

PROGRAMMER EN VBA

Connaître l'historique d'Excel	13
Découvrir l'évolution des macros et de la programmation	21
Respecter quelques principes de développement	23
Aperçu rapide des différents concepts et outils	29

VBA (*Visual Basic pour Application*) est le langage de programmation associé aux outils de la suite Office de Microsoft. Il permet d'automatiser un certain nombre de tâches et de traitement, à l'aide de "macros".

Le terme "macro" est l'abréviation de "macro-commande", qui peut se comprendre comme "suite de commandes". Aujourd'hui, cette appellation est un peu réductrice lorsque l'on se penche sur les possibilités offertes par ce véritable langage de développement. Il est en effet possible de réaliser de véritables applications d'entreprises, fiables et performantes. Il serait donc plus logique de parler de "programmes".

Dans ce chapitre, nous ferons un rapide tour d'horizon de l'évolution d'Excel puis nous nous intéresserons plus spécifiquement à l'évolution des macros et de la programmation. Ensuite, nous tenterons de définir quelques principes de développement. Enfin, nous passerons en revue les principaux concepts et outils développés dans la suite de cet ouvrage.

1.1. Connaître l'histoire d'Excel

Tableur est la traduction de l'expression anglaise *electronic spreadsheet*, le terme *spreadsheet* désignant une grande feuille de papier, divisée en lignes et en colonnes et utilisée pour présenter les comptes d'une entreprise. Un tableur est donc la version informatisée d'un tableau de calcul.

Les débuts du tableur

Les prémices du tableur remontent à 1961, avec la création par Richard Mattesich (professeur à l'université américaine de Berkeley) d'un *computerized spreadsheet* écrit en Fortran et fonctionnant sur un "gros" système. Les débuts du tableur, proprement dit, peuvent être datés de 1978 avec la mise au point, en Basic, de ce qui allait devenir Visicalc, par un étudiant de Harvard, Daniel Bricklin.

Ce dernier doit réaliser des tableaux comptables pour une étude de cas sur Pepsi-Cola. Il préfère se simplifier la vie en réalisant un programme lui permettant d'effectuer rapidement des calculs. En mai 1979, le logiciel VisiCalc, issu du prototype de Daniel Bricklin, fait son apparition sur le marché. Il est vendu 100 dollars et peut fonctionner sur l'Apple II. Le succès est rapide car ce logiciel met à la portée de

tous ce qui était auparavant réservé aux seuls programmeurs. Son apparence est d'ailleurs voisine des tableurs actuels (la souris en moins).

En 1982, sort le premier tableur de Microsoft : Multiplan pour compatibles PC. Il ne connaît pas un grand succès aux États-Unis.

En 1983, Lotus 1-2-3 fait son apparition, développé par une équipe de programmeurs "dissidents" de VisiCorp, emmenés par Mitch Kapor. Lotus 1-2-3 supprime bientôt VisiCalc que ses concepteurs, englués dans des querelles internes, ne sauront pas faire évoluer suffisamment rapidement. Ce nouveau logiciel apportait des améliorations indéniables : possibilités graphiques, gestion de bases de données, macros.

L'année 1984 voit la mise sur le marché de la première version d'Excel pour le Macintosh d'Apple, dont il met à profit l'interface graphique et la souris. Cela rend ce logiciel plus populaire que son prédécesseur Multiplan. Excel contribue au très grand succès du Macintosh (et réciproquement).

En 1987, Microsoft met sur le marché la version PC d'Excel. Jusqu'en 1992, ce sera le seul tableur disponible sous Windows, lui assurant ainsi la domination du marché au détriment de Lotus 1-2-3, malgré quelques assauts de Borland (Quattro) et de Computer Associates (SuperCalc).

L'évolution d'Excel depuis Excel 2

La version actuelle d'Excel dont la dénomination commerciale est Excel 2010 a pour nom Excel 14. Pourtant, il ne s'agit que de la onzième version. Une petite mise au point s'impose.

Comme nous l'avons vu, la première version d'Excel pour Windows date de 1987 ; il s'agit d'Excel 2. Pourquoi Excel 2 ? Tout simplement pour correspondre à la version Macintosh (puisque Macintosh était précurseur).

Excel 3 voit le jour en 1990. Cette version bénéficie d'améliorations au niveau des fonctionnalités comme de l'ergonomie. Citons, entre autres, les barres d'outils, les graphiques en 3D, les dessins.

