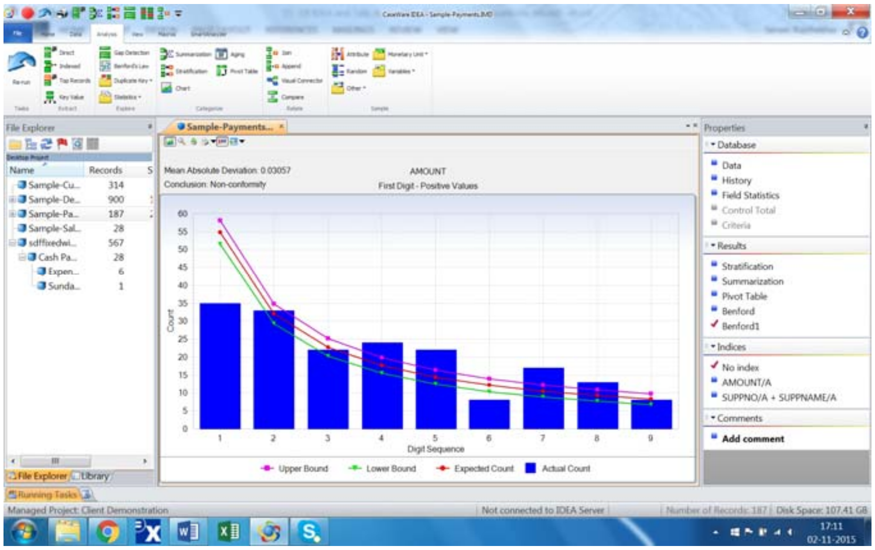
In the X axis title box, enter **Digit Sequence**.

In the Y axis title box, enter **Count**.

In the Chart title box, enter **AMOUNT – First Digit – Positive Value**.

Click **OK**.

The **Chart Data Results** output becomes active.



The first digit graph shows a spike in the digit 7 results.

Technical Notes

The **First Digit** test is the test of first digit proportions. The first digit of a number is the leftmost digit in the number. Zero can never be the first digit. This is a high level test. Analysts will not usually spot anything unusual unless it is blatant. This is a test of goodness-of-fit to see if the first digit actual proportions conform to Benford's Law. The **First Digit** test is an overall test of reasonableness. The upper and lower bounds are merely guidelines for the auditor. The **First Digit** graph could show a high level of conformity but the data set could still contain errors or biases.

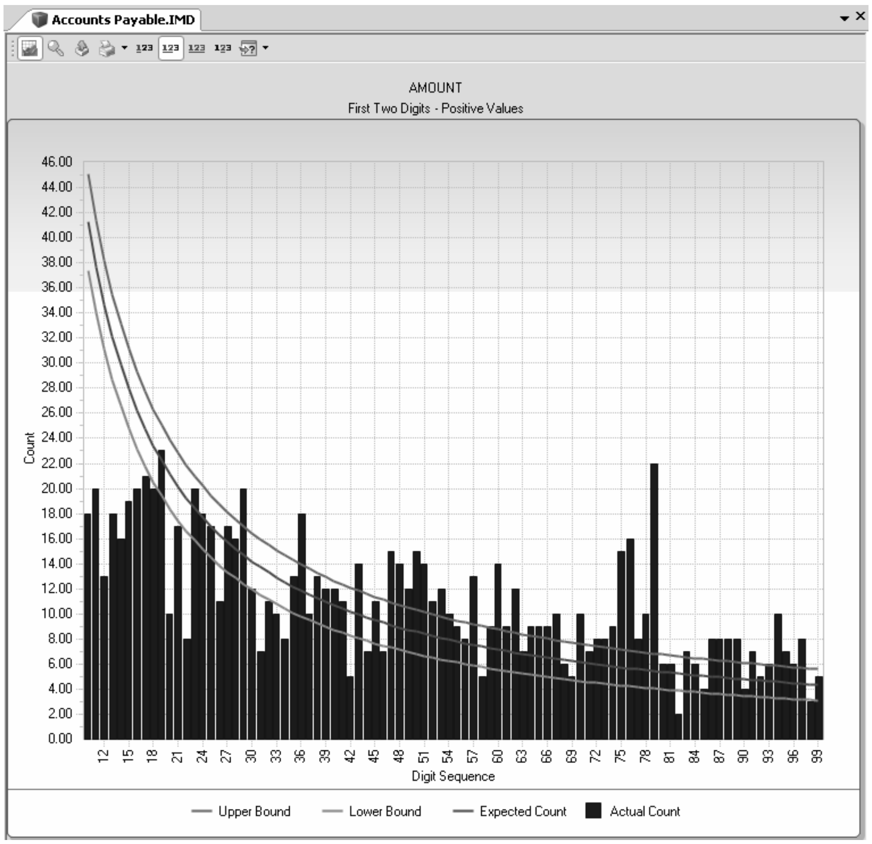
4. Click the **Data** link in the *Properties Window* to return to the **Benford First Digit** database that was created as part of this analysis. The **DIFFERENCE** field shows the difference between the expected and actual occurrences of the digits. When the **DIFFERENCE** field is indexed in ascending order, the digit 7 results show the largest negative difference (positive spike).

	DIGITS	EXPECTED	LOWBOUND	HIGHBOUND	ACTUAL	DIFFERENCE
1	7	57.82	53.26	62.38	113	-55.18
2	5	78.94	73.61	84.27	106	-27.06
3	6	66.75	61.84	71.65	90	-23.25
4	9	45.62	41.57	49.67	61	-15.38
5	8	51.00	46.71	55.28	63	-12.00
6	4	96.62	90.72	102.52	108	-11.38
7	3	124.56	117.87	131.26	114	10.56
8	2	175.56	167.61	183.51	154	21.56
9	1	300.13	289.73	310.52	188	112.13

This result warrants further investigation, as the CFO has indicated that any items in excess of Rs.80,000.00 require additional approval. This spike could be indicative of an abnormal level of items being processed just below the additional approval level.

Close the **Benford First Digit** database.

5. To view the **Benford's Law** analysis result again, ensure that **Accounts Payable** is the active database and click the **Benford** link in the **Results** area of the *Properties Window*. To view the **First Two Digit** graph, click on the  button on the **Results** toolbar. The 79, 76, and 75 two-digit combination spikes are clearly visible in this graph.



Technical Notes

The **First Two Digit** test is a more focused test. The first two digit numbers are the leftmost two digits. There are 90 possible two-digit combinations ranging from 10 to 99. This test is performed to find anomalies in the data that are not readily apparent from either the **First Digit** test or the **Second Digit** test when viewed on their own. A spike occurs where the actual proportion exceeds the expected proportion as predicted by Benford's Law. Positive spikes (above the Benford's curve) represent excessive duplication. One of the objectives of this test is to look for spikes that conform to internal thresholds like authorization limits.

6. We look at the transactions that comprise the 79 two-digit combination by clicking on the graph and selecting **Display Records**

to drill down to the transactions. We notice the number of transactions just under the Rs. 80,000 approval limit.

Preview Database

	SUPPNO	PAYEE	INVOICE	INV_DATE	AMOUNT	CHECK	PAY_DATE	AUTH
1	C202	Cary S Matic	CS - 589 -97	22/03/2006	79,217.47	701241	16/04/2006	H.M.V.
2	M025	Luke Hair	51505	24/01/2006	79,500.00	701073	17/02/2006	H.M.V.
3	F128	Ern Payed	82 95 1	15/02/2006	79,826.53	701148	13/03/2006	WJT
4	P007	Nellie Dunn	000511CJW	26/02/2006	7,988.72	701179	26/03/2006	WJT
5	T004	Richmond	IN/1151/97	07/03/2006	79,839.54	701207	01/04/2006	WJT
6	N001	Mike Atsil	AZ362	22/04/2006	79,910.95	701346	17/05/2006	HV
7	R007	Philip Upp Garage	1421/21	01/07/2006	79,365.72	701558	28/07/2006	HMV
8	M020	Linda Hand	GR223 97	04/07/2006	79,002.68	701572	30/07/2006	HMV
9	R008	Philip Upp Garage	1464/21	16/10/2006	7,973.99	701910	12/11/2006	BC
10	H025	Ivan Aker	WNZ28C	02/01/2006	79,571.88	701002	31/01/2006	BC
11	T006	Round Table	BUR1355-G	27/03/2006	79,839.65	701260	25/04/2006	H.M.V.
12	K001	Jackie Tupp	917356	22/03/2006	79,990.86	701262	19/04/2006	H.M.V.
13	N005	Miles Long	FR-1026 82	30/06/2006	79,306.51	701547	28/07/2006	HMV
14	M014	Kurt N Upp	123513	16/07/2006	79,902.90	701628	13/08/2006	HMV
15	M014	Kurt N Upp	123522	19/08/2006	79,765.18	701732	16/09/2006	WJT
16	B008	A Raid	3501101	07/08/2006	7,948.57	701586	05/08/2006	WJT
17	R020	Polly Gunn	L-1347/73	02/09/2006	79,458.30	701765	30/09/2006	WJT
18	P009	P Green	566698T	04/01/2006	79,237.49	701012	03/02/2006	BC
19	Z001	Miles Long	FR-969 39	17/01/2006	795.59	701064	15/02/2006	HMV
20	T005	Ri Pent	PPN98831	07/04/2006	79,086.25	701300	06/05/2006	BC
21	R007	Peter Rabbitt	KG0385	16/09/2006	79,901.01	701820	15/10/2006	BC
22	N005	Miles Long	FR-1073 140	25/10/2006	7,962.64	701943	25/11/2006	HMV

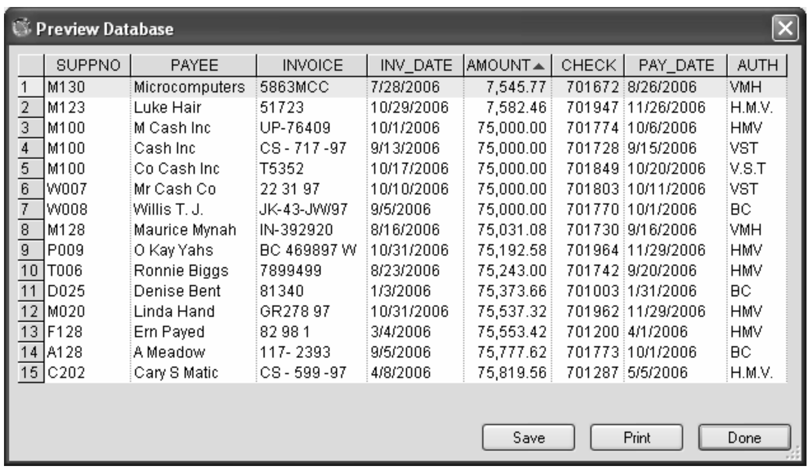
Of the 22 transactions that make up the 79 two-digit combination, 17 are between 79,000 and 80,000. Each of the payables clerks authorized some of the 17 transactions, but HMV was responsible for the bulk of these transactions, having authorized 9 out of 17.

Preview Database

	SUPPNO	PAYEE	INVOICE	INV_DATE	AMOUNT	CHECK	PAY_DATE	AUTH
1	K001	Jackie Tupp	917356	22/03/2006	79,990.86	701262	19/04/2006	H.M.V.
2	N001	Mike Atsil	AZ362	22/04/2006	79,910.95	701346	17/05/2006	HV
3	M014	Kurt N Upp	123513	16/07/2006	79,902.90	701628	13/08/2006	HMV
4	R007	Peter Rabbitt	KG0385	16/09/2006	79,901.01	701820	15/10/2006	BC
5	T006	Round Table	BUR1355-G	27/03/2006	79,839.65	701260	25/04/2006	H.M.V.
6	T004	Richmond	IN/1151/97	07/03/2006	79,839.54	701207	01/04/2006	WJT
7	F128	Ern Payed	82 95 1	15/02/2006	79,826.53	701148	13/03/2006	WJT
8	M014	Kurt N Upp	123522	19/08/2006	79,765.18	701732	16/09/2006	WJT
9	H025	Ivan Aker	WNZ28C	02/01/2006	79,571.88	701002	31/01/2006	BC
10	M025	Luke Hair	51505	24/01/2006	79,500.00	701073	17/02/2006	H.M.V.
11	R020	Polly Gunn	L-1347/73	02/09/2006	79,458.30	701765	30/09/2006	WJT
12	R007	Philip Upp Garage	1421/21	01/07/2006	79,365.72	701558	28/07/2006	HMV
13	N005	Miles Long	FR-1026 82	30/06/2006	79,306.51	701547	28/07/2006	HMV
14	P009	P Green	566698T	04/01/2006	79,237.49	701012	03/02/2006	BC
15	C202	Cary S Matic	CS - 589 -97	22/03/2006	79,217.47	701241	16/04/2006	H.M.V.
16	T005	Ri Pent	PPN98831	07/04/2006	79,086.25	701300	06/05/2006	BC
17	M020	Linda Hand	GR223 97	04/07/2006	79,002.68	701572	30/07/2006	HMV
18	P007	Nellie Dunn	000511CJW	26/02/2006	7,988.72	701179	26/03/2006	WJT
19	R008	Philip Upp Garage	1464/21	16/10/2006	7,973.99	701910	12/11/2006	BC
20	N005	Miles Long	FR-1073 140	25/10/2006	7,962.64	701943	25/11/2006	HMV
21	B008	A Raid	3501101	07/08/2006	7,948.57	701586	05/08/2006	WJT
22	Z001	Miles Long	FR-969 39	17/01/2006	795.59	701064	15/02/2006	HMV

Of the 16 transactions that make up the 76 two-digit combination, 10 are between 76,000 and 77,000. Of these 10 transactions, HMV authorized 7 transactions.

Of the 15 transactions in the 75 two-digit combination, 13 are between 75,000 and 76,000. Like before HMV authorized 6 of the 13 transactions. In addition, there are 5 transactions for exactly 75,000. Out of these 5 transactions, 4 were payments to companies with "Cash" in the payee name. Each of these 4 transactions was paid within a few days of their invoice date, clearly a violation of company policy.



Preview Database

	SUPPNO	PAYEE	INVOICE	INV_DATE	AMOUNT	CHECK	PAY_DATE	AUTH
1	M130	Microcomputers	5863MCC	7/28/2006	7,545.77	701672	8/26/2006	VMH
2	M123	Luke Hair	51723	10/29/2006	7,582.46	701947	11/26/2006	H.M.V.
3	M100	M Cash Inc	UP-76409	10/1/2006	75,000.00	701774	10/6/2006	HMV
4	M100	Cash Inc	CS - 717 -97	9/13/2006	75,000.00	701728	9/15/2006	VST
5	M100	Co Cash Inc	T5352	10/17/2006	75,000.00	701849	10/20/2006	V.S.T
6	W007	Mr Cash Co	22 31 97	10/10/2006	75,000.00	701803	10/11/2006	VST
7	W008	Willis T. J.	JK-43-JW/97	9/5/2006	75,000.00	701770	10/1/2006	BC
8	M128	Maurice Mynah	IN-392920	8/16/2006	75,031.08	701730	9/16/2006	VMH
9	P009	O Kay Yahs	BC 469897 W	10/31/2006	75,192.58	701964	11/29/2006	HMV
10	T006	Ronnie Biggs	7899499	8/23/2006	75,243.00	701742	9/20/2006	HMV
11	D025	Denise Bent	81340	1/3/2006	75,373.66	701003	1/31/2006	BC
12	M020	Linda Hand	GR278 97	10/31/2006	75,537.32	701962	11/29/2006	HMV
13	F128	Ern Payed	82 98 1	3/4/2006	75,553.42	701200	4/1/2006	HMV
14	A128	A Meadow	117- 2393	9/5/2006	75,777.62	701773	10/1/2006	BC
15	C202	Cary S Matic	CS - 599 -97	4/8/2006	75,819.56	701287	5/5/2006	H.M.V.

7. Open the **Benford First Two Digits** database that was created as part of the analysis. Index the **DIFFERENCE** field in ascending order. Notice the large negative differences (positive spikes) in the first two digit combinations of 79, 76, and 75. These were the underlying cause of the spike in digit 7 results in the **First Digit** graph.

	DIGITS	EXPECTED	LOWBOUND	HIGHBOUND	ACTUAL	DIFFERENCE
1	79	5.45	4.05	6.85	22	-16.55
2	76	5.66	4.23	7.09	16	-10.34
3	75	5.74	4.30	7.17	15	-9.26
4	60	7.16	5.55	8.76	14	-6.84
5	50	8.57	6.82	10.33	15	-6.43
6	36	11.86	9.80	13.93	18	-6.14
7	47	9.12	7.30	10.93	15	-5.88
8	51	8.41	6.67	10.15	14	-5.59
9	57	7.53	5.88	9.18	13	-5.47
10	94	4.58	3.30	5.87	10	-5.42
11	29	14.68	12.38	16.98	20	-5.32
12	48	8.93	7.14	10.72	14	-5.07
13	62	6.93	5.35	8.51	12	-5.07
14	78	5.52	4.11	6.92	10	-4.48
15	43	9.95	8.06	11.85	14	-4.05
16	53	8.09	6.39	9.80	12	-3.91
17	70	6.14	4.66	7.63	10	-3.86
18	67	6.41	4.90	7.93	10	-3.59
19	97	4.44	3.18	5.71	8	-3.56
20	49	8.75	6.97	10.52	12	-3.25
21	74	5.81	4.37	7.26	9	-3.19
22	89	4.84	3.52	6.16	8	-3.16
23	88	4.89	3.57	6.22	8	-3.11
24	87	4.95	3.61	6.28	8	-3.05
25	86	5.01	3.66	6.35	8	-2.99
26	52	8.25	6.52	9.97	11	-2.75
27	66	6.51	4.98	8.04	9	-2.49
28	95	4.53	3.26	5.81	7	-2.47
29	77	5.59	4.17	7.01	8	-2.41

8. Because of the size of this data set, a number of transactions were identified for further investigation by using the **First Digit** test and the **First Two Digits** tests. In larger data sets, a finer filter of transactions is necessary.

> **Technical Notes**
>
> Where there are 90 possible two-digit combinations, there are 900 possible three-digit combinations from 100 to 999. The **First Three Digits** test is a highly focused test that gives the analyst relatively smaller sections due to abnormal duplication and allows for a more narrowly focused analysis. This test is also valuable to look for spikes just below internal and psychological thresholds like authorization limits. To make the most effective use of this test, the source data set should normally exceed 10,000 records.

2. Open the **Benford First Three Digits** database that was created as part of the analysis. Index the **DIFFERENCE** field in ascending order.

	DIGITS	EXPECTED	LOWBOUND	HIGHBOUND	ACTUAL	DIFFERENCE
1	750	0.58	0.12	1.03	6	-5.42
2	807	0.54	0.10	0.98	5	-4.46
3	769	0.56	0.11	1.01	5	-4.44
4	511	0.85	0.29	1.40	5	-4.15
5	187	2.31	1.40	3.22	6	-3.69
6	962	0.45	0.05	0.85	4	-3.55
7	799	0.54	0.10	0.98	4	-3.46
8	798	0.54	0.10	0.98	4	-3.46
9	533	0.81	0.27	1.35	4	-3.19
10	492	0.88	0.32	1.44	4	-3.12
11	469	0.92	0.35	1.50	4	-3.08
12	377	1.15	0.50	1.79	4	-2.85
13	186	2.32	1.41	3.24	5	-2.68
14	299	1.45	0.72	2.17	4	-2.55
15	296	1.46	0.74	2.19	4	-2.54
16	940	0.46	0.05	0.87	3	-2.54
17	295	1.47	0.74	2.19	4	-2.53
18	922	0.47	0.06	0.88	3	-2.53
19	899	0.48	0.07	0.90	3	-2.52
20	841	0.51	0.08	0.94	3	-2.49
21	795	0.54	0.10	0.99	3	-2.46
22	278	1.55	0.81	2.30	4	-2.45
23	762	0.57	0.12	1.02	3	-2.43
24	275	1.57	0.82	2.32	4	-2.43
25	749	0.58	0.12	1.03	3	-2.42
26	739	0.59	0.13	1.04	3	-2.41
27	660	0.66	0.17	1.14	3	-2.34
28	624	0.69	0.19	1.19	3	-2.31
29	615	0.70	0.20	1.21	3	-2.30

The 750 digit combination has the largest negative difference (positive spike).

Select **Window** > **Close All**.

16.2 Other Case Studies on Fraud Investigations using IDEA

Other case studies on Fraud Investigations using IDEA have been summarized below with reference links to specific chapters containing detailed write-ups.

S No	Function	Investigation Area	Reference
1	Accounts Payable	• Employees as Vendors • Favorable treatment of Vendors	Point 14.1.3 in Chapter 14.1

		• Transactions at or near Spending Authorities	
2	Accounts Receivable	• Teeming and Lading • Forced fictitious Sales to meet Sales Targets	Point 14.2.3 in Chapter 14.2
3	Inventory	• Fraudulent Inventory Valuations	Point 14.3.3 in Chapter 14.3
4	Payroll	• Payments to Ghost Employees • Unusual Overtime Payments	Point 14.4.2 in Chapter 14.4

Chapter 17

Documentation of CAATs

Documentation of CAATs: In the Planning for use of CAATs

17.0 Introduction

Data identification and its acquisition play a crucial role in the use of CAATs; these facilitate the efficient and effective use of CAATs and help realize guaranteed results and returns from its use.

