



have structure and complexity - data that changes over time in content and volume and there are usually 'imperfections and exceptions' to deal with. Excel is a universal tool in business and other organisations BUT completing multi-stage tasks (business flow) in spreadsheets and working with structured data very quickly becomes unwieldy and increasingly cumbersome to maintain and test, let alone automate. Grappling with intricate cell formulae requires feats of mental gymnastics and what you thought was a simple objective grows in scope and complexity. If you need to learn how to use macros and Visual Basic (VB) you are really pushing spreadsheets beyond what they were designed for. Some business users do develop highly functional but intricate spreadsheet solutions but you will find that your IT Team are rarely prepared to support them.

Using Database technology?

'Big Data' techniques aside (covered in a separate KBL article), technology to store and access highly structured data has changed little in 25 years. RDBMS (Relational Database Management Systems) is the predominant technology with SQL (Structured Query Language) from the 1980s, still the main tool for data extraction and manipulation. You will find Microsoft's SQLServer and Oracle technology used in the majority of medium to large companies and organisations.

For data located in a database, a query with just a few tables, JOINS, WHEREs and maybe a GROUP clause, a SQL script will look elegant but 'queries' very soon become complicated and inefficient when additional logic is added. SQL is fine for set reports, heavy-duty transactional systems and data warehouses but lacks flexibility so is unsuitable as an 'end-user' tool for a busy office environment where data and processes can be messy and work deadlines are tight. If you use SQL, you are probably an IT professional and even you will need to include other tools and languages to complete IT tasks. For the adventurous business user, SQL is available in Microsoft Access but this is not an easy tool for the office as it too requires high technical skill levels and is not built to handle procedural logic – furthermore, few IT departments will support MS Access-based applications.



“Business IT ranges from the simple spreadsheet task rapidly changing to a scenario where there’s no option but to seek specialist IT help and resources. There’s precious little choice of business tool for fulfilling requirements in between. Why is this? Objectively speaking, traditional IT is inflexible and serves business poorly for what it costs. If you are dependent on IT, the gap between business and technology poses an increasingly worrying dilemma and shows little sign of being addressed by big technology providers”

Enter KBL

Kegsoft's KBL language and Studio have been designed top-down, completely from scratch to address this yawning gap between office business user IT capability and costly, specialised IT resources. Kegsoft takes a radically different approach as logic is not written with incomprehensible cell formulae or heavy chunks of program code. Instead, KBL is composed in English-style scripts to thread together and describe business-flow in easily understood manageable steps. KBL can be written by able business-focussed users and, most importantly, KBL logic can easily be peer reviewed, tested and run by other business-focussed users.

KBL does not replace or try to compete with spreadsheets, SQL or the other IT foundation languages. KBL does, however, introduce a simpler, swifter and flexible alternative for executing nuanced business logic to manipulate data and to interact more intuitively with users. It is designed to be used and understood by *business users*, not highly specialised IT practitioners. It's not a magic wand or universal panacea as KBL skills do need to be learned and developed, but it is easily mastered by most people who can use spreadsheets and understand business processes. KSE (Keg Studio Express) enables you to build simple user-interfaces and create end-to-end, tactical office applications quickly. After gaining experience in KBL, users may wish to upgrade to the Studio Plus (KSP) version which has extra command sets for creating more strategic, line-of-business applications that can be deployed and shared with end-users as KSP can access and share local, cloud and hybrid data sources.

“KBL introduces a simpler alternative to executing nuanced business logic that manipulates data and interfaces with users. It's designed for business-oriented users as well as IT practitioners”

For the data-savvy, KBL can perform relational-style operations on and between spreadsheets and other data files of any size and varying format. This ability extends what business-oriented users can do with their spreadsheets and other data sources – before the need for engaging specialist IT resources. KBL enables users to think about how they need to work their data and processes without worrying about the underlying technology.

A Typical Business Example

Here is a typical business IT example that requires a series of relatively simple data analysis and process steps. In order to boost sales, a mid-sized company wants to leverage its customer base and mail past customers with its latest product offers. The company does not have a specialist CRM application but can easily extract customer and order transaction data needed for the analysis.

- The marketing budget is limited so you'll start by identifying your 'best customers' and contact them by their preferred mailing method. Incremental costs are known: emails cost £0.05, a posted mini-brochure £0.85 in the UK (more for overseas)
- Some customers prefer contact by email, others prefer posted material, others don't want to be contacted at all
- Depending on mailing volumes and calculated costs, you may decide to mail in tranches – If the first tranche is successful, it could warrant budget for further tranches
- You think your data is good – but not perfect and in the past, have noticed duplicate and incomplete customer records as well as cancelled/returned orders which have been kept on file – your mailing campaign should identify and reject substandard data by applying simple quality rules
- However, you would like to repair and use some sub-standard customer data where practicable
- You have two key data source (files) to work with Customer details and Past Order details
- The marketing manager is pro-active and likely to adjust who to mail and who not to mail – e.g. for cashflow purposes, prepay customers might be prioritised over those with accounts with credit terms. Recent customers will have already received brochures and so to save

money, could be excluded from postal mailings. Other areas to explore might include targeting customers who bought particular products