En 1992, accompagnant l'essor commercial de Windows, Excel 4 fait son apparition. Mettant l'accent sur la facilité d'utilisation, cette version est davantage accessible pour les novices.

La version Excel 5 apparaît, quant à elle, en 1994. Elle constitue un véritable bond en termes de fonctionnalités, avec notamment les

classeurs multifeuilles. Le développement des macros est grandement amélioré grâce à l'apparition de VBA (*Visual Basic pour Application*).

Première version "millésimée", Excel 95 (en fait Excel 7) voit le jour en 1995. Sa principale nouveauté (peu visible pour l'utilisateur) est l'emploi du code 32 bits.

Excel 97 (Excel 8) constitue réellement une avancée par rapport aux précédentes. Les barres d'outils et les menus changent d'apparence. Le nombre de lignes maximal dans une feuille de calcul est multiplié par 4. L'aide en ligne et le langage VBA progressent également largement. Enfin, cette version introduit un nouveau format de fichier.

Excel 2000 (Excel 9), apparu en 1999, autorise, entre autres légères améliorations, l'enregistrement au format HTML.

Excel 2002 (Excel 10) est commercialisé en 2001 au sein de la suite Office XP. Cette version est axée sur la facilité de prise de main par le débutant (bouton d'options, balises actives...), le travail collaboratif (révision de documents en circuit, signature numérique...) et la fiabilité de récupération des données en cas de plantage. Le volet Office fait également son apparition.

Excel 2003 (Excel 11) prend la suite à partir d'octobre 2003. Cette version propose une ergonomie légèrement remaniée (barres d'outils redessinées). Les nouvelles fonctionnalités sont principalement :

- l'apparition de volets Office supplémentaires (accueil, gestion des classeurs, recherche...) ;
- la possibilité de créer et de gérer des listes de données (véritables "mini bases de données" au sein des feuilles de calcul) ;
- la comparaison de classeurs en côte à côte (défilement synchronisé de deux feuilles de calcul) ;
- l'amélioration de certaines fonctions statistiques (améliorations dans la précision et l'arrondi des résultats) ;
- la gestion de l'accès à l'information (gestion des droits d'accès aux données grâce au module *Information Rights Management*) ;
- le partage de documents (utilisation des sites Microsoft Windows SharePoint Services afin de faciliter le travail collaboratif autour d'un ensemble de documents Office) ;
- la prise en charge de XML (*eXtensible Markup Language*, langage de balisage extensible).

Excel 2007 : la rupture

La version 12 d'Excel, commercialisée sous le nom d'Excel 2007, marque une rupture avec les versions précédentes concernant l'ergonomie. Un nombre croissant d'utilisateurs estimait en effet que les menus devenaient parfois inextricables, rendant complexe l'accès à certaines fonctions. Les concepteurs d'Excel 2007 ont donc répondu à cette préoccupation en changeant radicalement l'accès aux diverses fonctions.

L'interface, qui reposait jusqu'alors sur des menus et des barres d'outils, a été remplacée par un Ruban constitué d'onglets organisés en fonction de l'action à accomplir. Excel dispose par exemple d'onglets tels que **Mise en page**, **Insertion** ou **Formules** qui regroupent les commandes selon les tâches que vous souhaitez accomplir.

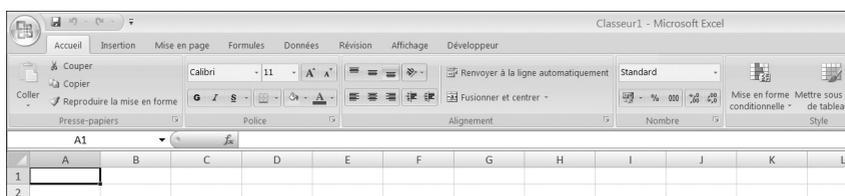


Figure 1.1 : Les onglets de commandes d'Excel 2007

Ces onglets sont associés aux galeries qui vous proposent de choisir parmi un ensemble de résultats potentiels, plutôt que de spécifier des paramètres dans des boîtes de dialogue.