The stages in Data Identification and Acquisition are

- Planning a CAAT
- Objective and assurances
- What data is available?
- The flow of data in a system
- Identifying the file for interrogation
- Specifying the information required for CAAT
- Specifying the format of the data file
- File downloading and conversion
- The Standard requirements for data
- Modes of data storage and transfer
- Who or what can help with data identification / acquisiton?

17.1 Regulatory Admissions

The Information Systems Audit and Control Association, U.S.A., has issued an IS Auditing Guideline on 'Use of Computer Assisted Audit Techniques (CAATs) Document G 3'

According to the Guideline

The major steps to be undertaken by the IS Auditor in preparing for the application of the selected CAATs are:

- Set the audit objectives of the CAATs
- Determine the accessibility and availability of the organisation's IS facilities, programs/ system and data.
- Define the procedures to be undertaken (e.g., statistical sampling, recalculation, confirmation etc.)
- Define output requirements
- Determine resource requirements, i.e., personnel, CAATs, processing environment (organisation's IS facilities or audit IS facilities)
- Obtain access to the organisation's IS facilities, programs/ system, and data, including file definitions.
- Document CAATs to be used, including objectives, high-level flowcharts and run instruction.

The documentation to be built during the stages of Data Identification and Acquisition are explained below.

17.2 Planning a CAAT

- The user needs to select the most appropriate file/ files for CAATs interrogations that best suit the objective to be tested.
- The user should make use of flow charts of the client department's system to ascertain which files are being used. It is important to select data that is as close to the original data entry process as possible.
- The flow charts and file/files selected should be documented and kept on record.
- Preferably the objectives along with the file/files selected should be documented in the manner shown below

S No	Objective	File	Form	Fields	Source	Owner	Import Method

17.3 Objectives and Assurances

- For any type of engagement it is necessary for the user to get to know the function and practices/ policies of the entity to be audited. This knowledge will help the user to plan the engagement effectively.

- For example, in a Financial Review reconciliation, High-Value item detection, exception reporting, and sampling plays a pivotal role.
- To achieve the stated objectives, a detailed knowledge of the system procedures needs to be made.
- *These documents, policy notes ,and flow charts need to be documented by the user as a part of his engagement evidence.*
- *In forensic reviews, the engagement evidence has to be irrefutable, exhaustive and unambiguous.*

17.4 What Data is available?

- The user needs to have a thorough knowledge of the operating system used by the entity being reviewed.
- The user needs to have a detailed knowledge of the application system being reviewed to identify the best possible data source and data format for each objective being reviewed.
- The user needs to engage in a meaningful discussion with the IT personnel of the entity. At times, an intermediary may be needed to accompany the user. The intermediary should have a good grasp of both audit and IT fundamentals.
- *This meeting with the IT personnel needs to be undertaken formally with a clear written agenda and with space for the IT personnel response. Any format can be followed which is convenient to the user team so long as consistency of the Agenda document is always maintained.*

17.5 The Flow of Data in the System

- The user needs to obtain from the entity Accounting Hand Books / Process Manuals, Operational Process Flow Charts to have a clear understanding of the details concerning transaction files, master files affecting (given?) lot of transactions.
- *These flow charts need to be documented with the user and clear marks should be placed on the nature of data within the system.*
- This documentation will help the user in the current engagement as well as in future cyclical engagements.

17.6 Identifying the file for interrogation

- The user must aim at selecting files which are closer to the raw data or initial input file. The study done by the user in the preceding stage will assist and facilitate in identifying such files.
- *The user may choose to liaise with the IT personnel to identify appropriate data files. This liaison may be done through email, or in person, but never orally.*
- For example, the user should know when to use a detailed transaction file, or a summarized version of a file or a table directly from the database or a customer master file. .

17.7 Specifying the information required for the CAAT

- Once the right file has been identified, it may contain information not required for the audit. At this stage, it is prudent for the user to seek specific fields for his review.
- This has performance benefits for the user as the file import will get hastened with specific fields rather than with all fields from the file.
- For example, a Data Table with 40 Million rows and 25 fields will take around 10 minutes for import within IDEA. The same table with 5-10 relevant fields will get done faster.
- *If the information to be sourced is not in the ideal format or key information is missing, it may be necessary to liaise with the IT personnel to arrange some form of file manipulation such as an embedded audit module or a file manipulation program to produce the data in a more useful format. This request to IT has to be made in writing.*

17.8 File Download and Conversion

- Once the data is readied, the data may be placed on the User's PC, or it may be placed on a shared audit folder on the Local Area Network.
- If the data is not too large, it can be sent through group office mail or even through secondary storage devices like CDs and Pen Drives.
- Data Transfer should always be accompanied by a virus scan at all times.

- Upon receiving the data, the user must sanitize the data i.e., clean the data to make it suitable to an IDEA Import compatible format.
- For example, MS-Excel files should be cleaned; the first row should reflect the header row.,There should be no blanks in the data grid, and there should be no repeating header or footer rows in the data grid.
- Once the data is finally readied, the user can deploy the best technique within the Tool for Data Import.
- For example: MS-Excel for Excel Files, SAP/AIS for AIS SAP Flat Files and more.
- *The user may initiate further technical liaison in writing with the IT to improve the file layout – record layout for future. This usually occurs when the entity changes its accounting system or migrates to a contemporary version of the application.*

17.9. Who or What can help in Data Identification?

- The user needs to identify the right resource from the IT Section and Audit Section for the CAAT project.
- This identification can be done by seeking assistance from the HR Section and the Heads of IT and Audit respectively.
- *The criteria for selection, such as the knowledge of data, knowledge of file formats, cross sectional knowledge of functions, and the like e should be documented by the user.*
- These criteria need to be provided to the HR Head, IT Head and Audit Head while making the selection.

17.10 Specimen Format of a Data Indent

To –

From –

Date –

Subject – Data Request for review of (insert area)

Pursuant to our recent meeting with respect to our planned audit of Payroll, using CAAT Data Analysis Software, we would like to arrange for download of the attached data files and fields to the FTP Server set up for the purpose.

This review involves the audit of (insert audit area) in the following entities: (insert entities). We understand your office can provide information for the following areas (insert areas). Since this is an enterprise-wide review, we will be requesting similar data from (insert organizations).

We request that each of the above entities be forwarded as separate files and that three files be created for each entity, to reflect all transactions captured by the systems for (insert audit period).

To meet our planned review objectives and timing, the data must reflect all transactions captured by the system up to and including (insert audit period).

To facilitate our CAAT analysis, the data should be given to us in the form of flat files and placed in "(provide location)" on Drive F.

Together with the data, please document the file record layouts, including the skip length, if applicable and record lengths. For each fields, document the field name, field start position, field length, data type, and formats of numeric and date items. We also require documentation of key control information, including the transaction cutoff date and time, the number of records in the file and control totals of numeric fields where applicable.

This represents our initial data request associated with this audit. We anticipate that it may be necessary to make more data requests, but we will endeavor to keep these requests to a minimum.

We appreciate your assistance with data access, and look forward to working with you during the course of our audit.

If you have any questions please do not hesitate to contact me (**********), or at.

17.11 Specimen Format of a CAATs Working Paper

1.	Client and Location	-
2.	Review Period	-
3.	Audit Area	-
4.	Work Program	-
5.	Name of Auditor and Team Leader	-
6.	CAAT performed on	-

7.	Documentation of Source Data		-
	• Source Reports		-
	• Data File Names and Type	-	
	• Data Fields relevant for CAAT	-	
	• Data Field Name, Type and Length	-	
8.	Input Reconciliation		-
9.	Details of CAAT analysis performed	-	
10.	CAAT Results		-
11.	Conclusion and Recommendations	-	

Chapter 18

Continuous Auditing Using IDEA

Continuous Auditing (CA) with IDEA

18.0 Introduction

There is a thin conceptual line between Continuous Monitoring and Continuous Auditing.

“Continuous monitoring refers to the management’s responsibility to assess the adequacy and effectiveness of controls in an organization. Continuous Assurance is when auditors perform continuous control and risk assessment (i.e. continuous auditing) and evaluate the adequacy of management’s continuous monitoring activities.”

Continuous auditing changes the nature of evidence, timing, procedures and effort involved in modern day audits.

Transactional verification, compliance verification, estimate verification and judgment verification are the four levels of continuous audit.

18.1 Benefits of Continuous Auditing

Continuous Control Assessment provides financial executives, business process managers, and risk and compliance officers with independent and timely assurance over real time controls. Intelligent technology –enabled analytics like transaction validations and analytical reviews accompanied by drill down capabilities help discover the cause of the control failures of organizations.

Some of the key benefits of CA are:-

- Less Travel as CAATs can be deployed centrally on central data reservoirs.
- Low Staff and allied audit resource requirement.
- Low IT Hardware resource utilization like network and bandwidth as tasks can be scheduled after office hours, on holidays and weekends.
- High user satisfaction as automated tasks fired in non-peak hours give quicker results than queries fired in peak business hours.

- User prestige and recognition as CCM is the final mature stage in the use of CAATs, which is preached rather than practiced worldwide. A true CCM user gains unimaginable benefits and recognition in professional peer circles.
- Analytic monitors like Embedded Audit Modules and Statistical Tools observe material events, trigger alarms, drill down to grass root levels, facilitate data integration across multiple platforms and allow repetitive tests at much lower costs.

18.2 Implementing Continuous Auditing

Planning

- Defining the objectives for Continuous Auditing
- Obtaining Senior Management Support
- Ascertaining the degree to which Management is performing its Role
- Identifying and prioritizing areas to be addressed
- Performing a type of continuous auditing Identifying key information systems
- Understanding the underlying business processes and application systems

Executing

- Selecting and purchasing analysis tools
- Developing and maintaining auditor analysis skills and techniques
- Cleansing and preparing the Data
- Assessing the data integrity and reliability
- Indentifying crucial control points, and control rules
- Designing a technology-assisted approach to test controls
- Identifying deficiencies

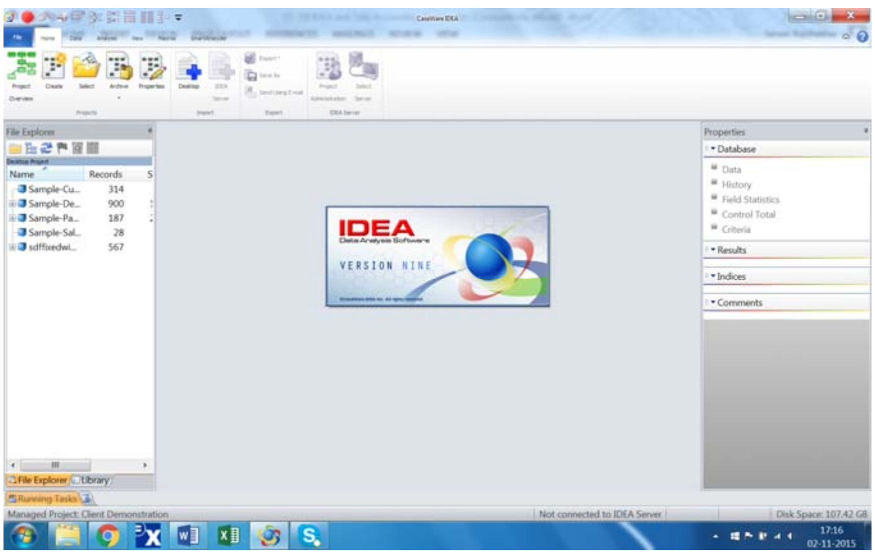
Reporting

- Identifying control deficiencies
- Identifying increased levels of risk

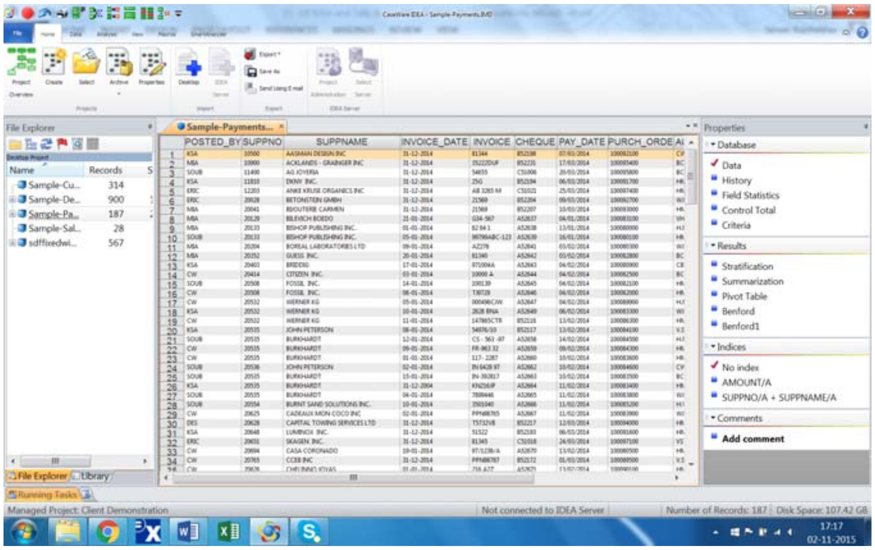
- Prioritizing results
- Initiating suitable audit response
- Making results known to Management
- Managing results by tracking
- Reporting
- Monitoring
- Providing follow-up
- Evaluating results of actions taken.

18.3 Sample Case Study on Continuous Auditing using IDEA

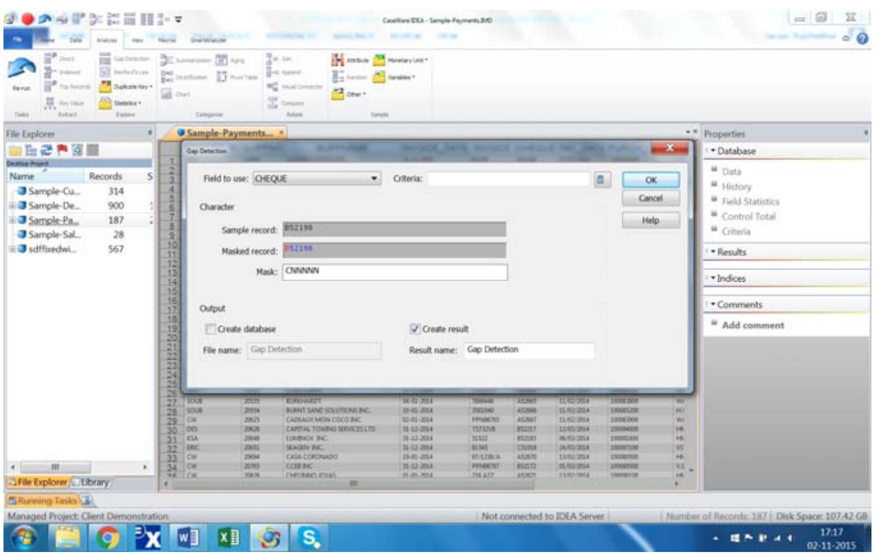
18.3.1 Start IDEA



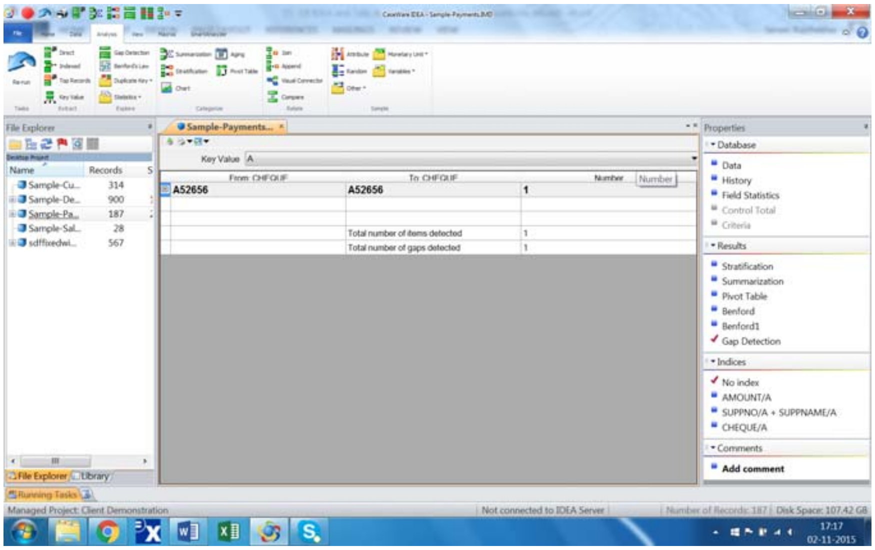
18.3.2 Record Macro (Control R)



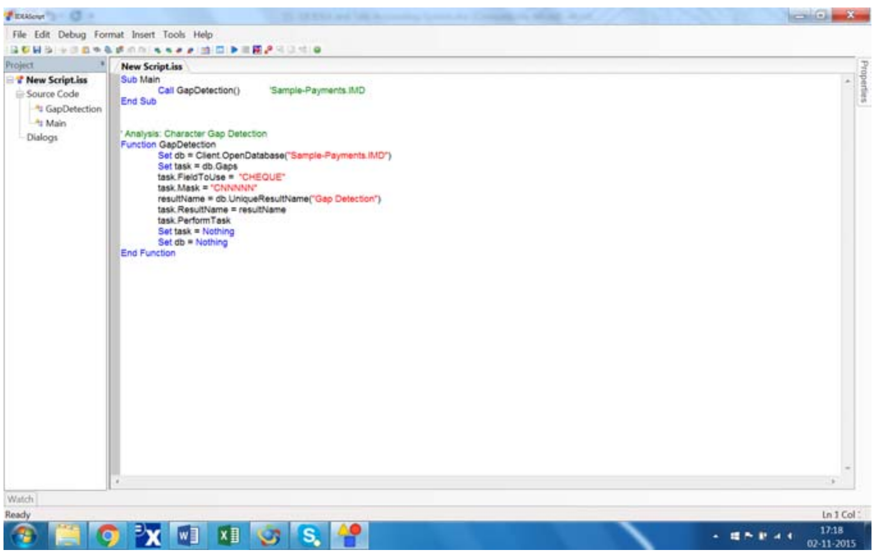
18.3.3 Run sample IDEA Function - Gap Detection on Cheques as per the Objective



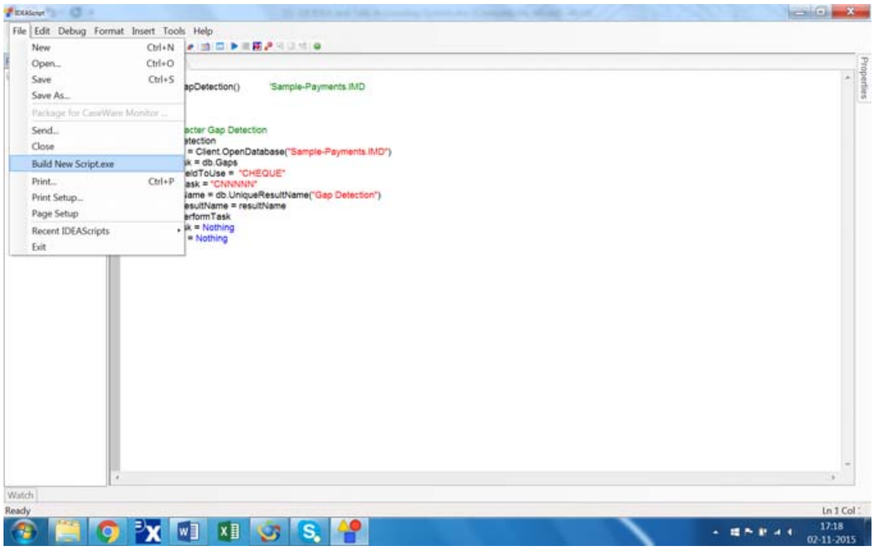
18.3.4 Arrive at Result and Review in terms of the Objective. Follow Up



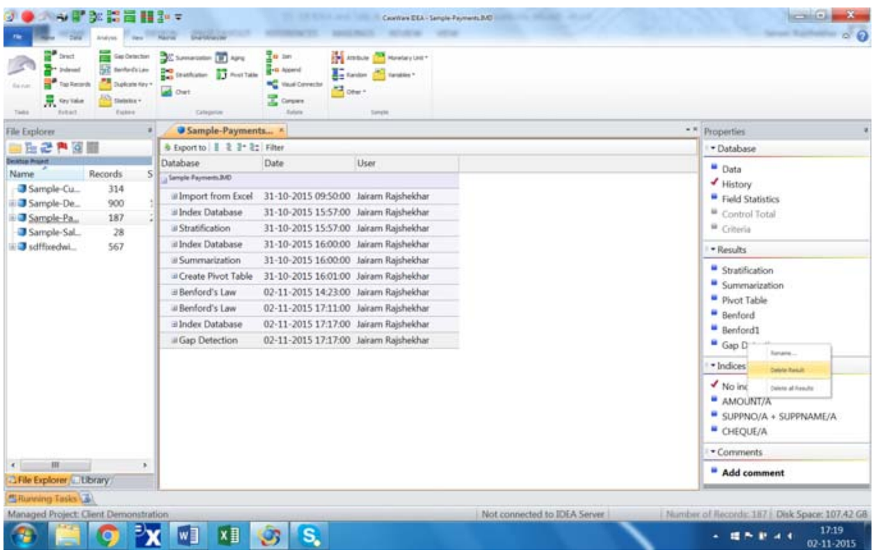
18.3.5 Stop recording Macro. IDEA Script for Function gets generated