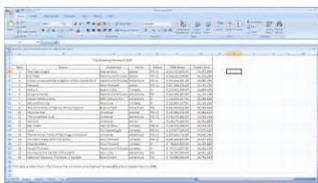
- You will need to provide volumetric data for requested scenarios
- All mailings – postal and electronic will be fulfilled by a mailing house who have prescribed formats for the separate mailing files
- You need the ability to manually exclude certain customers from the mailing and make sure you're not breaking any data-protection / privacy code of conduct for certain individuals
- You want the ability to repeat the process, maybe with slightly different parameters, several times a year, so the method you develop should enable you to easily repeat the exercise
- You will need to produce summary stats of the mailings and calculated costs
- When mailing is completed, you would like a means to assess the effectiveness of the 'campaign' from customer responses

These are quite typical office requirements that need IT methods to fulfil them; however, when drawn out on a flowchart, you have quite a collection of logic flow to deal with. Management interaction will very likely make small changes to selection criteria and it's likely that the marketing manager will want to play out several scenarios to get the most from the limited budget ('bang for bucks'), so your approach must be flexible and adaptable.

Getting the job done

The basic logic to determine mailing priority is to sum value of orders by customer – checking for earliest and latest order dates and sort customers by descending value. Then you combine this information with filtered customer details. You will need to take into account preferred contact method and eliminating any duplicates, 'do not market', excluded and incomplete addresses etc. (where you can, you'll repair rejected customer records). You'll work out cumulative costs (postage and emailing), depending on the selection criteria so you can decide the affordable cut-off points for the population mailed. Finally, when the precise criteria and break-points are decided, you can create the mailing files with campaign codes and create a master list of mailing activities. The master file will help process any future tranches or campaigns and, if needed, be updated with customer responses (for analysis purposes).

Using Spreadsheets?



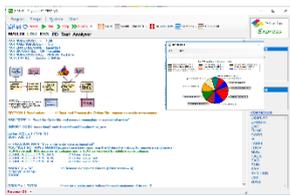
Using spreadsheets, you have your work cut out and will certainly be busy with VLOOKUPS and a number of intermediate worksheets. It will be a highly manual / disjointed operation to process the business flow that the tasks imply. Grouping and joining data, especially where the quality is questionable, and where there are subtle exceptions, is problematic. Doing the work in tranches and repeating the exercise several times a year adds to the administrative load. Corrections can be done on separate worksheets but validation and matching with historical orders requires high levels of Excel skill.

Using SQL / database technology?



The core task lends itself well to a relational data approach. In SQL, we'd need to design some input and intermediate tables to progress through the series processing steps, the *underlying* stage should be easily dealt with by joining Orders and Customers data – either using original data tables or loading provided files into database tables. With some 'Where' and 'Group' clauses candidate sets can be created and further operated on. The work will need to be completed in several scripts to handle intermediate stages and analytical what-ifs. Exception processes are not easily dealt with – especially where human intervention is needed to decide on outcome. Data correction and exclusions will not be done in SQL – but possibly using a user spreadsheet to input a sub-population of customers. Both the SQL programmer and the marketing manager might find it frustrating working together as it takes time to amend queries to produce information for alternative scenarios. The outputs from the exercise is likely to be a series of SQL scripts and the final data presented in spreadsheet form.

With KBL and Studio Express?



Writing scripts In KBL for this task is straightforward and can be dealt with in two reusable scripts – one to produce the cost figures and actual mailing lists – and another to correct poor data / exclude selected customers. The mailing script breaks down the task into consecutive business flow steps with all controls (mailing costs, budget, cut-off dates, import/export file names etc.) 'parameterised' at the start via an input form. Customer and Orders data are imported and pre-processed in consecutive sections of the script, then merged when necessary to sequence details of customer 'value'. Adding cumulative costs by mailing type to determine the budget spent point couldn't be simpler. For subsequent tranches, just re-run the script with different cut-off parameters and campaign code. And, the marketing manager would have little difficulty checking the script to see if it met the business expectations. Graphical output for visual evaluation would be included in the scripts.

KBL Studio Express (KSE) is aimed at tactical task solving with the ability to create relatively simple applications where repetition is involved. It contains a full set of KBL commands that will cover most IT problems that are likely to be met in the office.

KBL Studio Plus (KSP), for release in early 2018, has everything that KSE has but with the addition of full line of business application building features. Access and share Cloud and Hybrid data, create end-user, browser-based applications. KSP is suited to more experienced KBL users who wish to further exploit their script writing skills and progress more strategic tasks. KSP shares the same IT simplification and productivity objectives as KSE – you won't need to be an IT guru to create KBL-based applications.

Peter Domanski is Co-founder of the KBL business language and joint author of the textbook 'A Practical Guide to Relational Database Design'