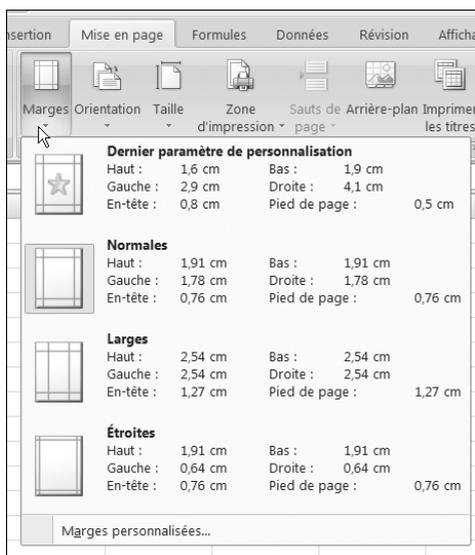


Figure 1.2 : Une galerie

Une fonction d'aperçu instantané permet de visualiser directement, sur le document, l'effet d'une modification ou d'une mise en forme lorsque vous déplacez le pointeur de votre souris sur les résultats proposés dans une galerie.

Excel 2007 recèle bon nombre d'autres nouveautés :

- Les feuilles de calcul comprennent désormais 1 048 576 lignes (contre 65 536 auparavant) et 16 384 colonnes (contre 256 auparavant). Cela multiplie la capacité de stockage d'une feuille par 1 024. Les colonnes sont à présent "numérotées" de A à XFD.
- Le tri des données a été également considérablement amélioré. Vous pouvez trier selon 64 critères (3 auparavant) et trier selon la couleur.
- Le filtre automatique est lui aussi plus performant, dans la mesure où vous pouvez, entre autres, spécifier plusieurs critères sur une même colonne.
- La fonction de mise en forme conditionnelle, déjà très performante, subit une véritable révolution. Cette fonction permet d'identifier encore plus facilement les tendances, les valeurs extrêmes d'un ensemble de données à l'aide de nuances de couleurs, de barres de données et même d'icônes. Bon nombre de conditions "classiques" sont déjà prédéfinies ; il suffit de les sélectionner dans une galerie pour les appliquer.

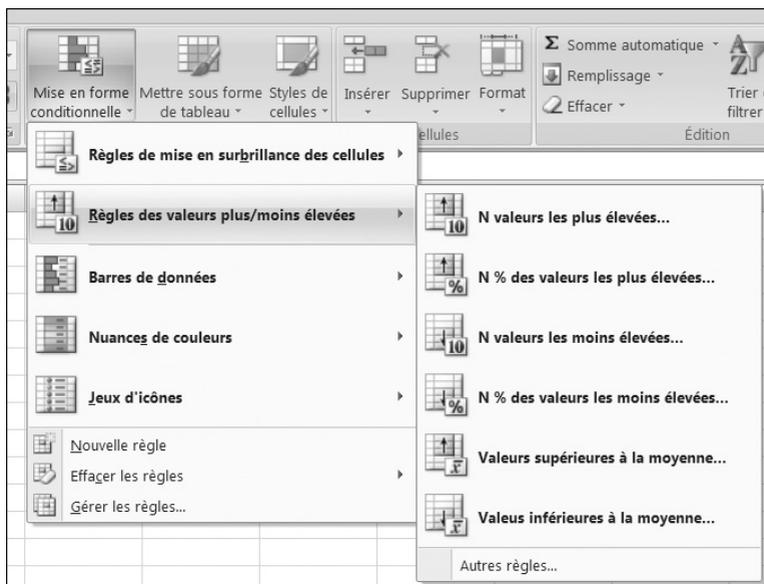


Figure 1.3 : La mise en forme conditionnelle a considérablement évolué

- La création et la mise en forme des graphiques ont été considérablement simplifiées et enrichies. La bibliothèque de graphiques prédéfinis a été très largement remaniée et étendue.
- Le nouveau format de fichiers est appelé Microsoft Office Open XML. Ainsi les classeurs Excel 2007 auront pour suffixe *.xlsx*. Ce nouveau format est fondé sur les formats *XML* et *ZIP*. Il autorise une meilleure intégration des données externes et permet de réduire la taille des classeurs. Si Excel 2007 permet d'ouvrir sans problèmes les fichiers créés avec les versions précédentes, il faudra en revanche installer des mises à jour sur ces versions pour ouvrir les fichiers enregistrés au format *Microsoft Office Open XML*.
- Excel 2007 vous permet également d'enregistrer vos classeurs au format *PDF* ou *XPS*. Vous créez ainsi rapidement des versions de vos tableaux non modifiables, simples à mettre en ligne et à transmettre par mail.