18.3.6 Save IDEA Script as EXE File

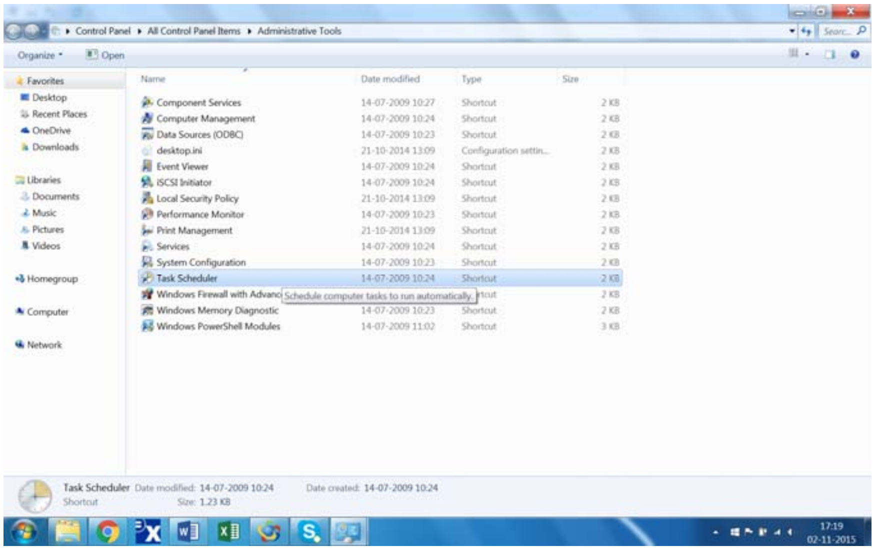


18.3.7 After Review of Result, delete or archive

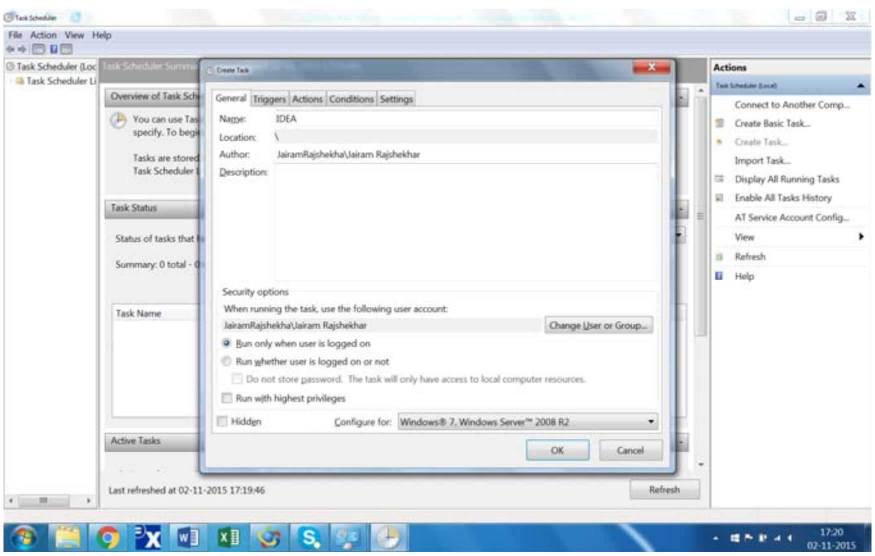


18.3.8 Close IDEA

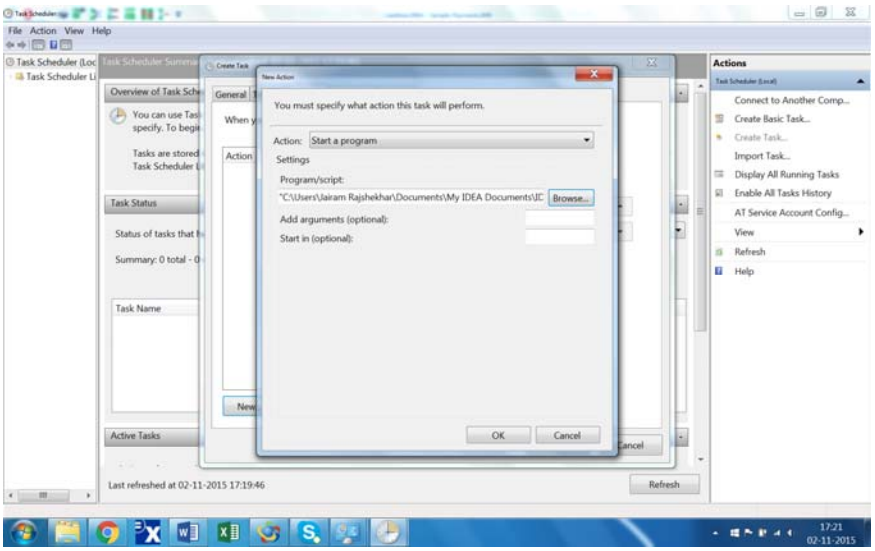
18.3.9 Go To Task Scheduler in Control Panel



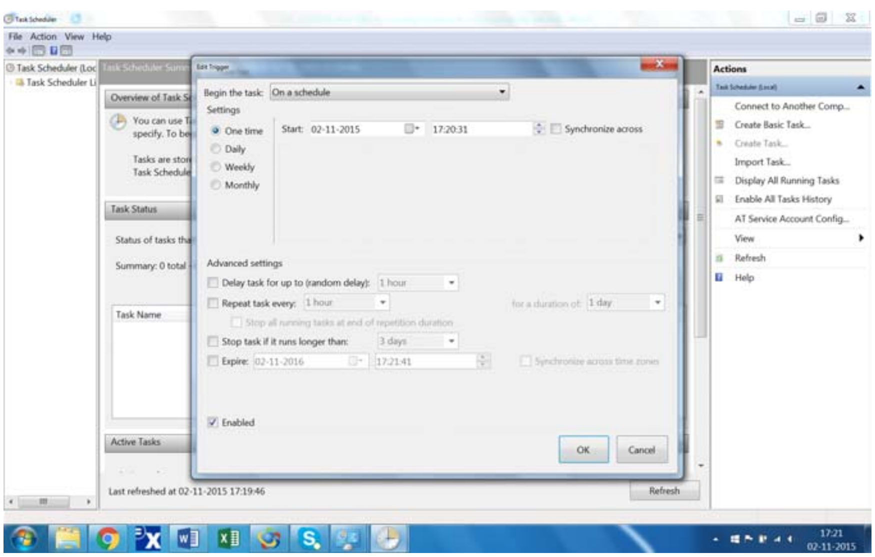
18.3.10 Add Scheduled Task



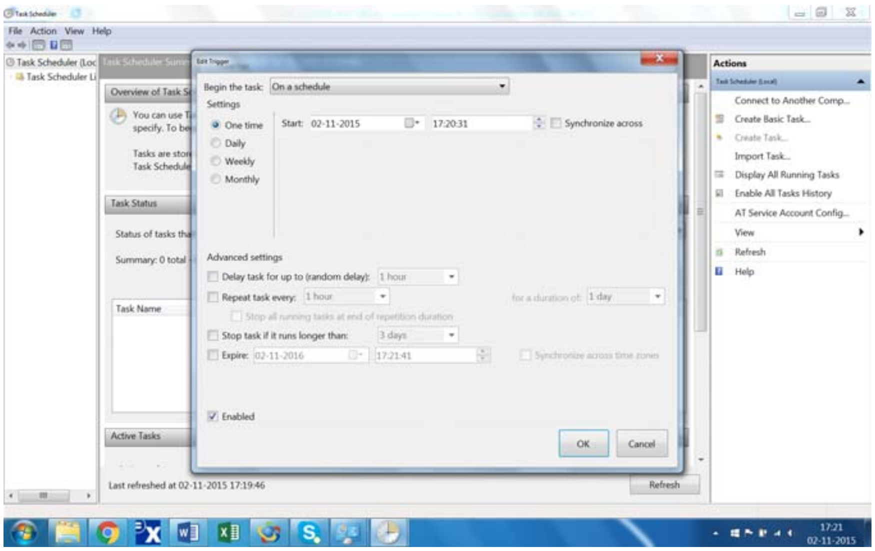
18.3.11 Select the program we want Windows to Run – IDEA



18.3.12 Decide on the frequency of execution and the name Task



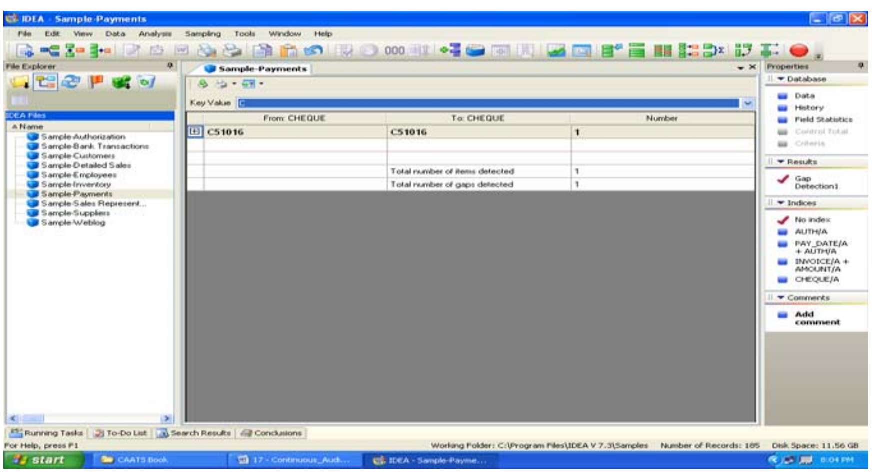
18.3.13 Set Date and Time for execution of the Task

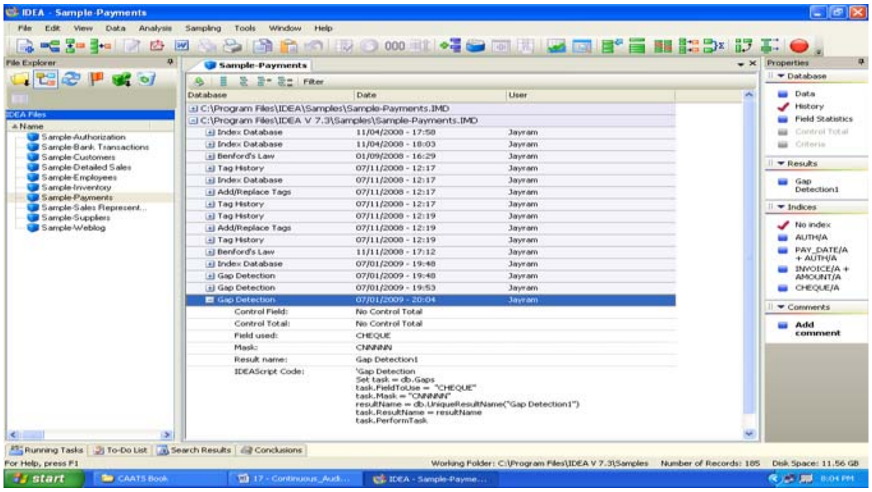


18.3.16 Click Apply and OK

18.3.18 The Task will run automatically at the scheduled date and time

18.3.18 Review the new result, follow up and review.




IDEA - Sample-Payments
File Edit View Data Analysis Sampling Tools Window Help
File Explorer
Sample-Authorization
Sample-Bank Transactions
Sample-Customers
Sample-Detailed Sales
Sample-Employees
Sample-Inventory
Sample-Payments
Sample-Sales Represent...
Sample-Suppliers
Sample-Weblog
Sample-Payments
Filter
Database Date User
C:\Program Files\IDEA\Samples\Sample-Payments.IMD
C:\Program Files\IDEA V 7.3\Samples\Sample-Payments.IMD
Index Database 11/04/2008 - 17:58 Jayram
Index Database 11/04/2008 - 18:03 Jayram
Benford's Law 01/09/2008 - 16:29 Jayram
Tag History 07/11/2008 - 12:17 Jayram
Index Database 07/11/2008 - 12:17 Jayram
Add/Replace Tags 07/11/2008 - 12:17 Jayram
Tag History 07/11/2008 - 12:17 Jayram
Tag History 07/11/2008 - 12:19 Jayram
Add/Replace Tags 07/11/2008 - 12:19 Jayram
Tag History 07/11/2008 - 12:19 Jayram
Benford's Law 11/11/2008 - 17:12 Jayram
Index Database 07/01/2009 - 19:48 Jayram
Gap Detection 07/01/2009 - 19:48 Jayram
Gap Detection 07/01/2009 - 19:53 Jayram
Gap Detection 07/01/2009 - 20:04 Jayram
Control Field: No Control Total
Control Total: No Control Total
Field used: CHEQUE
Mask: CNNNNN
Result name: Gap Detection1
IDEAScript Code: 'Gap Detection
Set task = db.Gaps
task.FieldToUse = "CHEQUE"
task.Mask = "CNNNNN"
resultName = db.UniqueResultName("Gap Detection1")
task.ResultName = resultName
task.PerformTask
Properties
Database
Data
History
Field Statistics
Results
Gap Detection1
Indices
No index
AUTH/A
PAY_DATE/A + AUTH/A
INVOICE/A + AMOUNT/A
CHEQUE/A
Comments
Add comment
Running Tasks
To-Do List
Search Results
Conclusions
For Help, press F1
Working Folder: C:\Program Files\IDEA V 7.3\Samples
start
CAATS Book
17 - Continuous_Audi...
IDEA - Sample-Payme...

Chapter 19

Advanced Statistical Methods in IDEA

19.0 Introduction

IDEA's Advanced Statistical Methods can be used by analysts as a component of a continuous monitoring set of routines.

This chapter covers the use of two key statistical methods in IDEA and provides examples of how these methods can be used in a wide variety of settings:

- Correlation
- Trend Analysis

While calculations for these statistical methods can be done by other software programs that havestatistical routines including linear regression, or by using the formula for the Pearson product momentcorrelation, or the formulas found in statistics books for the slope and intercept, the unique feature inIDEA is that these calculations can be done on a group-by-group basis.

In IDEA's Advanced Statistical Methods module, a group is referred to as an audit unit. While the termaudit unit may be used to mean location, division, franchisee, or some other reporting entity, the testscould be run for a task that is not an audit as such, but perhaps an analysis to screen for investmentprospects, or to screen for significant data errors, or to report results for a professional or academicresearch study.

19.1 Correlation

Correlation is a statistical technique being used by auditors and other data analysts to compare thepatterns in numerical values in two data sets. A high level of correlation (that is, a correlation scoreclose to 1) means that high values in one data set are matched with high values in the second dataset, and conversely, low values in one data set are matched with low values in the second data set.

The correlation score is a single number that tells us how good this match is. Correlation scores canrange from -1 to +1.

Correlation was first used in auditing as a component of a continuous monitoring set of audit routines. For example, the average level of sales for each month of the year were calculated for therestaurants owned and franchised by a company, and then the correlation between each individual restaurant's sales and the average sales was calculated. A low correlation score for any restaurantshowed that the seasonal pattern for that restaurant deviated from the norm. Another example wasthe analysis of election data where it was believed that each candidate should receive the samepercentage of votes in each location. In a simplified setting assume that candidate #1 received 60percent of the votes, candidate #2 received 30 percent of the votes and candidate #3 received 10percent of the votes in an election. Correlation was used to measure how well this 60:30:10 ratioapplied to each voting location. Low correlations for any location suggested that (a) either the candidates were more or less well favored in those locations, or (b) that there was an error in thevote counts.

The applications of correlation are not limited to these types of data sets but can also be used tocompare both monthly and annual financial reports to those of prior periods. In fact, the techniquecan be used whenever a set of numbers must be compared to some norm. For example, theelectricity consumption of all schools in a school district could be compared to the normal (average)pattern, or the bad debt write-offs on a monthly basis could be compared for each branch in a bankto the average pattern for the year.

A database output or Result output could be generated from the Correlation analysis. The Resultoutput can be viewed in a grid or as a graph. The graph would display the correlation scores rankedfrom smallest to largest allowing one to see the profile of the correlation scores (that is, theproportions of low and high scoring audit units). The graphs can be further customized to suit thetask at hand.

15.1.1 Correlation Case Study

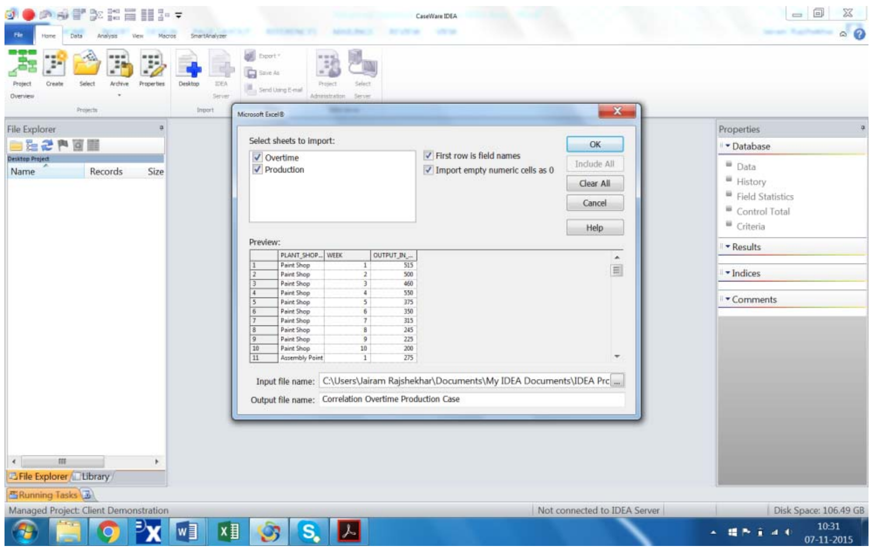
Background

An internal auditor engaged by a Manufacturing Company to review the overtime payments to staff and workers has been provided with two excel files – one – plant shop centre, week wise overtime and the other – plant shop centre, week wise production.

By linking the two files together in IDEA, the auditor would like to employ Correlation to check for specific plant shop centres where production is on a decline whereas overtime is on a rise. Correlation is an effective visualization technique to identify high risk audit units in a population. So in the given example we would like to identify high risk plant shop centres from an overtime payment control point of view amongst all plant shop centres in the Company.

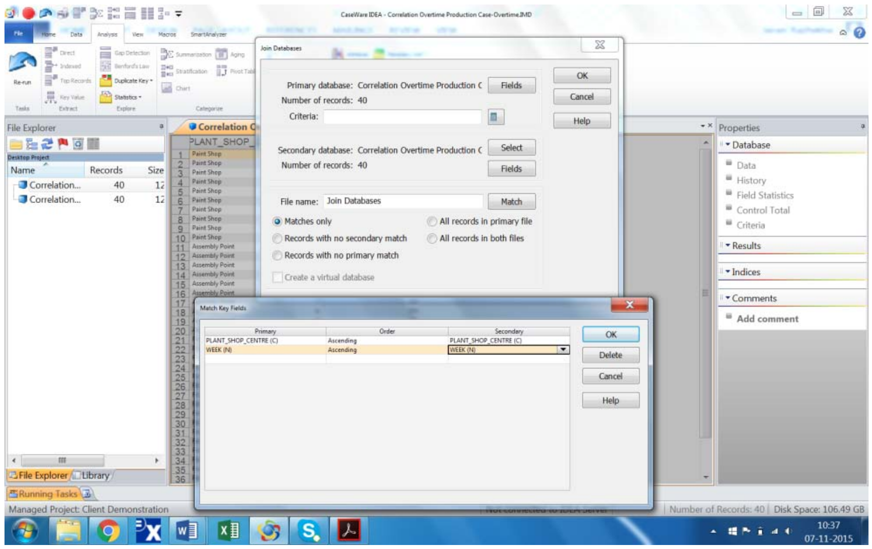
Step 1 – Import of Data Files

Import the two excel worksheets into IDEA using Home – Desktop – Import Assistant



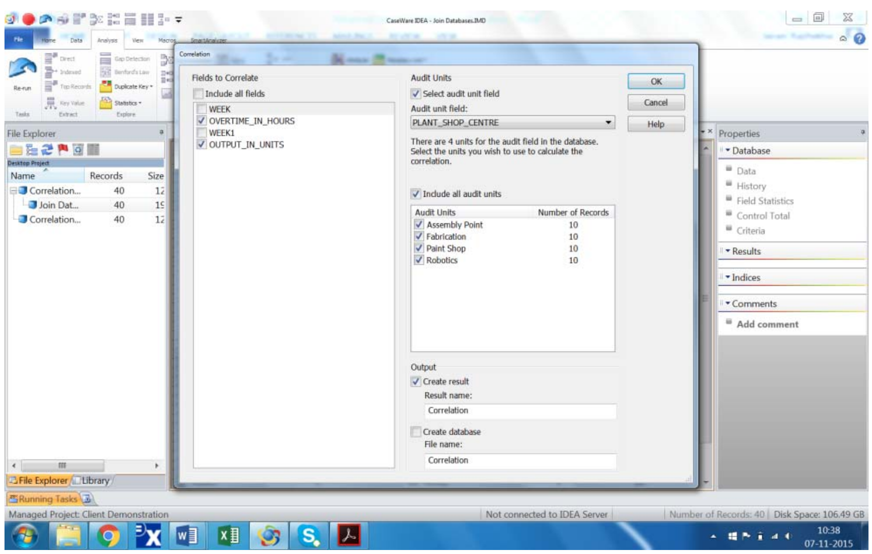
Step 2 – Join the Data Files

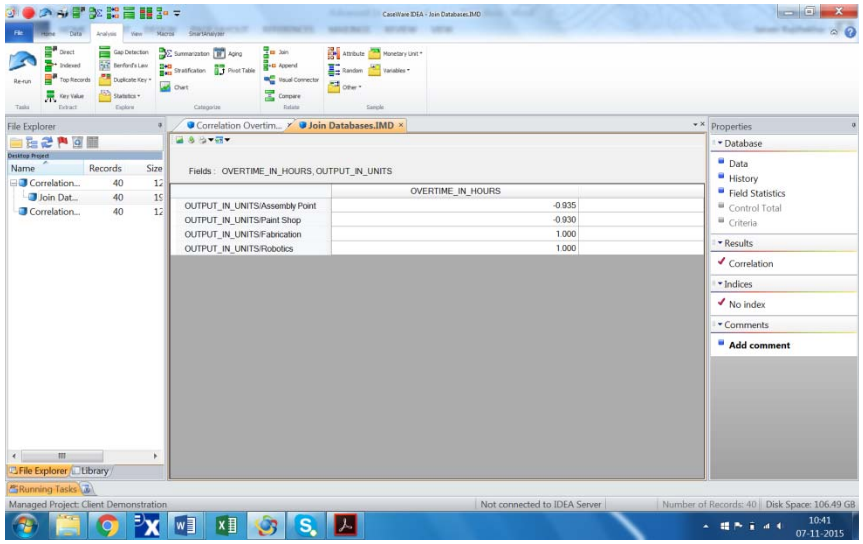
Open the Overtime file as the Primary database in IDEA. Click on Analysis – Relate and Join. Select the Production file as the Secondary database in the Join dialog box. Enter the matching key fields as Plant Shop Centre and Week between the Primary and Secondary files.