Excel 2010 en quelques mots

Excel 2010, nom commercial d'Excel 14, s'inscrit dans la lignée de la précédente version en ce qui concerne notamment l'ergonomie et les formats de fichiers. En effet, cette nouvelle version arbore toujours le Ruban, découvert avec 2007.

Le Ruban a été modifié pour intégrer un menu **Fichier**, qui permet d'accéder au nouveau mode Backstage. Ce dernier permet d'accéder aux fonctionnalités principales de gestion des fichiers, d'impression, de partage des documents et de définition des options.

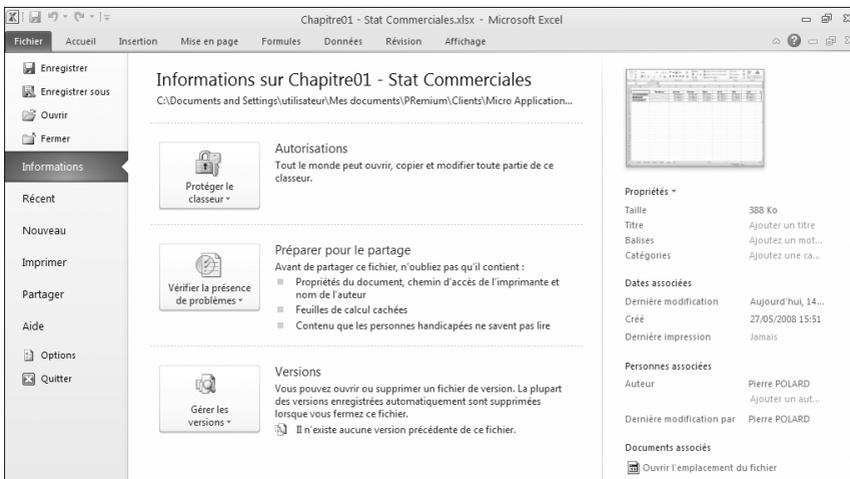


Figure 1.4 : Le mode Backstage

La personnalisation du Ruban est facilitée. Désormais, il est possible de créer des onglets personnalisés et d'y ajouter des commandes alors qu'auparavant, cette opération nécessitait de programmer les modifications désirées en langage XML.

La fonction Copier-coller s'enrichit d'un aperçu des données qui permet de visualiser le résultat des différentes options de collage avant de procéder à l'opération de collage proprement dite.

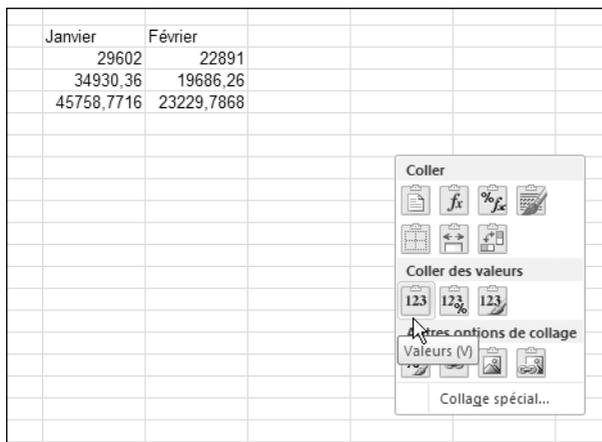


Figure 1.5 : L'aperçu des données de la fonction Copier-coller

Excel 2010 présente bien d'autres améliorations et fonctions nouvelles :

- L'exploitation des tableaux Excel (qui remplacent les listes Excel depuis Excel 2007) est facilitée. En cas de défilement vers le bas, les entêtes du tableau remplacent les entêtes de colonnes de feuille de calcul. Les boutons de filtre automatique restent donc toujours visibles, ce qui permet de trier et de filtrer les données sans avoir à revenir au début du tableau.