Step 3 – Apply the Correlation

Click on Analysis – Explore – Statistics and Correlation. Select the Overtime in Hours and Output in Units as Fields to Correlate. Select Plant Shop Centre as the Audit Units. This will give correlation scores for each Plant Shop Centre.

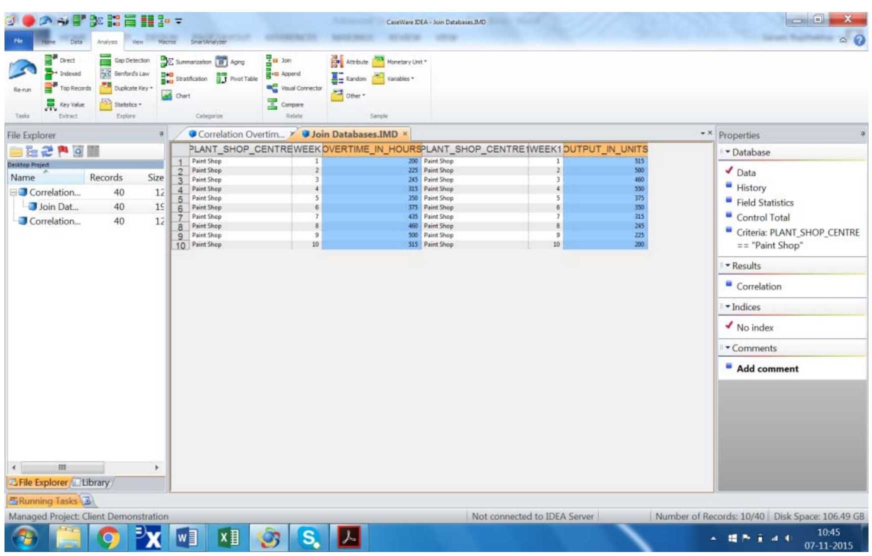




Conclusion

The Correlation Score for Paint Shop is -0.930 indicating an inverse relationship between the Overtime in Hours and Output in Units.

A closer look at the weekly overtime and output figures for Paint Shop will reveal a falling output from week 1 to week 10 with a rising overtime for the same period. This would allow the internal auditor to focus his/her attention on reviewing, verifying, investigating the detailed overtime payments for the Paint Shop from week 1 to 10.



19.2 Trend Analysis

Trend Analysis is a statistical technique that is being used by auditors and other data analysts tocalculate the trend of data over time and to forecast values into the future on the assumption thatthe trending pattern will continue. The Trend Analysis technique used in IDEA's Advanced StatisticalMethods module is based on linear regression using the method of least squares. This technique willfit the best fitting straight line to the data and will forecast ahead using this straight line. If the data has a seasonal component (higher values associated perhaps with certain months, or days of theweek, or times in the day) then the Time Series statistical method should be used instead of Trend Analysis.

Trend Analysis can be used in auditing as a component of a continuous monitoring set of audit routines. For example, it could be used to test for unintentional or intentional errors in the reportedsales numbers of all the restaurants owned and franchised by a company. One possible indicator oferrors could be a decreasing sales trend for a restaurant when the average, over all locations, showsan increase in sales. Trend Analysis (or linear regression) could be used to quantify the trend in thesales for any location. A negative trend would be an indicator of possible problems with the salesnumbers.

Another use for Trend Analysis could be to simply provide a forecast for various accounting entitiesin a continuous monitoring environment. The forecasts could then be compared to the actual numbers and where the forecasts differed from the actual numbers, it would be an indication that these numbers have deviated from their past trend suggesting that some change has taken place.

Trend Analysis can provide the answers to many types of monitoring questions such as:

- For a franchisor, which locations have decreases in sales compared to the amount extrapolated from past trends?
- For an investment analyst, which investments show a decreasing trend in market values?
- For a school district, which schools show a supplies usage that exceeds the amount extrapolated from past trends?
- For a cruise ship or hospital, which ships or departments show a high level or an increasing trend in employee sick days?
- For a courier service, which locations show the largest increases in fuel expenses?

- For a college, which departments show decreasing numbers of majors?

15.2.1 Trend Analysis Case Study

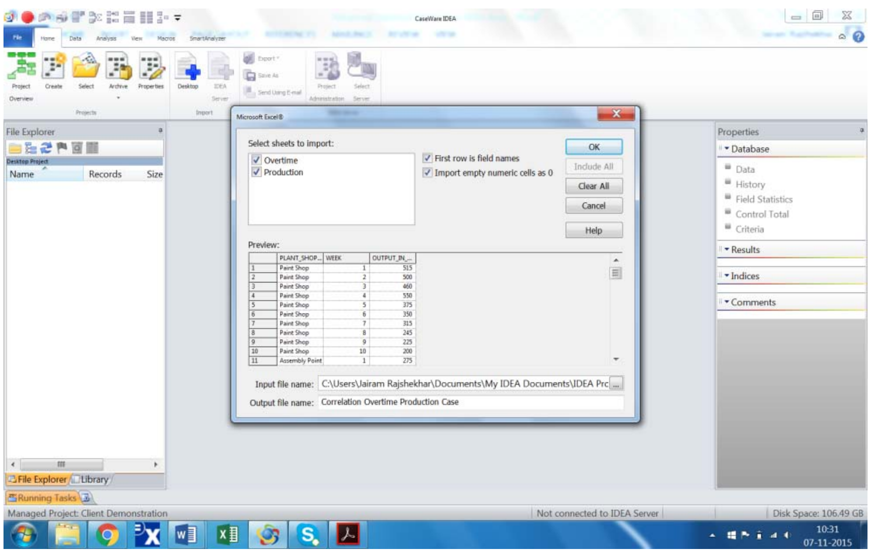
Background

An internal auditor engaged by a Manufacturing Company to review the overtime payments to staff and workers has been provided with two excel files – one – plant shop centre, week wise overtime and the other – plant shop centre, week wise production. The historical data in both the files is for 10 weeks of the year.

By linking the two files together in IDEA, the auditor would like to employ Trend Analysis to check for the 11th week forecast for each plant shop centre. This will allow the auditor to have a ready comparison between the 11th week forecast and 11th week actuals during the course of the audit.

Step 1 – Import of Data Files

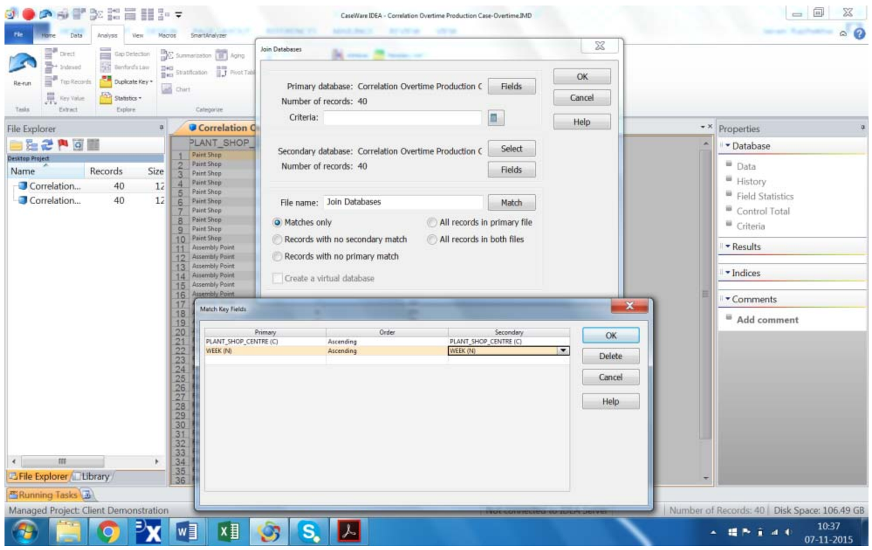
Import the two excel worksheets into IDEA using Home – Desktop – Import Assistant



Step 2 – Join the Data Files

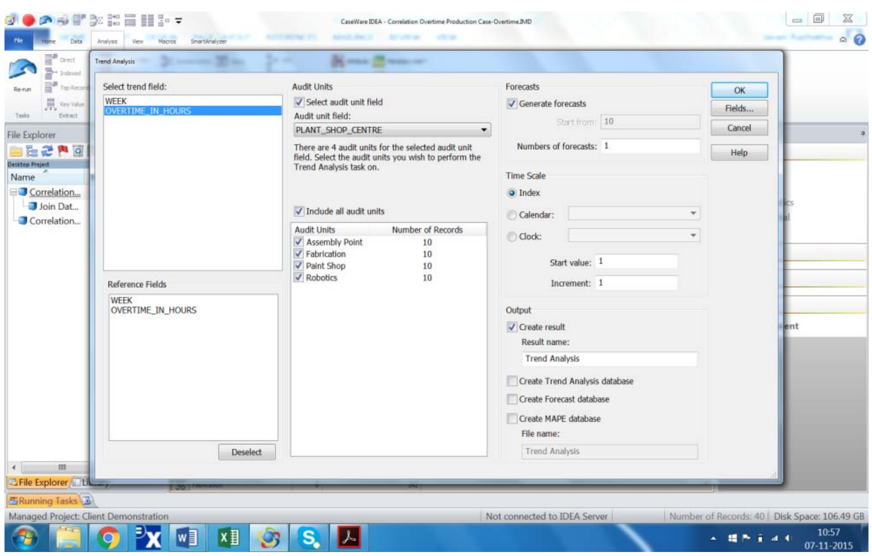
Open the Overtime file as the Primary database in IDEA. Click on Analysis – Relate and Join. Select the Production file as the Secondary database in the Join dialog box. Enter the matching key fields as Plant Shop Centre and

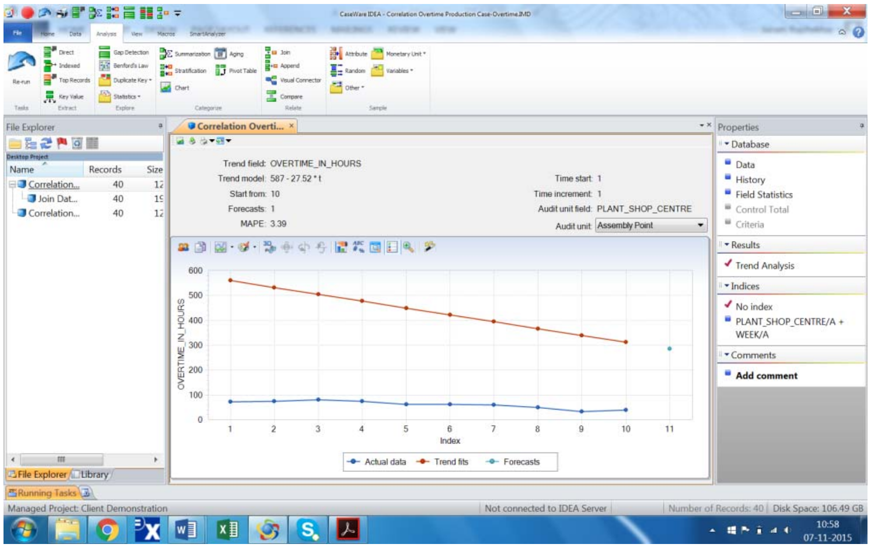
Week between the Primary and Secondary files.

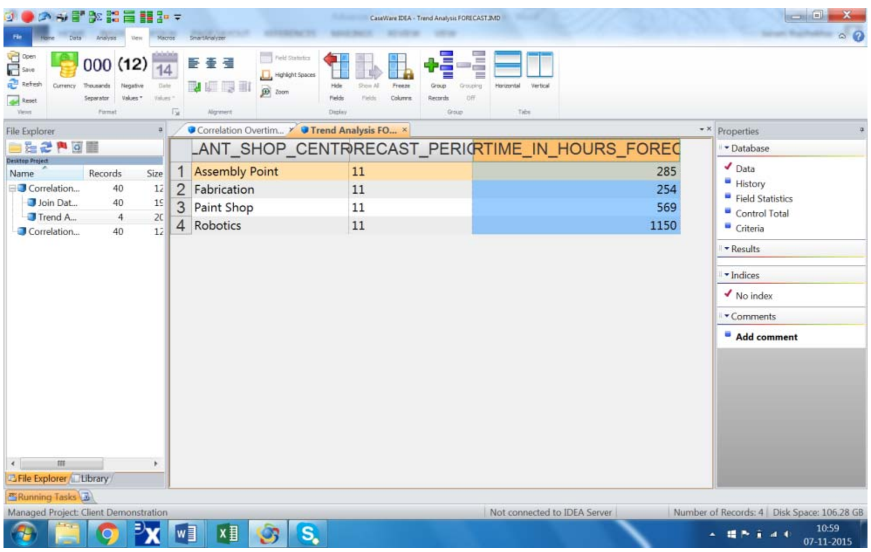


Step 3 – Apply the Trend Analysis

Click on Analysis – Explore – Statistics and Trend Analysis. Select the Overtime in Hours as the Trend Field. Select Plant Shop Centre as the Audit Units. Click on Generate Forecasts for 11th week. This will give the trend forecast for the 11th week for each Plant Shop Centre.




Trend field: OVERTIME_IN_HOURS
Trend model: 587 - 27.52 * t
Start from: 10
Forecasts: 1
MAPE: 3.39
Time start: 1
Time increment: 1
Audit unit field: PLANT_SHOP_CENTRE
Audit unit: Assembly Point
OVERTIME_IN_HOURS
Index
Actual data
Trend fits
Forecasts


LANT_SHOP_CENTR
RECAST_PERIO
RTIME_IN_HOURS_FOREC
1 Assembly Point 11 285
2 Fabrication 11 254
3 Paint Shop 11 569
4 Robotics 11 1150

Chapter 20

Do’s and Don’ts while using IDEA

This Chapter contains our collective practical suggestions for ‘Do’s and Don’ts’ with our decade plus experience and implementation learnings with IDEA –

S No	Function in IDEA	Usage Tips
1	Set Working Folder	Once you create a new folder, select the new folder and click on open. This will ensure you work inside the folder.
2	Import Assistant – Microsoft Excel	• Ensure that the first row in the MS-Excel file contains the field headers. • Ensure that the file is closed prior to import. • Check both the boxes ‘First Row Is Field Names’ and ‘Import Empty Numeric Cells as Zero’.
3	Import Assistant – Print Report and Adobe PDF	When selecting a date field in the field editor (Report Reader) change field type from character to date and set mask as DD-MMM-YY if the date is like 02-MAR-15. Even though the date mask has been set as DD-MMM-YY, the date field will get displayed in the IDEA database based on the regional date settings on the laptop/PC which has IDEA. So if the laptop/PC has the regional date settings as MM-DD-YYYY, even though we set the mask in the Report Reader as DD-MMM-YY, the date will get displayed in the IDEA database as MM-DD-YYYY. While writing a date equation on this field you will always use DATE=”YYYYMMDD” in the Equation Editor.
4	Index	Double-click on the field of choice once to sort on ascending basis i.e. Low to High.

S No	Function in IDEA	Usage Tips
		Double-click once again to sort on descending basis i.e. High to Low. This will only create a different view of the original database.
5	Direct Extraction	• While entering an equation in the Equation Editor please ensure that every condition has a field, arithmetical operator and value/field. Eg. PASS_AMT>BILL_AMT • Multiple conditions to be connected with a logical operator like (.AND. .OR. .NOT.) Eg. (PASS_AMT>BILL_AMT) .OR. (TOKEN_DATE>CHEQUE_DATE) • Character condition strings to be placed in text encapsulates. Eg. NAME== "ABCD" • Date condition values to be place in text encapsulates. Eg. CHEQUE_DATE ="20150126". • Enter the values/strings correctly depending on how the data is appearing in the database. Do not leave any leading space in the text encapsulates. Eg. SCHEME=" NREGA" is wrong. This will look for NREGA with a leading space in the database. SCHEME="NREGA" is the right way. • Save equations in the Equation Editor for re-use later. • If you get an error message while writing an Equation – "Incorrect Syntax" – please check whether you have captured the field name correctly. Eg. TOKEN_NO = "" is right. TOK_NO = "" will give "Incorrect Syntax". • If you get an error message while writing an Equation – "Mismatch Field

S No	Function in IDEA	Usage Tips
		Type” – please check the compatibility of the field and value in a given condition. Eg. CITY=”DELHI” is right. CITY=50000 is wrong and will give “Mismatch Field Type”
5	Criteria	After you enter a criteria under Properties (right-side of IDEA Screen), IDEA will give you a view of the original database. Click on File-Save As to save the view as an independent file. After File-Save As, visit the original database again, right click on Criteria and clear.
6	Duplicate Key – Detection	To be used when you are looking for duplicates in one or more field/s up to 8 fields. Eg. Same Controller, Same DDO, Same PAO, Same Function Head, Same Scheme, Same Agency Name, Same Sanction Date
7	Duplicate Key – Exclusion	To be used when you are looking for duplicates in one or more field/s up to 8 fields where an independent field is different. Eg. Same PPO Number Different Pensioner Name in Central Pension Accounting.
8	Gap Detection	Can be used to look for missing items only on fixed length/formatted numeric, date or character fields. Eg. If you are looking for missing cheque numbers in a cheque series where multiple cheque books are being used. Say 123456, 123457, 123459 and 256787, 256788 and 256789. Select cheque number as Fields to Use in the Gap Detection screen and set the mask as CCCNNN. Cheque number should be set as a character field using Field Manipulation. This will give missing cheques per cheque book series only and

S No	Function in IDEA	Usage Tips
		not within the cheque books which is invalid.
9	Field Manipulation	While appending a new field to any database in IDEA, append a virtual field as it allows for the equation to be changed at any time.
10	Summarization	• Used for arriving at a count/frequency/number of times. By selecting a field in 'numeric fields to total' on the right hand side of the summarization screen, you will get sum totals as well along with counts. • When you have a database where the relationship between values like say GOODS RECEIVED AMOUNT and BILL AMOUNT are not one to one. For example it may be many goods received to many bills. In such a case summarize the data by Purchase Order Number or Token No and choose numeric fields to total as GOODS RECEIVED AMOUNT and BILL AMOUNT. Once you get the summary you can apply a direct extraction equation to check for BILL AMT SUM>GOODS RECEIVED SUM. This is not required where there is a one-to-one relation though.
11	Stratification and Stratified Random Sampling	Before you perform a Stratification, visit Field Statistics and observe the MIN, MAX and AVG statistics for the field on which stratification is going to be performed. This will help you in interval creation in the Stratification dialog box. Eg. MIN is Rs. 1000, AVG is Rs. 50000 and MAX is Rs. 500000 for a field TAX_AMOUNT. While performing Stratification create intervals as 0-50000, 50000-100000. Anything above