	Région	Ville	Famille	Produit	Exercice	Trimestre	Mois	CA	Unité	
217	EST	STRASBOURG	RANGEMENT	TIROIR	2009		2	6	4 261,05	637,21
218	EST	STRASBOURG	ACCESSOIRES	FAUTEUIL	2009		3	7	4 267,73	349,80
219	EST	NANCY	ACCESSOIRES	CHAISE	2008		1	1	4 288,90	440,03
220	EST	NANCY	MOBILIER	ARMOIRE	2009		3	7	4 300,04	738,58
221	EST	STRASBOURG	RANGEMENT	CAISSON	2009		2	6	4 307,84	568,14
222	EST	STRASBOURG	ACCESSOIRES	CHAISE	2008		4	12	4 310,07	671,74
223	EST	STRASBOURG	RANGEMENT	CAISSON	2008		1	2	4 337,92	577,05
224	EST	NANCY	ACCESSOIRES	LUMINAIRE	2009		2	6	4 339,03	778,69
225	EST	STRASBOURG	ACCESSOIRES	DECORATION	2009		1	3	4 342,37	252,88
226	EST	NANCY	RANGEMENT	CAISSON	2009		1	1	4 342,37	378,76
227	EST	NANCY	MOBILIER	TABLE SALON	2008		2	5	4 343,49	387,67
228	EST	NANCY	RANGEMENT	ETAGERES	2008		1	2	4 352,40	412,18
229	EST	STRASBOURG	MOBILIER	ARMOIRE	2008		1	1	4 353,51	346,45

Figure 1.6 : Les en-têtes du tableau toujours visibles

- Les possibilités de mise en forme conditionnelle ont été enrichies avec de nouveaux jeux d'icônes, plus complets. Les barres de données ont également été améliorées pour être plus représentatives des données sous-jacentes. Lorsque vous spécifiez des critères pour les règles conditionnelles, vous pouvez désormais faire référence à des valeurs d'autres feuilles de calcul du classeur.
- Afin de faciliter l'exploitation des tableaux croisés dynamiques, les *slicers* ont été introduits. Ils permettent de filtrer les données de façon très intuitive et très visuelle. Une fois que vous avez inséré un slicer, vous utilisez des boutons pour segmenter et filtrer rapidement les données afin d'afficher uniquement les informations requises.



Figure 1.7 : Les slicers permettent de mieux exploiter les données

- Une nouveauté parmi les graphiques : les graphiques sparklines. Il s'agit de petits graphiques insérés dans une cellule qui permettent de visualiser des tendances en un coup d'œil.

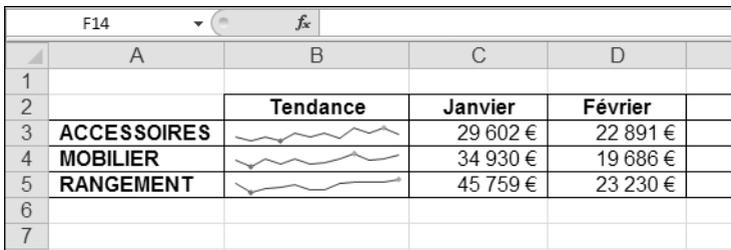


Figure 1.8 : Les graphiques sparklines donnent une vision des tendances

- Excel 2010 comprend une version améliorée du Solveur, que ce soit au niveau de l'ergonomie ou de la précision des résultats.
- Le complément Power Pivot (qui nécessite un téléchargement gratuit) permet d'accéder à des bases de données, de faire des requêtes sur ces données d'origine différente et d'exploiter le résultat dans Excel. Ce complément représente un pas de plus d'Excel en direction du monde de l'informatique décisionnelle d'entreprise.
- L'application Web Excel, compagnon en ligne d'Excel, permet d'accéder aux classeurs depuis un navigateur Internet. Dans le navigateur, il est par exemple possible de trier ou filtrer les données des feuilles de calcul Excel.

1.2. Découvrir l'évolution des macros et de la programmation

Après avoir passé en revue l'évolution des fonctions "classiques" des tableurs et surtout d'Excel, intéressons-nous à présent à l'envers du décor, c'est-à-dire aux macros et à la programmation.

Lotus 1-2-3 est le premier tableur à proposer à ses utilisateurs la possibilité de réaliser des macros. Il s'agissait d'un langage rudimentaire, fondé sur les séquences de touches correspondant aux opérations réalisées manuellement. Les macros étaient saisies directement dans les feuilles de calcul, ce qui les rendait vulnérables aux erreurs de manipulations. Pourtant, les utilisateurs ont saisi cette occasion (inespérée) de s'émanciper de leurs services informatiques et d'acquérir une certaine autonomie pour automatiser leurs traitements. Ils devaient déployer des trésors d'astuces pour contourner les limitations du langage archaïque qui leur était proposé.