S No	Function in IDEA	Usage Tips
		Rs. 100000 will be an Upper Limit Exception (ULE). The ULE cases are A Class cases i.e. High Value. Click on the number of records in ULE in the Stratification result and extract the records to a file titled ‘Class A’. Now you can perform a random sample on the ‘Class A’ file.
12	Top Records Extraction	• To be used to identify ‘top records’ or ‘bottom records’ in any database. Eg. Identify top 25 assesses in the tax collection report. Enter 25 as number of records to extract and perform top records for field ‘TAX_AMOUNT’. • If your objective now required you to capture top 25 assesses per ward, then follow the same method as above. In Addition choose ‘WARD’ as the field to Group.
13	JOIN	• While joining a primary and secondary database using File – Join in IDEA, ensure that both the databases have atleast one common key field. The key field need not have the same name, but it must have the same field type like ‘TOKEN_NO’ (Character) and ‘TOKEN_NUMBER’ (Character). • The Matches only option in the Join, matches the first match from the secondary database always. • If you have a situation where both the primary and secondary database have multiple records per key field. Eg. TOKEN_NO in BILL FILE has multiple records per TOKEN NUMBER and TOKEN_NUMBER in CHEQUE FILE has multiple records per TOKEN NUMBER, then none of the options in

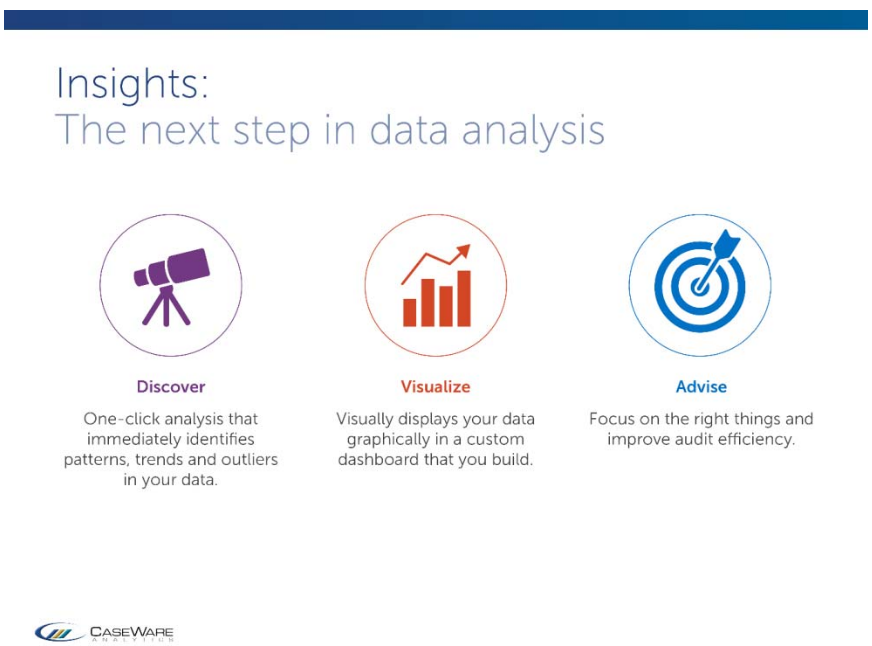
S No	Function in IDEA	Usage Tips
		Join will help you compare the BILL with CHEQUE details since it will only pick up the first matching cheque from the CHEQUE FILE. In such a case use COMPARE under File. COMPARE will summarize the multiple records per TOKEN NO in both the primary and secondary databases and you can choose to compare BILL AMOUNT with CHEQUE AMOUNT. • If you have one transaction file like BILLS and another master file like Delegation of Financial Powers. Make the BILLS file the Primary database. Always choose the transaction file as the Primary database and apply the master details to the transaction file. • If you have two transaction files like Pension File April 2014 and Pension File March 2015, any can be chosen as Primary. But we take Pension April 2014 as Primary given it is earlier in time period than March 2015.
14	Edit – Re-Run Task	• Always use Edit-Re-Run Task to correct a mistake made or to re-run the task last performed on any database active on your screen. • While performing functions for new objectives make sure you go back to the Original Database to avoid performing a new objective on the result of the earlier objective.

Chapter 21

New and Emerging Features in Data Analysis

"More than half of chief audit executives and directors believe that continuously assessing and identifying emerging risks will be the greatest challenge for internal audit throughout 2015 - but only one-third are highly confident in their ability to identify these risks."

— The Institute of Internal Auditors Audit Executive Center. March 2015. The Pulse of Internal Audit: Navigating an Increasingly Volatile Risk Environment. Retrieved from https://na.theiia.org/news/Pages/2015-North-American-Pulse-of-Internal-Audit-Released.aspx

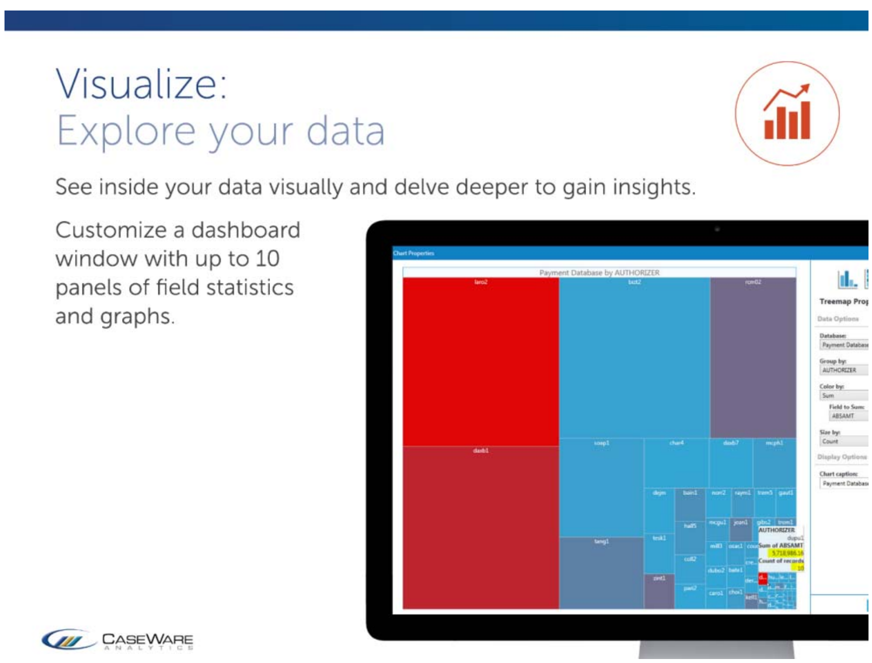


21.1 Visualization – Know What's in Your Data – Next Evolution in Data Analysis

The Visualize Task IDEA 10 introduces the new Visualize task, a user-friendly visualization tool available through the Analytics tab. Working from one or even multiple databases, Visualize allows you to build customized visualization dashboards so you can see the outliers, distributions and trends in your data. Your Dashboards can then be saved and shared through the Library. Visualize also provides the ability to drill down with a grid view of your data and extract insights from any particular slice, bar, column or area of a chart. Additionally, the new Auto-Stratification feature stratifies on numeric and date fields by automatically setting appropriate ranges within your graphs and charts.

Visualization unlocks IDEA's Analytic Intelligence so you can forget the days of painstakingly scanning through a series of tables. Visualization's tasks automatically create charts and field statistics, profiling your data so you can quickly pinpoint patterns, trends, outliers and correlations that may have gone unnoticed in a table. Raising the bar even higher, Visualization displays data graphically in up to 10 panels within a dashboard that you can tailor to show exactly the results you want. All the information you need to prepare a more focused audit.

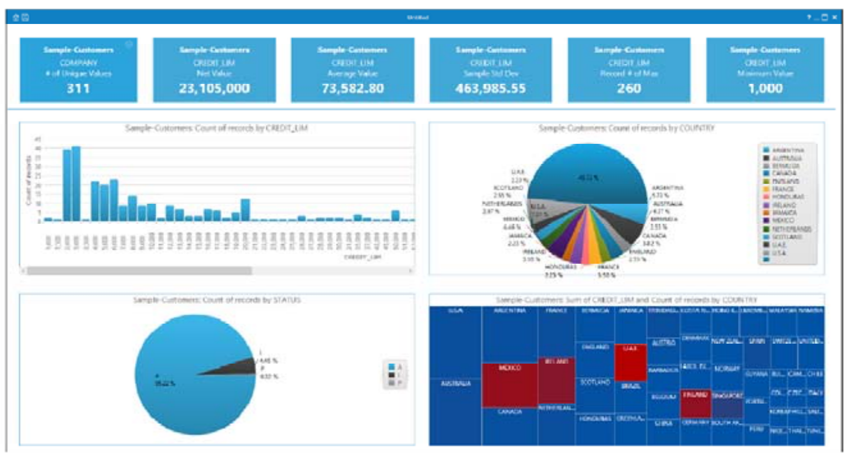
The screen – shot below depicts Visualization in IDEA.



21.2 Discover

Discover unlocks the power of IDEA's Analytic Intelligence. This proprietary intelligence, made up of pre-written algorithms, identifies trends, patterns or anomalies in the data that deserve investigation. Discover automatically presents charts or field statistics on a dashboard visually displaying areas of interest and anomalies that may have been hidden in the data. Any dashboard created through Discover can be modified, saved and shared with others through the Library. In addition, Discover allows you to identify key fields within a database and extract the key field statistical information you need. Based on the sound statistical method called Inter-Quartile Range (IQR), you can also identify outlier distribution and flag areas of interests in a database on numeric and date fields.

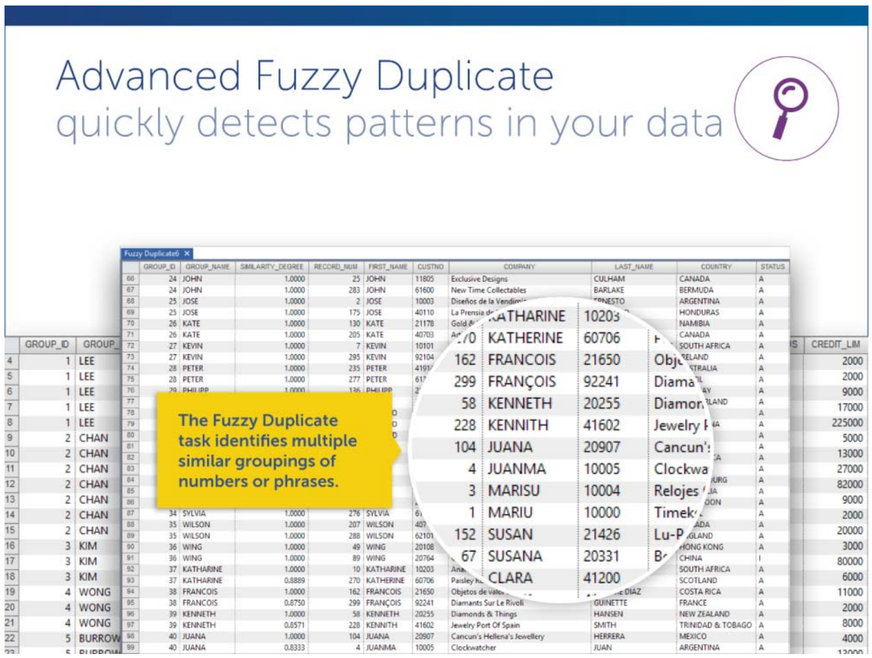
The screen – shot below depicts Discover in IDEA.



21.3 Advanced Fuzzy Duplicates

The Advanced Fuzzy Duplicate tool is more powerful than ever, identifying multiple similar records using an unheard of three Character fields. Highlighting similar records and then grouping them based on the degree of similarity, this tool does your work for you, detecting data entry errors, multiple data conventions for recording information, and fraud. You can even fine-tune your data groups based on case sensitivity.

The screen – shot below depicts Advanced Fuzzy Duplicates in IDEA.



APPENDICES

Appendix I

Illustrative List of Computer Assisted Audit Tools (CAATs)

1. IDEA Data Analysis Software (*www.caseware-idea.com*)
2. Methodware Risk Management and Internal Auditing Software (*www.methodware.com*)
3. TeamMate Audit Management System (*http://tax.cchgroup.com/TeamMate/default*)
4. Approva Controls Intelligence Software (*www.approva.net*)
5. Caseware Working Papers (*www.caseware-idea.com*)
6. Audit Leverage Audit Management Software (www.auditleverage.com)
7. Protiviti Governance, Risk, Compliance (GRC) Software (www.protiviti.com)
8. PAWS Audit & Risk Management Software (www.pentana.com)

PRACTICE NOTE

1009

COMPUTER-ASSISTED AUDIT TECHNIQUES

[Issued December 2003; revised September 2004 (name change)]

Contents

Paragraphs

PRACTICE NOTE
1009
COMPUTER-ASSISTED AUDIT TECHNIQUES

The purpose of Practice Notes issued by the Hong Kong Institute of Certified Public Accountants is to assist auditors in applying Statements of Auditing Standards (SASs) and Standards on Assurance Engagements (SAEs) of general application to particular circumstances and industries.

They are persuasive rather than prescriptive. However they are indicative of good practice and have similar status to the explanatory material in SASs and SAEs, even though they may be developed without the full process of consultation and exposure used for SASs and SAEs. Auditors should be prepared to explain departures when called upon to do so.

This Practice Note replaces Auditing Guideline 3.262 "Computer-Assisted Audit Techniques (CAATS)".

Introduction

1. The overall objectives and scope of an audit do not change when an audit is conducted in a computer information technology (IT) environment. The application of auditing procedures may, however, require the auditors to consider techniques known as Computer-Assisted Audit Techniques (CAATs) that use the computer as an audit tool.

2. CAATs may improve the effectiveness and efficiency of auditing procedures. They may also provide effective tests of control and substantive procedures where there are no input documents or a visible audit trail, or where population and sample sizes are very large.

3. The purpose of this Practice Note (PN) is to provide guidance on the use of CAATs. It applies to all uses of CAATs involving a computer of any type or size. Special considerations relating to small entity IT environments are discussed in paragraph 26.

Description of Computer-Assisted Audit Techniques (CAATs)

4. This PN describes computer-assisted audit techniques including computer tools, collectively referred to as CAATs. CAATs may be used

in performing various auditing procedures, including the following:

a. Tests of details of transactions and balances, for example, the use of audit software for recalculating interest or the extraction of invoices over a certain value from computer records;

b. Analytical procedures, for example, identifying inconsistencies or significant fluctuations;

c. Tests of general controls, for example, testing the set-up or configuration of the operating system or access procedures to the program libraries or by using code comparison software to check that the version of the program in use is the version approved by management;

d. Sampling programs to extract data for audit testing;

e. Tests of application controls, for example, testing the functioning of a programmed control; and

f. Reperforming calculations performed by the entity's accounting systems.

5. CAATs are computer programs and data the auditors use as part of the audit procedures to process data of audit significance contained in an entity's information systems. The data may be transaction data, on which the auditors wish to perform tests of controls or substantive procedures, or they may be other types of data. For example, details of the application of some general controls may be kept in the form of text or other files by applications that are not part of the accounting system. The auditors can use CAATS to review those files to gain evidence of the existence and operation o f those controls. CAATS may consist of package programs, purpose-written programs, utility programs or system management programs. Regardless of the origin of the programs, the auditors substan tiate their appropriateness and validity for audit purposes before using them.

a. *Package programs* are generalized computer programs designed to perform data processing functions, such as reading data, selecting and analyzing information, performing calculations, creating data files and reporting in a format specified by the auditors.

b. *Purpose*-written programs perform audit tasks in specific circumstances. These programs may be developed by the auditors, the entity being audited or an outside programmer hired by the auditors. In some cases the auditors may use an entity's existing programs in their original or modified state because it may be more efficient than developing independent programs.

c. *Utility programs* are used by an entity to perform common data processing functions, such as sorting, creating and printing files. These programs are generally not designed for audit purposes, and therefore may not contain features such as automatic record counts or control totals.

d. *System Management programs* are enhanced productivity tools that are typically part of a sophisticated operating systems environment, for example, data retrieval software or code comparison software. As with utility programs, these tools are not specifically designed for auditing use and their use requires additional care.

e. *Embedded Audit Routines* are sometimes built into an entity's computer system to provide data for later use by the auditors. These include:

   i. Snapshots: This technique involves taking a picture of a transaction as it flows through the computer systems. Audit software routines are embedded at different points in the processing logic to capture images of the transaction as it progresses through the various stages of the processing. Such a technique permits auditors to track data and evaluate the computer processes applied to the data.

   ii. System Control Audit Review File: This involves embedding audit software modules within an application system to provide continuous monitoring of the system's transactions.

   The information is collected into a special computer file that the auditors can examine.

f. *Test data techniques* are sometimes used during an audit by entering data (for example, a sample of transactions) into an entity's computer system, and comparing the results obtained with predetermined results. Auditors might use test data to:

i. test specific controls in computer programs, such as on-line password and data access controls;

ii. test transactions selected from previously processed transactions or created by the auditors to test specific processing characteristics of an entity's information systems. Such transactions are generally processed separately from the entity's normal processing; and

iii. test transactions used in an integrated test facility where a "dummy" unit (for example, fictitious department or employee) is established, and to which test transactions are posted during the normal processing cycle.

When test data are processed with the entity's normal processing, the auditors en sure that the test transactions are subsequently eliminated from the entity's accounting records.

6. The increasing power and sophistication of PCs, particularly laptops, has resulted in other tools for the auditors to use. In some cases, the laptops will be linked to the auditors' main computer systems. Examples of such techniques include:

a. expert systems, for example in the design of audit programs and in audit planning and risk assessment;

b. tools to evaluate a client's risk management procedures;

c. electronic working papers, which provide for the direct extraction of data from the client's computer records, for example, by downloading the general ledger for audit testing; and

d. corporate and financial modeling programs for use as predictive audit tests.

These techniques are more commonly referred to as "audit automation."

Considerations in the Use of CAATs

7. When planning an audit, the auditors may consider an appropriate combination of manual and computer assisted audit techniques. In determining whether to use CAATs, the factors to consider include:

a. the IT knowledge, expertise and experience of the audit team;

b. the availability of CAATs and suitable computer facilities and data;

c. the impracticability of manual tests;

d. effectiveness and efficiency; and

e. timing.

Before using CAATS the auditors consider the controls incorporated in the design of the entity's computer systems to which the CAATS would be applied in order to determine whether, and if so, how, CAATs should be employed.

IT Knowledge, Expertise, and Experience of the Audit Team

8. SAS 3 10 "Auditing in a computer information systems environment" deals with the level of skill and competence the audit team needs to conduct an audit in an IT environment. It provides guidance when auditors delegate work to assistants with IT skills or when the auditors use work performed by other auditors or experts with such skills. Specifically, the audit team would need to have sufficient knowledge to plan, execute and use the results of the particular CAAT adopted. The level of knowledge required depends on the complexity and nature of the CAAT and of the entity's information system.

Availability of CAATs and Suitable Computer Facilities

9. The auditors consider the availability of CAATs, suitable computer facilities (controlled as described in paragraphs 18-23) and the necessary computer-based information systems and data. The auditors may plan to use other computer facilities when the use of CAATs on an entity's computer is uneconomical or impractical, for example, because of an incompatibility between the auditors' package program and the entity 's computer. Additionally, the auditors may elect to use their own facilities, such as PCs or laptops.

10. The cooperation of the entity's personnel may be required to provide processing facilities at a convenient time, to assist with activities such as loading and running of the CAATs on the entity's system, and to provide copies of data files in the format required by the auditors.

Impracticability of Manual Tests

11. Some audit procedures may not be possible to perform manually because they rely on complex processing (for example, advanced statistical analysis) or involve amounts of data that would overwhelm any manual procedure. In addition, many computer information systems perform tasks for which no hard copy evidence is available and, therefore, it may be impracticable for the auditors to perform tests manually. The lack of hard copy evidence may occur at different stages in the business cycle.

    a. Source information may be initiated electronically, such as by voice activation, electronic data imaging, or point of sale electronic funds transfer. In addition, some transactions, such as discounts and interest calculations, may be generated directly by computer programs with no specific authorization of individual transactions.

    b. A system may not produce a visible audit trail providing assurance as to the completeness and accuracy of transactions processed. For example, a computer program might match delivery notes and suppliers' in voices. In addition, programmed control procedures, such as checking customer credit limits, may provide hard copy evidence only on an exception basis.

    c. A system may not produce hard copy reports. In addition, a printed report may contain only summary totals while computer files retain the supporting details.

Effectiveness and Efficiency

12. The effectiveness and efficiency of auditing procedures may be improved by u sing CAATs to obtain and evaluate audit evidence. CAATs are often an efficient means of testing a large number of transactions or controls over large populations by:

    a. analyzing and selecting samples from a large volume of transactions;

    b. applying analytical procedures; and

    c. performing substantive procedures.

13. Matters relating to efficiency that auditors might consider include:

a. the time taken to plan, design, execute and evaluate a CAAT;

b. technical review and assistance hours;

c. designing and printing of forms (for example, confirmations); and

d. availability of computer resources.

14. In evaluating the effectiveness and efficiency of a CAAT, the auditors consider the continuing use of the CAAT application. The initial planning, design and development of a CAAT will usually benefit audits in subsequent periods.

Timing

15. Certain data, such as transaction details, are often kept for only a short time, and may not be available in machine-readable form b y the time the auditors want them. Thus, the auditors will need to make arrangements for the retention of data required, or may need to alter the timing of the work that requires such data.

16. Where the time available to perform an audit is limited, the auditors may plan to use a CAAT because its use will meet the auditors' time requirement better than other possible procedures.

Using CAATs

17. The major steps to be undertaken by the auditors in the application of a CAAT are to:

a. set the objective of the CAAT application;

b. determine the content and accessibility of the entity's files;

c. identify the specific files or databases to be examined;

d. understand the relationship between the data tab les where a database is to be examined;

e. define the specific tests or procedures and related transactions and balances affected;

f. define the output requirements;

g. arrange with the user and IT departments, if appropriate, for copies of the relevant files or database tables to be made at the appropriate cutoff date and time;

h. identify the personnel who may participate in the design and application of the CAAT;

i. refine the estimates of costs and benefits;

j. ensure that the use of the CAAT is properly controlled and documented;

k. arrange the administrative activities, including the necessary skills and computer facilities;

l. reconcile data to be used for the CAAT with the accounting records;

m. execute the CAAT application; and

n. evaluate the results.

Controlling the CAAT Application

18. The specific procedures necessary to control the use of a CAAT depend on the particular application.

In establishing control, the auditors consider the need to:

a. approve specifications and conduct a review of the work to be performed by the CAAT;

b. review the entity's general controls that may contribute to the integrity of the CAAT, for example, controls over program changes and access to computer files. When such controls cannot be relied o n to ensure the integrity of the CAAT, the auditors may consider processing the CAAT application at another suitable computer facility; and

c. ensure appropriate integration of the output by the auditors into the audit process.

19. Procedures carried out by the auditors to control CAAT applications may include:

a. participating in the design and testing of the CAAT;

b. checking, if applicable, the coding of the program to en sure that it conforms with the detailed program specifications;

c. asking the entity's computer staff to review the operating system instructions to ensure that the software will run in the entity's computer installation;

d. running the audit software on small test files before running it on the main data files;

e. checking whether the correct files were used, for example, by checking external evidence, such as control totals maintained by the user, and that those files were complete;

f. obtaining evidence that the audit software functioned as planned, for example, by reviewing output and control information; and

g establishing appropriate security measures to safeguard the integrity and confidentiality of the data.

When the auditors intend to perform audit procedures concurrently with on-line processing, the auditors review those procedures with appropriate client personnel and obtain approval before conducting the tests to help avoid the inadvertent corruption of client records.

20. To ensure appropriate control procedures, the presence of the auditors is not necessarily required at the computer facility during the running of a CAAT. It may, however, provide practical advantages, such as being able to control distribution of the output and ensuring the timely correction of errors, for example, if the wrong input file were to be used.

21. Audit procedures to control test data applications may include:

a. controlling the sequence of submissions of test data where it spans several processing cycles;

b. performing test runs containing small amounts of test data before submitting the main audit test data;

c. predicting the results of the test data and comparing it with the actual test data output, for the individual transactions and in total;

d. confirming that the current version of the programs was used to process the test data; and

e. testing whether the programs used to process the test data were the programs the entity used throughout the applicable audit period.

22. When using a CAAT, the auditors may require the cooperation of entity

staff with extensive knowledge of the computer installation. In such circumstances, the auditors consider whether the staff improperly influenced the results of the CAAT.

23. Audit procedures to control the use of audit-enabling software may include:

    a. verifying the completeness, accuracy and availability of the relevant data, for example, historical data may be required to build a financial model;

    b. reviewing the reasonableness of assumptions used in the application of the tool set, particularly when using modeling software;

    c. verifying availability of resources skilled in the use and control of the selected tools; and

    d. confirming the appropriateness of the tool set to the audit objective, for example, the use of industry specific systems may be necessary for the design of audit programs for unique business cycles.

Documentation

24. The standard of working paper documentation and retention procedures for a CAAT is consistent with that for the audit as a whole (see SAS 230 "Documentation").

25. The working papers need to contain sufficient documentation to describe the CAAT application, such as:

    a. Planning

        i CAAT objectives;

        ii. consideration of the specific CAAT to be used;

        iii. controls to be exercised; and

        iv. staffing, timing and cost.

    b. Execution

        i. CAAT preparation and testing procedures and controls;

        ii. details of the tests performed by the CAAT;

        iii. details of input, processing and output; and

iv. relevant technical information about the entity's accounting system, such as file layouts.

c. Audit Evidence

i. output provided;

ii. description of the audit work performed on the output; and

iii. audit conclusions.

d. Other

i. recommendations to entity management.

In addition, it may be useful to document suggestions for using the CAAT in future years.

Using CAATs in Small Entity IT Environments

26. Although the general principles outlined in this PN apply in small entity IT environments, the following points need special consideration:

a. The level of general controls may be such that the auditors will place less reliance on the system of internal control. This will result in greater emphasis on tests of details of transactions and balances and analytical review procedures, which may increase the effectiveness of certain CAATs, particularly audit software.

b. Where smaller volumes of data are processed, manual methods may be more cost effective.

c. A small entity may not be able to provide adequate technical assistance to the auditors, making the use of CAATs impracticable.

d. Certain audit package programs may not operate on small computers, thus restricting the auditors' choice of CAATs. The entity's data files may, however, be copied and processed on another suitable computer.

Compatibility with International Auditing Practice Statements

27. This Practice Note is, in all material respects, in accordance with International Auditing Practice Statement 1009 "Computer-Assisted Audit Techniques".

Appendix II

Attribute Sampling
Technical Specification

A CaseWare IDEA Research Department document

June 17, 2003

Contents

1. Attribute Sampling Overview

The purpose of attribute sampling is to estimate the number of times an event occurs in a population, based on the examination of a sample from the population. In auditing applications, the events of interest are typically misstatements (differences between recorded and audited balances), deviations from internal control procedures, or failure to comply with laws and regulations. This document describes the mathematical procedures followed by IDEA in planning and evaluating attribute samples.

Attribute sampling functionality for IDEA is provided by three main functions, each of which performs one of the following tasks:

1. Planning the sample size to control beta risk
2. Planning the sample size to control alpha and beta risks
3. Determination of one or two-sided reliability intervals for a given sample (also known as sample evaluation).

These functions are described in the sections that follow. Further discussion of certain issues, relating to numerical algorithms used, may be found in [2].

2. Sample size determination controlling beta risk

Inputs:

N: population size

UOR: upper threshold occurrence rate

EOR: expected occurrence rate

BetaR: beta reliability or confidence

Outputs:

n : sample size

x_{cr}: critical number of occurrences in sample

Other quantities referred to below:

OR: true (unknown) population occurrence rate

X: true (unknown) number of occurrences in population; $X = N \times OR$

x_h: hypergeometric random variable (distribution depends on *N, n, X*)

X_{tol}: upper threshold value of *X*; $X_{tol} = N \times UOR$

Description:

This planning function finds the minimum *n* and corresponding occurrence rate x_{cr} such that:

1. It can be inferred with a reliability of at least *BetaR* that the true occurrence rate is less than or equal to *UOR*, for population size *n* and number of sample occurrences up to x_{cr} .
2. x_{cr} /*n* is at least as big as *EOR*.

Please refer to [1] for a description of the meaning of reliability in this context.

Condition 1) is met in the following way: Set

$$X_{tol} = \text{floor} (N \times UOR) + 1$$

where 'floor' denotes the rounding down operation. Then, *n* and x_{cr} must satisfy

$$\text{prob} (x_h > x_{cr} \mid N, n, X_{tol}) \geq BetaR \text{ .}$$

where 'prob' denotes total probability. This means that, if the true number of occurrences *X,* were as big as X_{tol}, then the probability is at least *BetaR* and the number of occurrences in a sample of size *n* would be *greater than* x_{cr}. In other words, provided the sample of size *n* contains no more than x_{cr} occurrences, the interval [0, X_{tol} –1] may be expected to contain the true value *X* with reliability at least *BetaR*. Dividing by *N* to obtain occurrence rates, we see that the interval

[0, *UOR*] may be expected to contain the true value *OR* with reliability *BetaR*. Further discussion of the beta risk control can be found in [2].

3. Sample size determination controlling alpha and beta risks

Inputs:

N: population size

UOR: upper threshold occurrence rate

LOR: lower threshold occurrence rate

BetaR: beta - reliability or confidence

AlphaR: alpha - reliability or confidence

Outputs:

n : sample size

x_{cr}: critical number of occurrences in sample

Other quantities referred to below:

OR: true (unknown) population occurrence rate; $OR = N\times X$

X: true (unknown) number of occurrences in population

x_h: hypergeometric random variable (distribution depends on *N, n, X*)

X_{tol}: lower limit of *X* to ensure tolerable beta-risk (1 –*BetaR*)

X_{min}: lower limit of *X* to ensure tolerable alpha-risk (1 – *AlphaR*)

Description:

This planning function finds the minimum *n* and corresponding occurrence rate x_{cr} such that:

1. It can be inferred with a reliability of at least *BetaR* that the true occurrence rate is less than or equal to *UOR*, for population size *n* and number of sample occurrences up to x_{cr}. Equivalently, the risk of deciding that $OR \le UOR$ when in fact $OR > UOR$, is no greater than the beta risk (1 – *BetaR*).
2. The risk of deciding that $OR > UOR$ when in fact $OR \le LOR$, is no greater than the alpha risk (1 – *AlphaR*).

The reader is referred to [1] for a description of the meanings of reliability and risk in this context.

Condition 1) is met in the following way: Set

$$X_{tol} = \text{floor}\ (\ N\times UOR\ ) + 1$$

where 'floor' denotes the rounding down operation. Then, *n* and x_{cr} satisfy

$$\text{prob}\ (x_h > x_{cr}\ |\ N, n, X_{tol}) \ge BetaR$$

where 'prob' denotes total probability. This is the same method as is used to control beta risk in planning to control only beta risk; its interpretation is discussed in Section 2 above.

Condition 2) is met in the following way: Set

$$X_{\min} = \text{ceil}\ (\ N \times LOR\ )$$

Where 'ceil' is the rounding up operation. Then, n and x_{cr} satisfy

$$\text{prob}\ (x_h \le x_{cr} \mid N, n, X_{\min}) \ge AlphaR$$

This means that, if the true number of occurrences is $X_{\min}$ (or less), then the probability is at least *AlphaR* that the number of occurrences in a sample of size n will be x_{cr} or smaller, and therefore that we will not falsely conclude a number of population occurrences greater than X_{tol} . Dividing by N to obtain occurrence rates, we see that the reliability *AlphaR* is achieved for any true population occurrence rate of *LOR* or less.

4. Determination of 1- or 2-sided reliability intervals

Inputs:

sides: 1 or 2; number of interval endpoints to calculate.

N: population size

n : sample size

x: number of occurrences in sample

R: reliability or confidence

Outputs:

OR_{right}: upper precision limit of OR; $OR_{right} = X_{right}/N$

OR_{left}: lower precision limit of OR; $OR_{left} = X_{left}/N$; not calculated if *sides* = 1.

Other quantities referred to below:

X: true (unknown) number of occurrences in population

X_{right}: upper precision limit of X

X_{left}: lower precision limit of X; not calculated if *sides* = 1.

x_h: random variable with hypergeometric distribution specified by N, n, X.

OR: population occurrence rate; $OR = X/N$

Description:

If *sides* = 2, risk is divided equally between upper and lower limits, i.e.:

If *sides* = 1, $R' = R$.

If *sides* = 2, $R' = (1+R)/2$.

If *sides* = 1, then X_{left} is not calculated (returns 0), and X_{right} satisfies the following:

1. If $X > X_{right}$, then prob $(x_h > x \mid N, n, X) \geq R'$
2. If $X \leq X_{right}$, then prob $(x_h > x \mid N, n, X) < R'$

This means that, if the true number of occurrences X were any more than X_{right}, then the probability is at least R' that the number of occurrences in a sample of size n would be greater than the observed number x. In other words, X is less than or equal to X_{right} with reliability R'.

If *sides* = 2, then X_{right} is as above, and X_{left} satisfies the following:

1. If $X < X_{left}$, then prob $(x_h < x \mid N, n, X) \geq R'$
2. If $X \geq X_{left}$, then prob $(x_h < x \mid N, n, X) < R'$

This means that, if the true number of occurrences X were less than X_{left}, then the probability is at least R' that the number of occurrences would be less than the observed number x. In other words, X is greater than or equal to X_{left} with reliability R'.

The above definitions of X_{left} and X_{right} are similar to those given in [1] translated into mathematical notation, with one minor adjustment as described in [2].

Although it is a standard technique [1], the process of dividing the tolerable risk

$1 - R$ between the upper and lower limits to obtain two-sided intervals, can lead to unnecessarily conservative intervals whenever $X_{left} = 0$ [2]. This situation most frequently occurs when $x = 0$, but may occur for other low numbers of occurrences, when n and N are large. When it does, IDEA displays a message saying that the two-sided interval is unavailable.

References:

[1] Donald M. Roberts, *Statistical Auditing*, AICPA 1978; p. 184.

[2] K.M. Adeney, "Appendix A", CaseWare Idea Inc. Internal Report, April 2002.

Appendix A: Technical Note

1. Introduction

Attribute sampling functionality for IDEA is provided by three main functions, each of which performs one of the following tasks:

1. Planning the sample size to control beta risk
2. Planning the sample size to control alpha and beta risks
3. Determination of one or two-sided reliability intervals for a given sample (also known as sample evaluation)

The above functions are described in [2]. This document provides extra detail on certain aspects of the methodology.

2. Hyper geometric Risk Function Stability

Risk functions for attribute sampling are calculated precisely, using the hypergeometric probability function as appropriate for sampling without replacement. The hypergeometric risk (HGR) function is based on the cumulative hypergeometric probability – that is, the sum of probabilities of observed sample occurrence numbers ranging from zero to the specified number of occurrences. This quantity may be calculated in any of a number of ways, with different methods exhibiting different numerical properties. Since direct computation of each hypergeometric probability is computationally expensive and numerically unstable, recursive relationships between probabilities are exploited in calculating the sum.

Outputs of IDEA's HGR function were tested against those produced by existing software programs such as ATSIZ1, ATSIZ2 and ATEVAL, as described in [1], as well as an additional program called SAMPLE, which performs the same calculations. The HGR function used by the BASIC ATSIZ1 and ATSIZ2 code appears to calculate HGR using a mathematical recursion that is initialized with the probability of observing zero occurrences. Although this procedure is more stable than calculation of HGR using direct evaluations of factorial expressions, it has a bug: it always returns zero when the probability of observing zero occurrences in the sample is zero, even if the total hypergeometric risk is not zero. Furthermore, this implementation becomes unstable when the probability of observing zero sample occurrences is very small (i.e. when both population occurrence rate and sample size is large).

Errors in sample sizes returned by SAMPLE are consistent with the hypothesis that the HGR function used by SAMPLE also has the bug described above. In sampling to control beta risk, SAMPLE looks for lowest n such that the risk function is below 1 – *R* (where *R* is the reliability). Sample sizes that are too low may result from the fact that, because of the bug in the HGR function, the risk condition is met sooner (for lower *n*) than it should be.

IDEA's attribute sampling routines calculate HGR using a mathematical recursion that is similar to the one used in previous HGR functions, except that it is initialized with a value that is guaranteed not to be zero. This value is the probability of observing whatever number of occurrences is most likely for the given population occurrence rate and sample size. This implementation does not exhibit the bug described above.

Although the new HGR function can better tolerate smaller individual event probabilities compared to the previous HGR functions, ANY discreet cumulative probability distribution will become more unstable, as the individual event probabilities summed to obtain the cumulative probability become very small. This occurs when both sample size and population size are large. For example, for a population of 1,000,000 and a population occurrence rate of 10%, the HGR function is valid for sample sizes up to 8300.

The range of allowable sample sizes has not yet been characterized for an extensive number of occurrence rates and population sizes – in general, it increases as population occurrence rate decreases, and decreases as population size increases. However, the required reliability values would need to be very high, and the tolerable and expected occurrence rates very close, in order for such large sample sizes to be required in attribute sampling. Therefore, instability of the HGR function is not expected to be of practical importance. Code has been included to detect this instability if it arises. If instability of the hypergeometric function does pose a practical problem in future, a continuous approximation to the cumulative hypergeometric probability may be used to circumvent this problem.

3. Precision Limit Definitions

The evaluation function finds reliability interval endpoints, or precision limits, while the two planning functions use precision limits implicitly in order to find sample sizes. It is important that precision limits used by all programs be defined consistently. This section describes some issues that have been identified with the definitions of precision limits found in Roberts [1] and in the outputs of past sampling programs.

Firstly, it is noted that numbers of population occurrences are always integers, so precision limits on population occurrence rates should correspond to integer population occurrence numbers. This is not the case with outputs from the previous DOS program SAMPLE, although it appears to hold for the outputs of the DOS program ATEVAL. Precision limits will be discussed here in terms of numbers of occurrences, rather than occurrence rates, since these discrete quantities are the ones actually calculated using the hypergeometric risk function.

We first consider the definitions of the left and right precision limits in Roberts' precision limit evaluation function, ATEVAL [1]. According to the description of that function, the right precision limit on the number of population occurrences is some number X_{right} such that

$$\text{prob}\ (x_h > x_{obs} \mid N, n, X_{right}) = R\ , \qquad (1)$$

where x_{obs} is the observed number of sample deviations and other quantities are as described above. Here the difficulty arises. Because the hypergeometric distribution is discrete, it is highly unlikely that there is some value of R satisfying the above equation exactly. One may instead say that X_{right} is the *minimum* value of X satisfying

$$\text{prob}\ (x_h > x_{obs} \mid N, n, X) \geq R\ . \qquad (2)$$

The interpretation of Ineq. (2) is that, if the true number of occurrences were as big as X_{right}, then the probability is at least R that the number of occurrences in a sample of size n would be *greater than* the observed number x_{crit}. In other words, X is strictly less than X_{right} with reliability R, or equivalently X is less than or equal to $X_{right} - 1$ with reliability R. Values returned by SAMPLE are consistent with these definitions (except where they are in error due to instability of that program's hypergeometric risk function).

Alternately, one may define X_{right} to be the *maximum* value of X satisfying

$$\text{prob}\ (x > x_{obs} \mid N, n, X) < R\ . \qquad (3)$$

The difference between these two definitions is illustrated below, where X_2 is the value of X_{right} resulting from Ineq. (2) and X_3 is the value of X_{right} resulting from Ineq. 3.

X	:	•	•	•	•	•
prob > or < R?	:	<	<	≥	>	>
				↑	↑	
				X_3	X_2	

Note that $X_3 = X_2 - 1$ always.

In Attribute Sampling, Ineq. (3) is used, so that the left and right precision limits are included in the reliability *R* interval. For two-sided evaluation, X_{left} is the *minimum* value of *X* satisfying

$$\text{prob}\ (x < x_{obs} \mid N, n, X) < R\ . \qquad (4)$$

where *R* has been obtained by dividing the acceptable risk between upper and lower limits as described in [1].

4. Beta Risk Control

We now consider Roberts' descriptions of the sample size determination programs ATSIZ1 and ATSIZ2 [1], in particular the method of controlling beta risk. Roberts says that the returned sample size is the minimum possible such that the population deviation rate is *strictly less than* the upper threshold occurrence rate *UOR* (Roberts calls this the *tolerable deviation rate*) with reliability at least *R*, subject to constraints on either the expected deviation rate (ATSIZ1) or the alpha risk (ATSIZ2). This is the minimum sample size *n* satisfying (subject to constraints)

$$\text{prob}\ (x_h > x_{crit} \mid N, n, X_{tol}) \geq R\ . \qquad (5)$$

Here x_h is a hypergeometric random variable with population *N*, sample size *n*, and number of population occurrences X_{tol}, and x_{crit} is the critical number of sample deviations. Roberts does not specify the way in which X_{tol} is obtained from *UOR*. Either rounding up or rounding down (e.g. X_{tol} = ceil($UOR \times N$) or X_{tol} = floor($UOR\ \times N$)) may be used. DOS versions of SAMPLE and ATSIZ1 and ATSIZ2 appear to have used rounding down.

In summary, in Roberts' ATSIZ1 and ATSIZ2 programs, the true occurrence rate *OR* is *strictly less than* the *UOR* with reliability *R* – so the *UOR* is excluded from the 1-sided reliability *R* interval. In Attribute Sampling, the above procedure is modified slightly so that the population occurrence rate can be as high as the *UOR* without going below the desired beta reliability *R*. This is achieved by setting

$$X_{tol} = \text{floor}(UOR \times N) + 1 \qquad (6)$$

Thus, we have

$$UOR \times N < X_{tol} \leq UOR \times N + 1 \qquad (7)$$

The above value of X_{tol} is used in Ineq. (5). By the reasoning given following Ineq. (2), we know that with reliability *R*, the true population number of occurrences *X* satisfies

$$X \leq X_{tol} - 1 \quad (8)$$

Dividing Ineq. (8) by N to translate into occurrence rates, and combining with Ineq. (7), we get

$$OR = X / N \leq (X_{tol} - 1) / N \leq (UOR \times N + 1 - 1) / N = UOR \ (9)$$

with reliability R. Thus the UOR is now included in the 1-sided reliability R interval.