Les premières macros pour Excel étaient saisies dans des feuilles de macros, sauvegardées dans des fichiers *.xlm*. Elles ont fait leur apparition dans la version Excel 4, qui a entériné la suprématie d'Excel sur le monde des tableurs. Le langage XLM était certes rudimentaire, mais il s'agissait d'un langage à part entière. Il permettait de réaliser des tests, des boucles...

C'est avec la version 5 d'Excel qu'apparaît un changement fondamental : l'introduction de VBA (*Visual Basic pour Application*). D'abord réservé à Excel, ce langage a gagné progressivement l'ensemble des applications Office. VBA est un langage orienté objets, c'est-à-dire qu'il autorise la manipulation d'objets. Chaque application possède

des objets qui lui sont propres (le modèle d'objet) ; en revanche, la structure du VBA est la même pour l'ensemble des applications Office. La compatibilité avec le langage XLM est toujours assurée.

Excel 97 introduit des changements majeurs dans la gestion des macros VBA. En effet, jusqu'alors les macros étaient stockées dans des modules qui étaient des feuilles du classeur Excel. Désormais, les modules sont stockés dans un projet VBA associé au classeur et visible dans un environnement spécifique : l'éditeur Visual Basic ou VBE. Les boîtes de dialogue personnalisées (*UserForm*) font également leur apparition en tant qu'objets du projet VBA associé au classeur. Le nombre d'événements gérés par chaque type d'objet augmente de façon importante. Les contrôles ActiveX font également leur apparition.

L'ergonomie du VBE est également largement améliorée, avec une aide en ligne revue et un Explorateur d'objets plus performant.

Excel 2000 n'a pas apporté de changements majeurs, si ce n'est l'apparition des boîtes de dialogue non modales (permettant la création d'écran d'accueil pour les applications).

Les versions 2002 et 2003 ont introduit uniquement des changements marginaux.

Excel 2007 a amené des changements importants dans l'interface utilisateur avec l'apparition du Ruban. En revanche, l'environnement de développement VBE conserve "l'ancienne" apparence avec menus et barres d'outils. Pourtant, le Ruban aura des répercussions dans la programmation : il faudra utiliser de nouvelles techniques pour le programmer avec VBA.

Les nouvelles fonctionnalités d'Excel se traduisent par l'apparition de nouveaux objets associés aux formats conditionnels, de nouvelles méthodes pour l'objet `WorkBook`, de nouvelles propriétés associées à l'objet `Application`.

L'aide en ligne de l'environnement de développement VBE a également été améliorée et enrichie.

VBA 7, la version de VBA associée à Excel 2010, permet le développement de code pouvant être exécuté indifféremment avec les versions 32 bits et 64 bits d'Office.

Avec Excel 2010, il est désormais possible d'utiliser l'enregistreur de macros pour enregistrer les modifications apportées lors de la mise en forme de graphiques et des autres objets. Pour ce faire, de

nouvelles propriétés ont été ajoutées aux objets graphiques du modèle d'objets d'Excel.

La propriété *PageSetup* des feuilles de calcul a été améliorée pour prendre en compte toutes les fonctionnalités de mise en page auparavant supportées par la fonction `PAGESETUP()` du langage XLM (le premier langage de programmation associé à Excel).

1.3. Respecter quelques principes de développement

Que ce soit en VBA ou dans n'importe quel autre langage, développer un programme requiert un peu de rigueur et le respect de quelques étapes importantes. Nous allons nous placer dans l'optique d'un développeur qui travaille pour d'autres utilisateurs, mais la problématique reste sensiblement la même si l'on travaille pour soi.

Voici, à titre indicatif, les étapes à respecter pour le développement d'une application Excel avec VBA :

- définir clairement le besoin des utilisateurs ;
- préciser les données requises, ainsi que leur emplacement ;
- construire les feuilles de calcul et développer les macros VBA ;
- définir l'interface utilisateur ;
- tester l'application ;
- documenter l'application ;
- installer l'application et former les utilisateurs.