The attribute sampling functions return slightly smaller sample sizes for some cases than do programs that exclude the UOR from the reliability interval. This difference becomes more pronounced as the sample size grows relative to the population size.

5. Asymmetric Risks in Two-Sided Intervals

Although it is a standard technique [1], the process of dividing the tolerable risk $1 - R$ between the upper and lower limits to obtain two-sided intervals, can lead to unnecessarily conservative intervals, whenever $X_{\text{left}} = 0$ [2]. In such cases the found value of $X_{\text{left}} = 0$ would work for lower confidences than the specified value of $R' = (1+R) / 2$ [1], [2]. Said another way, the risk allowed for the left limit (i.e., $(1 - R)/2$) may not all be "used up" when the lower limit is zero. Therefore the true confidence of the supplied interval is higher than R.

A narrower interval (e.g., between $X_{\text{left}} = 0$ and some smaller value of X_{right}) could be found which would still have a total risk less than or equal to $1 - R$. For such an interval, the risks assigned to each of the upper and lower limits would be unequal, and would have to be determined by a software search. For the present time, IDEA instead displays a message saying that the two-sided interval is unavailable, whenever the calculated lower limit is 0.

Appendix III

Monetary Unit Sampling
Technical Specification

IDEA

A CaseWare IDEA Research Department document

CASEWARE
INTERNATIONAL INC.

May 7, 2003

CaseWare IDEA Inc.
145 King Street East, 2nd Floor
Toronto, Ontario
M5C 2Y8

Contents

Introduction

The purpose of Monetary Unit Sampling (MUS) is to estimate an upper limit on the error in a population, based on examination of a MUS sample from the population. The sampling unit in an MUS sample is an individual dollar, rather than an item or database record. Thus, the number of units in the sampled population (commonly denoted by N) is equal to the total monetary value of this population (commonly denoted by Y). The theory behind MUS is complex and its explanation is beyond the scope of this paper – interested readers may see [2], [3], [4].

This document describes the mathematical procedures followed by IDEA in planning and evaluating Monetary Unit samples. Different methodologies are used to evaluate samples depending on whether they are found to contain a low number of errors (fewer than 20), or a high number of errors (20 or more). For the low-error case, IDEA uses a cell-based MUS evaluation similar to that described in [2]. An earlier version of this implementation is described in a previous IDEA Users' Guide [1]. For the high-error case, classical PPS theory [4] is applied. IDEA also supports joint evaluation of several samples to estimate an upper error limit for the combined population (that is, the union of all of the populations from which the samples were drawn); this evaluation uses an extension of the Stringer bound described in [3].

Note that the basic precision pricing (*BPP*) and confidence level (*CL*) are input to the screens as percentages, but appear in the formulae below as decimal fractions, for example 0.95 rather than 95.

Handling of High Value Items

High value (HV) items are records in the population having recorded amounts greater than or equal to a user-specified *high value amount*. IDEA supports two methods of dealing with HV items. The first is to extract all high value items to a separate file, the *high value database*, during sample extraction. The remaining items are then extracted to the main sample database using an average sampling interval equal to the planned sampling interval. Errors in the HV database are summed and added to the most likely error and upper error limit derived from the main sample database. They do not affect the precision gap between the upper error limit and most likely error.

The second option is to treat HV items no differently from other items in the population. In this case high value items may be sampled one, two, or more times. Thus, the main sample database can contain multiple records for a single HV item in the population database. Errors in HV items in the main sample database affect the precision of the upper error limit in the same way as any other errors in the sample. This mode is useful for situations in which high value records in the database are aggregates of many smaller items.

The HV amount is the sampling interval by default. However, users may set the HV amount to any value.

Users may optionally specify a HV file in the evaluation dialogs (or one HV file for each sample database, in the case of a combined evaluation IDEA generates a warning whenever any recorded amount in the main sample database is greater than the lesser of the sampling interval or the specified HV amount, if one was provided in sample planning. If the sample was planned using IDEA's MUS - Plan Sample routine, the HV amount used for this test is the one specified during the planning stage. Otherwise, a HV amount equal to the sampling interval is used. The HV amount does not affect the values computed by the evaluation, but is only used for generating these warnings.

3. Planning

3.1 Planning when Population Size is known

Inputs:

Y	Total monetary value of population from which sample was drawn	No restrictions See Note 1.
X_{max}	Maximum tolerable monetary value of all errors in population (e.g., materiality for this test) – absolute value	$0 < X_{max} \leq Y$ See Note 1.
X_{exp}	Expected monetary value of all errors in population – absolute value	$0 \leq X_{exp} < X_{max}$ $X_{exp} < 0.9\ Y$ See Note 1.
BPP	Basic precision pricing (assumed worst-case tainting on errors in population that do not appear in the sample).	$0 < BPP < 10$
CL	Confidence level (a.k.a. reliability)	$0 < CL < 1$ See Note 1.

Note 1: As well as the above individual restrictions on the inputs, a restriction is placed on the group of inputs. This restriction is designed to filter out inputs that would lead to impractically large sample sizes or numbers of sample errors. It has the following form:

$$\frac{X_{\max}}{Y}(1-CL)\sqrt{X_{\max}-X_{\exp}} \;\geq\; 0.07 \qquad (3.0)$$

Outputs:

n : approximate sample size (for population with no HV items removed)

SI: average sampling interval; *N*/*n*

TT: Tolerable tainting in sample.

Other quantities referred to below:

*x**: tolerable number of 100% - tainted errors in sample.

*n**: minimum sample size that can tolerate *x** 100% - tainted errors

N: size of population from which sample was drawn.

For MUS, $N = |Y|$.

Description:

For planning the sample, and also for evaluating a single sample when few errors are present, IDEA's MUS functions use cell-based MUS methods similar to those presented in references [1], [2]. One slight change is that IDEA now uses hyper geometric factors when population size is known, while [1] and [2] describe methods in terms of Bernoulli and/or Poisson factors. It is well known that hyper geometric factors are the most appropriate for modeling sampling from finite populations (e.g., see [2]). However, at the time [1], [2] were written, computations of hyper geometric factors was not feasible, and the use of the Bernoulli and Poisson approximations was common. Since IDEA already has stable hyper geometric probability functions for its Attribute Sampling module, these are used in MUS as well. The practical effect of this change on planned sample sizes and computed error limits is very small.

When population size is known, we first compute the sample size *n* and then use this to compute the *SI*. This planning function finds the approximate minimum *n* and corresponding tolerable tainting *TT* such that it can be inferred with a reliability of at least *CL* that the true number of errors is less than the tolerable number of errors, X_{max}. Equivalently, the risk of deciding that $X < X_{max}$ when in fact $X \geq X_{max}$ is no greater than (1 – *CL*). The sampling interval, sample size, and number of tolerable errors in the sample are first computed for the case of 100% tainting, and then scaled by the BPP to obtain *SI*, *n*, and *TT*.

The hyper geometric distribution is used for the risk calculations. Nonetheless, the planning process is approximate, since the upper error limit obtained when a sample is evaluated depends not just upon the total tainting, but also on how the tainted dollars are distributed among the errors.

The cumulative hyper geometric distribution is given by

$$H(x,n,X,N) = \sum_{k=1}^{x}\left(\frac{X!}{(X-k)!k!} \times \frac{(N-X)!}{(N-X-(n-k))!(n-k)!} \times \frac{(N-n)!n!}{N!} \right) \tag{3.1}$$

$H(x,n,X,N)$ Is the risk of observing *x* or fewer errors in a sample of size *n* drawn from a population of size *N* in which the total number of errors is *X*.

The sample size that would be appropriate if all taintings were 100%, denoted by *n**, is computed in the following way:

Set the population total error as $X = X_{\max}$.

We first describe the case when the expected error is greater than zero. Then, we consider sample errors *x* = 0, 1, 2,. For each *x* considered, we find the minimum sample size, denoted *n*(*x*), which will tolerate *x* errors at the desired confidence level *CL*. That is, if up to *x* errors are found in a sample of size *n*(*x*), the *CL*-upper precision limit on the population error rate is less than $X_{\max}$. This is done as follows:

For each successive *x*, we find the lowest *n* such that $H(x,n,X,N)$ does not exceed the acceptable risk, 1 – *CL*. Thus,

$$n(x) = \min \left[n \ni H(x,n,X_{\max},N) \leq 1 - CL \right] \tag{3.2}$$

We thus obtain a succession of pairs $(x, n(x))$ giving sample sizes $n(x)$ which could tolerate *x* errors.

For each pair we then compute the sample error rate $x/n'(x)$, and compare this with the expected error rate for the population, $p_{\exp} = X_{\exp}/N$. Let x' and $x'+1$ be successive integers such that

$$\frac{x'}{n'(x')} \le \frac{X_{\exp}}{N} \le \frac{x'+1}{n'(x'+1)} \quad . \tag{3.4}$$

We find the values *x** and *n** by interpolating between these pairs to achieve the desired ratio $x^*/n^* = p_{\exp} = X_{\exp}/N$. This yields

$$\frac{x^* - x'}{x'+1-x'} = \frac{n^* - n'(x')}{n'(x'+1) - n'(x')} \quad . \tag{3.5}$$

Substituting $x^* = n^* X_{\exp} / N$ and solving, gives

$$n^* = ceil\left(\frac{x'\, n'(x'+1) - x''\, n'(x')}{\left[n'(x'') - n'(x') \right] p_{\exp} - 1} \right) \tag{3.6}$$

$$x^* = p_{\exp}\; n^*$$

Where *ceil* () denotes rounding up. When the expected error is zero, we set $x^* = 0$, and compute $n^*(0) = n(0)$ using (4.2).

Finally, we scale by the BPP to get

$$\begin{aligned} TT &= x^* \times BPP \\ n &= floor(n^* \times BPP) \\ SI &= N/(n^* \times BPP) \end{aligned} \tag{3.7}$$

Where *floor* () denotes rounding down. Note that *n* is an approximate sample size – the *achieved sample size* may different from *n* due to the removal of high value items from the database during sampling. High value extraction can also change the *achieved sampling interval* slightly (that is, the final population size less high value items, divided by the final sample size).

A Further Note on High Value Items

The planning function computes a sampling interval, tolerable total taintings, and sample size required to meet the objectives on upper error limit and confidence, without consideration of high value items. When the sample is extracted, removal of HV items decreases both the population value and the sample size from the planned values. If the expected and maximum tolerable error *rates* in the population after removal of HV items is the same as that used in planning (i.e., $X_{\exp}/N$ and $X_{\max}/N$ respectively), removal HV items will not affect the ability of the sample to achieve the planned precision to any noticeable degree.

3.2 Planning when Population Size is Unknown

Inputs:

$X_{\max}$	Maximum tolerable monetary value of all errors in population (e.g., materiality for this test) – absolute value	$X_{\max} > 0$
$X_{\exp}$	Expected monetary value of all errors in population – absolute value	$0 < X_{\exp} < X_{\max}$
BPP	Basic precision pricing (assumed worst-case tainting on errors in population that do not appear in the sample).	$0 < BPP < \infty$
CL	Confidence level (a.k.a. reliability)	$0 < CL < 1$

Outputs:

SI: average sampling interval; *N*/*n*

TT: Tolerable tainting in sample.

Other quantities referred to below:

*x**: tolerable number of errors in sample.

$\lambda(x, C)$: Poisson factor (a.k.a. UEL factor) for observed number of errors *x* and confidence level *C*.

N: size of population from which sample was drawn. for MUS $N = |Y|$.

Description:

When population size is unknown, we compute the *SI* directly using the Poisson distribution to approximate the hyper geometric.

We first find successive integers x' and $x'+1$ such that

$$\frac{x'}{\lambda(x', CL)} \le \frac{X_{\exp}}{X_{\max}} \le \frac{x'+1}{\lambda(x'+1, CL)} \quad . \tag{3.8}$$

The Poisson factors $\lambda(x', CL)$ and $\lambda(x'+1, CL)$ are found using a search procedure, which we shall now describe. The cumulative Poisson distribution is given by

$$P(\lambda, x) = \sum_{k=0}^{x} e^{-\lambda} \ \lambda^{k} \ k! \tag{3.9}$$

Where $k!$ denotes the factorial of *x*, i.e. $k! = k(k-1)(k-2)...(2)$.

The Poisson factor $\lambda(x, \ CL)$ is that value of λ satisfying

$$P(\lambda(x, CL), \ x) \approx 1 - CL \tag{3.10}$$

IDEA computes Poisson factors by incrementing a candidate λ until $\lambda(x, \ CL)$ satisfying (3.10) is found. The computation is accurate to five decimal places. One interpretation of the Poisson factor is as follows: $\lambda(x, \ CL)$ is the *CL* - upper limit on the *expected* number of errors in any future sample, when an examined sample is found to have *x* errors.

Having found x' and $x'+1$, and their corresponding Poisson factors, we next interpolate to find values *x** and $\lambda*$ such that $x*/\lambda* = X_{\exp} / X_{\max}$. Thus,

$$\frac{x*-x'}{x'+1-x'} = \frac{\lambda*-\lambda(x', CL)}{\lambda(k'+1, CL) - \lambda(x', CL)} \ . \tag{3.11}$$

Substituting $x* = \lambda* X_{\exp} / X_{\max}$ and solving, gives

$$\lambda* = \frac{x' \ \lambda(k'+1, CL) \ - \ x'' \ \lambda(k', CL)}{[\lambda(x'+1, CL) - \lambda(x', CL)] \times X_{\exp} / X_{\max} - 1} \tag{3.12}$$

$$x* = \ \lambda* \times X_{\exp} / X_{\max}$$

Finally, we compute the *SI* and tolerable taintings, scaling by the BPP to get

$$SI = \frac{X_{\max}}{\lambda* \times BPP} \ . \tag{3.13}$$

$$TT = x* \times BPP$$

A Further Note on High Value Items

As in the case for known population size, removal of HV items decreases both the population value and the sample size from the planned values. Once again, if X_{exp} and $X_{\max}$ are scaled to reflect the decrease in population size, removal HV items will not affect the ability of the sample to achieve the planned precision to any noticeable degree.

3.3 Warning Messages from Planning

IDEA generates warning messages in pop-up windows if certain conditions are encountered during MUS Planning. Warnings do not prevent the user from executing the planning operation as planned, but are merely notifications. The table below shows these conditions and the resulting warnings.

Condition	Warning
n > 0.5 × number of records in population database	WARNING: The computed sample size is more than half of the number of records in the population database. Results from evaluating this sample may be overly conservative. Classical Variables sampling may be more appropriate for this test. Consultation with a statistical sampling expert is recommended.
$X_{\max} > N$	Warning: the tolerable error exceeds the total population value.
$X_{\text{exp}} > 0.5X_{\max}$	Warning: you have specified an expected error of more than half the tolerable error. Monetary unit sampling is inefficient for such high-error populations. Classical Variables sampling may be more appropriate for this test.
High Value items extracted to separate file and High Value Amount > Sampling Interval	Warning: you have specified a high value amount greater than the sampling interval. Items having recorded amounts between the high value amount and the sampling interval may be sampled multiple times; statistical projection of errors found in such items is not recommended. Change HVA / Confirm.

Note: IDEA does not support sampling intervals of less than 1.0 or sample sizes of greater than 1,000,000. If the planning function computes values outside of these accepted limits, it fails and the user is asked to try different parameters.

4. Sample Evaluation

4.1 Single Sample Evaluation when Fewer than 20 Errors are found

Inputs:

Y	Total monetary value of population from which sample was drawn	
n	Sample size	$n > 0$, $n \geq$ number of records in sample database, $n < Y$
BPP	Basic precision pricing (assumed worst-case tainting on errors in population that do not appear in the sample).	$0 < BPP < 10$
CL	Confidence level (a.k.a. reliability)	$0.0 < CL \leq .995$
$RA(k)$	Recorded amount for k^{th} record in main sample, for $k = 1 \ldots n$ (obtained from sample database)	$RA(k) \neq 0$
$AA(k)$	Audited amount for k^{th} record in main sample, for $k = 1 \ldots$ number of records (obtained from sample database)	$\left\lvert \dfrac{RA(k) - AA(k)}{RA(k)} \right\rvert \leq BPP$

Outputs:

Gross UEL o/s NoHV: upper precision limit of overstatements projected from main sample,

Gross UEL u/s NoHV: upper precision limit of understatements projected from main sample, disregarding overstatements and not including errors in HV items

Gross MLE o/s NoHV: most likely error for understatements projected from main sample, disregarding understatements and not including errors in HV items

Gross MLE u/s NoHV:	most likely error for understatements projected from main sample, disregarding overstatements and not including errors in HV items
Net UEL o/s NoHV:	upper error limit for overstatements net of understatements not including errors in HV items
Net UEL u/s NoHV:	upper error limit for understatements net of overstatements not including errors in HV items
Net MLE o/s NoHV:	MLE for overstatements net of understatements not including errors in HV items
Net MLE u/s NoHV:	MLE for understatements net of overstatements not including errors in HV items
HVE o/s:	total overstatements in high value items
HVE u/s:	total understatements in high value items
Gross UEL o/s wHV:	upper error limit for overstatements, disregarding understatements but including errors in HV items
Gross UEL u/s wHV:	upper error limit for understatements, disregarding overstatements but including errors in HV items
Gross MLE o/s wHV:	most likely error for overstatements, disregarding understatements but including errors in HV items
Gross MLE u/s wHV:	most likely error for understatements, disregarding overstatements but including errors in HV items
Net UEL o/s wHV:	upper error limit for overstatements net of understatements, including errors in HV items
Net UEL u/s wHV:	upper error limit for understatements net of overstatements, including errors in HV items

Net MLE o/s wHV: MLE for overstatements net of understatements, including errors in HV items

Net MLE u/s wHV: MLE for understatements net of overstatements, including errors in HV items

Precision o/s: Total precision of UEL for overstatements

Precision u/s: Total precision of UEL for understatements

Other quantities referred to below:

SI: average sampling interval

$t_{o/s}(k)$: tainting for the k^{th} overstatement in main sample, ordered from largest to smallest.

$t_{u/s}(k)$: tainting for the k^{th} understatement in main sample, ordered from largest to smallest.

$x_{o/s}$: total number of overstatements in main sample

$x_{u/s}$: total number of understatements in main sample

N: size of population from which sample was drawn. For MUS, $N = \left| Y \right|$.

Description:

IDEA uses cell evaluation [2] to compute a most likely error and an upper error limit for the population. These quantities are computed separately for overstatements and understatements. The resulting (gross) estimates are then combined to produce a net most likely error and a net upper error limit for the population. The hyper geometric probability distribution is used throughout.

Note that only the separate (Gross) upper error limit for overstatements has been extensively tested. The guidance of a statistical expert should be sought before the gross and net most likely errors, or the net upper error limit, are used in any audit decision.

Sampling Interval

The sampling interval is computed as

$$SI = N / n$$

Note that previous versions of IDEA have used the sampling interval used in the extraction, say *SI**, which is not exactly equal to *N*/*n* due to the presence of a remainder when *N* is divided by *SI** - that is, *n* = floor (*N* / *SI**). Since the derivation of MUS calculations [2], [3], is based on a *SI* of exactly *N*/*n*, this quantity is now used in the evaluation. The value of *SI** used for extracting the sample can be found in the sample history, if the sample was extracted using IDEA's MUS extraction.

Taintings

For each item in the main sample database, let

$$\tilde{t}(k) = \frac{RA(k) - AA(k)}{RA(k)} \ . \qquad (4.1a)$$

We denote taintings of overstatements and understatements respectively by

$$t_{o/s}(k) = \left| \tilde{t}(k') \right| ; \quad RA(k') > AA(k') \quad (4.1b)$$

$$t_{u/s}(k) = \left| \tilde{t}(k') \right| ; \quad RA(k') < AA(k') \ .$$

Note that recorded amounts may be either positive or negative. The following table gives possible ranges for tainting values, under the assumption that the audited amount has the same sign as the recorded amount.

	Recorded amount Sign	
	Positive: $RA(k) > 0$	Negative: $RA(k) < 0$
Overstatement: $RA(k) > AA(k)$	$0 < \tilde{t}(k) < 1$ $0 < t(k) < 1$	$-\infty < \tilde{t}(k) < 0$ $0 < t(k) < \infty$
Understatement: $RA(k) < AA(k)$	$-\infty < \tilde{t}(k) < 0$ $0 < t(k) < \infty$	$0 < \tilde{t}(k) < 1$ $0 < t(k) < 1$

Since they lead to taintings of greater than 100%, understatements for positive recorded amounts or overstatements for negative recorded amounts can dominate the results. IDEA generates a warning suggesting that a sampling expert be consulted, if any taintings greater than 100% are encountered.

Taintings of both overstatements and understatements are ordered from largest to smallest.

If any tainting is greater than the input *BPP*, the *BPP* must be adjusted to a value greater than or equal to the maximum tainting before the evaluation can proceed.

Projection of Most Likely Errors from Sample

We here describe the procedure for projecting the most likely errors from the sample, without consideration of errors in high value amounts. The formulae used are

$$\textit{Gross MLE o/s NoHV} \;=\; SI \times \sum_{k=1}^{x_{o/s}} t_{o/s}(k) \tag{4.2}$$

$$\textit{Gross MLE u/s NoHV} \;=\; SI \times \sum_{k=1}^{x_{u/s}} t_{u/s}(k)$$

Projection of Upper Error Limits from Sample

We here describe the procedure for projecting the upper error limits from the sample, without consideration of errors in high value amounts. We shall provide details for *Gross UEL o/s NoHV*; the computation for understatements is directly analogous. For some quantities below we omit the subscript *o/s*; all errors and taintings in the equations below should be understood to be overstatements. The upper limit is given by

$$\textit{Gross UEL o/s NoHV} = F(x) \times SI \,. \tag{4.3}$$

The factor $F(x)$ is computed using the cell evaluation method [2]. This technique takes advantage of certain constraints on the possible population errors, which are a consequence of the cell extraction method employed by IDEA's MUS Extraction routine. It is a recursive calculation employing hyper geometric factors $\lambda_H(k, n, N, CL)$, which will be defined below.

The recursive procedure is given by:

$$F(0) = \lambda_H(0, n, N, CL) \times BPP \tag{4.4a}$$

$$F(k) \;=\; \max\left\{ F(k-1) + t(k),\;\; \lambda_H(k, n, N, CL) \times \tfrac{1}{k} \sum_{i=1}^{k} t(i) \right\} \tag{4.4b}$$

for $k = 1, \ldots, x$.

We now define the hyper geometric factors. Let $X_{\max}(k)$ be the *CL*-upper precision limit on the number of errors in a population of size *N*, when *k* errors are found in a sample of size *n*. Note that for any *X*, it can be inferred with a confidence level $1-H(k,n,X,N)$ that the true population error rate is *less than X* , where $H(k,n,X,N)$ is the hyper geometric risk function defined in Eq. 3.1. This is because, if the error rate were as high as *X*, then the number of sample errors would be *greater than* the observed value of *x* with a probability of $1-H(x,n,X,N)$. Therefore, the upper precision limit on the number of errors in the population at a confidence level of *CL* = $1-H(x,n,X,N)$ is $X-1$. By the above reasoning, $X_{\max}(k)$ is

$$X_{\max}(k) = \min\left[X \ni H(k,\ n,\ X+1,\ N) \le 1-CL\right] . \qquad (4.5)$$

The hyper geometric factor is given by

$$\lambda_H(k,n,N,CL) = X_{\max}(k)/SI \ . \qquad (4.6)$$

$\lambda_H(k,n,N,CL)$ is the expected number of errors in a sample of size *n* drawn from a population of size *N* in which there are a total of $X_{\max}(k)$ errors. Note that the Poisson factor $\lambda(k,CL)$ as defined in Section 3 is a conservative approximation to the hyper geometric risk factor as defined above, with the accuracy of the approximation improving, as the population size grows large.

Incorporation of Errors in High Value Items

We now describe the incorporation of errors in high value amounts and the calculation of the final Gross and Net error projections appearing on the result tab. The equations used are:

HVE o/s = $\sum_{RA(i)>AA(i);} RA(i) - AA(i)$; *i* in HV database (4.7a)

HVE u/s = $\sum_{AA(i)>RA(i);} AA(i) - RA(i)$; *i* in HV database (4.7b)

Gross UEL o/s Whv = *Gross UEL o/s noHV + HVE o/s* (4.7c)

Gross UEL u/s wHV = *Gross UEL u/s noHV + HVE u/s* (4.7d)

Gross MLE o/s wHV = *Gross MLE o/s noHV + HVE o/s* (4.7e)

Gross MLE u/s wHV = *Gross MLE u/s noHV + HVE u/s* (4.7f)

Net UEL o/s noHV = *Gross UEL o/s noHV – Gross MLE u/s noHV* (4.7g)

Net UEL u/s noHV = *Gross UEL u/s noHV – Gross MLE o/s noHV* (4.7h)

Net MLE o/s noHV = *Gross MLE o/s noHV – Gross MLE u/s noHV* (4.7i)

Net MLE u/s noHV = *Gross MLE u/s noHV – Gross MLE o/s noHV* (4.7j)

Net UEL o/s wHV = *Gross UEL o/s wHV – Gross MLE u/s wHV* (4.7k)

Net UEL u/s wHV = *Gross UEL u/s wHV – Gross MLE o/s wHV* (4.7l)

Net MLE o/s wHV = *Gross MLE o/s wHV – Gross MLE u/s wHV* (4.7m)

Net MLE u/s wHV = *Gross MLE u/s wHV – Gross MLE o/s wHV* (4.7n)

Precision o/s = *Gross UEL o/s noHV – Gross MLE o/s noHV* (4.7o)

Precision u/s = *Gross UEL u/s noHV – Gross MLE u/s noHV* (4.7p)

4.2 Combined Evaluation of Multiple Samples

Inputs:

Y(*k*):	Total monetary value of population from which k^{th} error in total sample was drawn.	$\lvert Y \rvert > n$
n(*k*):	Size of component sample in which k^{th} error in combined sample was found (obtained from sample databases)	$n > 0$

BPP(*k*):	Basic precision pricing corresponding to k^{th} error in combined sample.	$0 < BPP < 10$
CL	Confidence level (a.k.a. reliability)	$0 < CL \leq .995$
RA(*k*)	Recorded amount for k^{th} record in combined sample, for $k = 1 \ldots n$ (obtained from sample databases)	$RA(k) \neq 0$
AA(*k*)	Audited amount for k^{th} record in main sample, for $k = 1 \ldots$ sum of sample sizes (obtained from sample databases)	$\left\lvert \frac{RA(k) - AA(k)}{RA(k)} \right\rvert \leq BPP(k)$

Outputs:

Gross UEL o/s NoHV : upper precision limit of overstatements projected from main sample, disregarding understatements and not including errors in HV items

Gross UEL u/s NoHV: upper precision limit of understatements projected from main sample, disregarding overstatements and not including errors in HV items

Gross MLE o/s NoHV: most likely error for understatements projected from main sample, disregarding understatements and not including errors in HV items

Gross MLE u/s NoHV: most likely error for understatements projected from main sample, disregarding overstatements and not including errors in HV items

Net UEL o/s NoHV: upper error limit for overstatements net of understatements not including errors in HV items

Net UEL u/s NoHV:	upper error limit for understatements net of overstatements not including errors in HV items
Net MLE o/s NoHV:	MLE for overstatements net of understatements not including errors in HV items
Net MLE u/s NoHV:	MLE for understatements net of overstatements not including errors in HV items
HVE o/s:	total overstatements in high value items
HVE u/s:	total understatements in high value items
Gross UEL o/s wHV:	upper error limit for overstatements, disregarding understatements but including errors in HV items
Gross UEL u/s wHV:	upper error limit for understatements, disregarding overstatements but including errors in HV items
Gross MLE o/s wHV:	most likely error for overstatements, disregarding understatements but including errors in HV items
Gross MLE u/s wHV:	most likely error for understatements, disregarding overstatements but including errors in HV items
Net UEL o/s wHV:	upper error limit for overstatements net of understatements, including errors in HV items
Net UEL u/s wHV:	upper error limit for understatements net of overstatements, including errors in HV items
Net MLE o/s wHV:	MLE for overstatements net of understatements, including errors in HV items
Net MLE u/s wHV:	MLE for understatements net of overstatements, including errors in HV items
Precision o/s:	Total precision of UEL for overstatements
Precision u/s:	Total precision of UEL for understatements

Other quantities referred to below:

SI(*k*):	average sampling interval for component sample in which k^{th} error in main sample was found.
RA(*k*):	recorded amount for k^{th} record in combined sample
AA(*k*):	audited amount for k^{th} record in combined sample
$t_{o/s}(k)$:	tainting for the k^{th} overstatement in combined sample, ordered from largest to smallest.
$t_{u/s}(k)$:	tainting for the k^{th} understatement in combined sample, ordered from largest to smallest.
$x_{o/s}$:	total number of overstatements in combined sample
$x_{u/s}$:	total number of understatements in combined sample
N(*k*):	size of population from which sample *k* was drawn. For MUS, $N(k) = \left\| Y(k) \right\|$.
N_t:	size of combined population: $N_t = \sum N(k)$
n(*k*):	size of sample *k*
n_t:	size of combined sample: $n_t = \sum n(k)$

Overview:

Suppose a monetary unit sample is taken in each of several distinct populations. IDEA uses the "global combined bound" proposed by Neter et al. [3] to determine the most likely error and upper error limit for the combined population – that is, the union of all of the sampled populations. This bound reduces to the bound introduced by K. Stringer, for the case of a single sample. The Stringer bound is most appropriate for samples containing few errors[1]; it tends to over-estimate the upper error limits when many errors are observed.

The most likely and upper error limits are computed separately for overstatements and understatements. The resulting (gross) estimates are then combined to produce a net most likely error and a net upper error limit for the combined population. The hyper geometric probability distribution is used throughout.

Note that only the separate (Gross) upper error limit for overstatements has been extensively tested. The guidance of a statistical expert should be sought before the gross and net most likely errors, or the net upper error limit, are used in any audit decision.

The Stringer bound is slightly more conservative than that produced by the cell evaluation method. However, unlike the cell evaluation method, the Stringer bound is valid for monetary unit samples drawn using either cell selection or simple random sampling. IDEA's "combined evaluation" option can also be used to obtain the Stringer UEL for a single sample, if this is desired.

We note that [3] does not consider the use of BPPs other than 100%. In IDEA, we have extended the "worst-case" basic precision proposed in [3], to allow for arbitrary BPPs.

Details:

Sampling Interval:

The sampling intervals are computed as

$$SI(k) = N(k) / n(k) \tag{4.8}$$

See comments in Section 4.1 for more details.

Taintings:

Taintings are calculated for each record in the combined sample as in Eq. (4.1a), and sorted into groups of overstatements and understatements, $t_{o/s}(k)$ and $t_{u/s}(k)$, as in E.g. (4.1b). The index k of a tainting denotes its position in a list of taintings from all samples, sorted from largest to smallest.

Projection of Most Likely Errors from Sample

The formula used for computing the most likely total overstatement in the combined population, based on the overstatements observed in all samples, is

$$\textit{Gross MLE o/s NoHV} = \sum_{k=1}^{x_{o/s}} t_{o/s}(k) \times SI(k)\ . \tag{4.9}$$

Here $t_{o/s}(k)$ is the tainting of the k^{th} tainting in the combined sample, ordered from largest to smallest. Taintings are described further in Section 4 – *taintings*. Note that (4.9) does not include errors in high value amounts stored in separate high value files – these errors are added onto *Gross MLE o/s NoHV* in computing the final error estimates (see *Calculation of Gross and Net Error Projections*).

Projection of Upper Error Limits from Sample

We here describe the procedure for projecting the upper error limits from the sample, without consideration of errors in high value amounts. We shall provide details for $Gross\ UEL_{o/s}\ noHV$; the computation of $Gross\ UEL_{u/s}\ noHV$ is directly analogous. All errors and taintings in the equations below should be understood to be overstatements. The upper limit is given by

$$GrossUEL\,o/s\ \ NoHV = BP + \sum_{k=1}^{x} PGW(k)\,. \tag{4.10}$$

The first term is the basic precision for the combined sample. For conservatism, it is taken to be the largest basic precision of all of the component samples:

$$BP = \max_{k}\left[\lambda_H\left(0, n_t, N_t, CL\right) \times BPP(k) \times SI(k)\right] \tag{4.11}$$

PGW(*k*) is the precision gap widening for the *k*th tainting in the combined sample. It is given by

$$PGW(k) = \left[\lambda_H\left(k, n_t, N_t, CL\right) - \lambda\left(k-1, n_t, N_t, CL\right)\right] t(k)\ SI(k) \tag{4.12}$$

In (4.11) and (4.12), $\lambda_H\left(k, n, N, CL\right)$ is the hyper geometric factor, as defined by Eq.s (4.5), (4.6)[2] Also, n_t and N_t are the combined sample and population sizes, and *t*(*k*) and *SI*(*k*) are the tainting and sampling interval corresponding to the *k*th tainting in the combined sample.

Calculation of Gross and Net Error Projections

The equations for computing the final Gross and Net error projections appearing on the result tab are exactly as Eq.s (4.7), (4.8), except that the misstatements in high value amounts are totaled over all specified high value files.

Footnotes:

1. For example, [3] describes extensive simulations with combined samples having a total of between 10 and 50 overstatements; for a specified target confidence level of 95%, the achieved “empirical” confidence level (that is, the frequency with which the bound exceeded the actual population error) was always above 99%.

2. Note that the combined population and sample sizes are used in computing these hyper geometric factors. This prevents precision gap widening from becoming negative, which could occur if component sample and population sizes were used. Though the factors thus computed are slightly larger than would be those computed with component sample and population sizes, they are still more efficient than the corresponding Binomial or Poisson approximations.

4.3 Single Sample Evaluation when More than 20 Errors are Found

Inputs:

Y(*k*):	Total monetary value of population from which *k*th error in total sample was drawn.	$\lvert Y \rvert > n$
n(*k*):	Size of component sample in which *k*th error in combined sample was found (obtained from sample databases)	$n > 0$
Limits	Desired precision interval endpoints	*Upper* or *Both*
CL	Confidence level (a.k.a. reliability)	$0.1 \le CL \le .995$ if *Limits* = *Both*, or $0.55 \le CL \le .995$ if *Limits* = *Upper*.
RA(*k*)	Recorded amount for *k*th record in combined sample, for *k* = 1 … *n* (obtained from sample databases)	*RA*(*k*) ≠ 0 AND all *RA*(*k*) have same sign
AA(*k*)	Audited amount for *k*th record in main sample, for *k* = 1 … sum of sample sizes (obtained from sample databases)	

Outputs:

UEL NoHV : upper precision limit of overstatements projected from main sample, disregarding understatements and not including errors in HV items

LEL NoHV[1]: lower precision limit of overstatements projected from main sample, disregarding understatements and not including errors in HV items

MLE NoHV: most likely error for understatements projected from main sample, disregarding understatements and not including errors in HV items

HVE: total overstatements in high value items

UEL wHV: upper error limit for overstatements, disregarding understatements but including errors in HV items

LEL wHV[1]: lower precision limit of overstatements projected from main sample, disregarding understatements and not including errors in HV items

MLE wHV: most likely error for overstatements, disregarding understatements but including errors in HV items

Precision: Total precision of UEL for overstatements

s_r : Estimated standard deviation of ratios

Other quantities referred to below:

RA(*k*): recorded amount for k^{th} record in main sample

AA(*k*): audited amount for k^{th} record in main sample

r(*k*): ratio of for the k^{th} error in main sample, ordered from largest to smallest.

$\bar{r}$: Average ratio of audited amount to recorded amount

x : total number of errors in main sample

$U(R)$: Reliability factor. The reliability factor is that value such that the area under the probability distribution function from $-\infty$ to $U(R)$ is equal to *R*.

Description:

The high-error evaluation uses Classical Proportional to Size (PPS) sampling theory [4] to estimate the most likely error in the population, the upper error limit, and optionally the lower error limit. The Student's *t* probability distribution is used throughout.

In contrast with the low-error rate method, separate estimates are not computed for each of over and understatements. Rather, estimates are based on both over and understatements. The following facts should hold in order for this method to be suitable [4]:

1. All recorded amounts must have the same sign (either positive or negative are allowed).
2. If positive recorded amounts are used, the majority of the errors should be overstatements, and any understatements should be small. One or more large understatements can inappropriately dominate the projections. If negative recorded amounts are used, the majority of errors should be understatements.
3. It is recommended that the sample should contain at least 20 errors in order for the assumptions on the sampling distribution to hold [4]. At least two errors are required in order to perform the calculation.

Since both low and high-error samples are extracted in the same way, a sample planned with the low-error (cell-based) planning function can be evaluated with the high-error (classical PPS) evaluation function.

Ratios, Mean Ratio, and Standard Deviation

For each item in the main sample database, a ratio is computed as

$$r(k) = \frac{AA(k)}{RA(k)} \; . \tag{4.13}$$

Classical PPS theory has not been tested for cases in which negative ratios are present (i.e., there are audited amounts having different sign from the corresponding recorded amounts). IDEA generates a warning if any negative ratios are encountered. Similarly, ratios greater than one can dominate the results; such ratios result from understatements on positive recorded amounts, or overstatements on negative recorded amounts. IDEA generates a warning whenever a ratio of greater than 2.0 is encountered. (A ratio of 2 results from 100% understatement of a positive recorded amount or 100% overstatement of a negative recorded amount.)

The mean of ratios for all items is

$$\bar{r} \;=\; \tfrac{1}{n}\sum_{k=1}^{n} r(k) \tag{4.14}$$

Note that negative ratios are allowed and are not inconsistent with the assumed normality of the sampling distribution. However, the behavior of the estimate when negative ratios are present is not well characterized by empirical tests; hence IDEA generates a warning whenever negative ratios are encountered.

The estimated standard deviation of the ratios is

$$s_r = \sqrt{\frac{1}{n-1}\left(\sum_{k=1}^{n} r^2(k) - n\bar{r}^2\right)} \tag{4.15}$$

Projection of Most Likely Errors from Sample

We here describe the procedure for projecting the most likely errors from the sample, without consideration of errors in high value amounts. The formulae used is

$$\mathit{MLE\ NoHV} = N \times \bar{r} \tag{4.16}$$

Projection of Error Limits from Sample

We here describe the procedure for estimating either the upper limit or upper and lower limits on the monetary value of the total error in the population. We first compute a reliability factor[2] $U(R)$ from the Student's *t* tables. The degrees of freedom used for this computation is *df* = *x* – 1. For a one sided interval, the reliability is $R = CL$ For a two-sided interval, the reliability is $R = 1 - \frac{1-CL}{2}$.

The precision is computed using the following formula:

$$\mathit{Precision} = \frac{U(R)\, N\, s_r}{\sqrt{n}} \tag{4.17}$$

The limits are then:

UEL NoHV	= *MLE NoHV* + (*Precision* × sgn(*MLE NoHV*))	(4.18a)
LEL NoHV [1]	= *MLE NoHV* – (*Precision* × sgn(*MLE NoHV*))	(4.18b)
UEL wHV	= *UEL NoHV* + *HVE*	(4.18c)
LEL NoHV [1]	= *LEL NoHV* + *HVE*	(4.18d)

Note that sgn(*x*) = –1 is *x* is negative, and sgn(*x*) = +1 if x is positive.

Footnotes:1 Computed only if *Limits* = *Both*

4.5 Warning Messages from Evaluation

IDEA displays notification or warning messages in the sample result, if certain conditions are encountered during MUS Evaluation. The table below shows these conditions and the resulting messages.

Condition	Warning
n > 0.5 × number of records in population database	Your sample size exceeds half the number of records in your population database. In these circumstances, the upper error limit projected by an MUS evaluation is conservative. Classical variables sampling may be more statistically efficient for the population you are considering.
Change to *BPP Low-error single evaluation only*	Warning: The Basic Precision Pricing (BPP) has been changed since this sample was planed. This happens whenever tainting is found in the sample that exceeds the planned BPP.
Sample database information shows that high value items were extracted to a file, but no HV file is specified for the evaluation.	High value items for this population were extracted to the file [*High Value Filename*], but no high value file was specified for the evaluation. The above results do not include errors in high value items.
Sample was not generated with IDEA's MUS Plan – Extract AND recorded amounts exceeding the sampling interval were found in the sample.	Warning: One or more recorded amounts exceeding the sampling interval have been found in the file [sample database name].
Sample was generated with IDEA's MUS Plan – Extract, AND HV items were extracted to a separate database, AND HV item found in sample.	Warning: One or more recorded amounts exceeding the high value amount have been found in the file [sample database name].
$t(k) > 1$ for some *k*. *Low-error single evaluation and combined evaluation only*.	Warning: one or more taintings of greater than 100% were found in the sample. Consultation with a statistical expert is recommended.

$r(k) > 2$ for some *k*. *High-error single evaluation only*.	Warning: one or more large understatement were found in the sample. Consultation with a statistical expert is recommended.
$r(k) < 0$ for some *k*. *High-error single evaluation only*.	Warning: one or more audited amounts whose signs differ from corresponding book values were found in the sample. Consultation with a statistical expert is recommended.
$x < 20$. *High-error single evaluation only*	Warning: This sample contains fewer than 20 errors. You have selected high error rate evaluation, which is recommended for samples containing 20 or more errors. You may wish to use the low error rate evaluation instead

References

[1] Canadian Institute of Chartered Accountants, *Users' Guide: IDEA for Windows Version 2*, CICA Press, 1997.

[2] D.A. Leslie, A.D. Teitlebaum, R.J. Anderson, *Dollar Unit Sampling a Practical Guide for Auditors*. Copp, Clark Pitman, Toronto, 1979.

[3] J. Neter, H.S. Kim and L.E. Graham, "On combining Stringer bounds for independent monetary unit samples from several populations." *Auditing*, v. 4, 1984. pp. 74-88.

[4] Donald M. Roberts, *Statistical Auditing*, AICPA 1978; p. 184.

Notes

Notes

Notes