Avant de détailler les différentes étapes mentionnées précédemment, il convient de combattre une idée reçue : les applications développées avec VBA et Excel ont parfois la réputation d'être des "usines à gaz" peu fiables. Cela n'est, bien sûr, pas dû à VBA, mais au non-respect des règles élémentaires de développement. En effet, Excel et VBA sont disponibles sur la plupart des ordinateurs si bien que certains utilisateurs s'improvisent développeurs d'applications avec plus ou moins de succès. La fiabilité d'une application repose, dans une large mesure, sur la rigueur de son développeur. Il ne s'agit pas d'être un informaticien chevronné mais de respecter, comme pour toute activité, un ensemble de bonnes pratiques. Celles que nous allons développer à présent n'ont pas la prétention d'être universelles mais constituent une base de travail.

Définir clairement le besoin des utilisateurs

En règle générale, on ne développe pas une application pour le plaisir, mais pour répondre aux besoins d'utilisateurs. Il s'agit souvent d'automatiser des tâches répétitives, de rendre plus rapide la synthèse de données, de faciliter la saisie et la restitution d'informations. Chaque problématique est spécifique mais il convient de la définir le plus précisément possible. Il s'agit de définir un cahier des charges, on parle aussi de "spécifications".

Une chose est sûre : une application développée à partir de spécifications imprécises donnera forcément des résultats insatisfaisants.

Avant de se lancer tête baissée dans le développement du code, il est donc impératif de prendre un minimum de recul, de dialoguer avec les utilisateurs pour les aider à formaliser leur besoin. La première étape consiste à recenser tous les utilisateurs : ceux qui vont utiliser l'application au quotidien, ceux qui seront destinataires des résultats... Excel étant un outil largement employé, de nombreux utilisateurs ont un point de vue sur la réalisation pratique de l'application ("Il faudrait un fichier comme ça, une formule qui donne ce résultat..."). Le développeur doit savoir faire abstraction de ces préconisations parfois hasardeuses pour se concentrer sur le réel besoin, qui peut parfois être implicite. Le temps passé à dialoguer avec les utilisateurs, loin d'être perdu, constituera par la suite un gain de temps considérable dans la mesure où il évitera de partir dans une mauvaise direction.

Cette phase d'écoute terminée, le développeur pourra utilement demander un *feed-back* aux utilisateurs pour s'assurer que ce qu'il a compris correspond à leurs attentes. Par expérience, on peut toutefois affirmer que les utilisateurs ne "croient que ce qu'ils voient" et que leur avis définitif sera prononcé uniquement lorsqu'ils recevront la première version de l'application. Pour autant, cette étape préparatoire évite les plus grosses déconvenues.

Une fois que tout le monde est d'accord sur les tenants et aboutissants de l'application, le développement peut commencer.

Préciser les données requises ainsi que leur emplacement

L'étape précédente aura permis de spécifier les modalités de traitement et les résultats attendus. Il faut maintenant rentrer dans le détail

des données nécessaires au bon fonctionnement de l'application, en réaliser une sorte de cartographie :

- Quelle est leur nature ?
- Seront-elles saisies directement dans l'application ?
- Devront-elles être stockées dans l'application ou dans une base données annexe ?
- Proviendront-elles d'autres applications ?
- Comment se fera la liaison avec les applications qui fournissent les données (liens OLE ou ODBC, fichiers textes à importer...) ?

Les réponses à ces questions auront un lourd impact sur la structure de la future application, ainsi que la complexité de son développement.

Construire les feuilles de calcul et développer les macros VBA

Une fois le travail préparatoire terminé, il s'agit de bâtir la structure de l'application. La première question à se poser est celle du choix de la version d'Excel. Ce choix est d'autant plus crucial si vous souhaitez développer avec les nouveaux formats de fichiers proposés par Excel 2007 et 2010. Il faudra vous assurer que les utilisateurs pourront les lire. Plus généralement, cette question demeure pertinente quelle que soit la version d'Excel utilisée.

Une fois cette question primordiale réglée, il s'agit de construire les feuilles de calcul nécessaires au stockage et à la présentation des données. Ces feuilles de calcul constitueront l'ossature, la charpente de votre application. Les macros VBA représenteront quant à elle les "muscles" qui la mettront en mouvement. C'est la bonne imbrication de ces deux entités qui assurera le bon fonctionnement de votre application.

Cette étape est fondamentale et constitue le cœur du processus de développement. Il est important de respecter quelques principes simples :

- modularité ;
- recyclage ;
- simplicité ;
- fiabilité ;
- relecture.