

# Alternative à MsgBox()

## Version revisitée

Dans cette nouvelle version, les messages ne sont plus intégrés à la fonction Dialogue, mais stockés dans un formulaire dédié. La gestion des messages (*création, modification, suppression*) se fait via une application spécifique qui rend le travail de programmation beaucoup plus simple et surtout plus convivial.

Une fois mis au point, le jeu de messages sera directement exporté dans l'application destination. La fonction Dialogue (*nouvelle version*) se chargera d'extraire les données du message (*depuis le formulaire dédié*) et l'affichera selon les paramètres définis (*via le formulaire Message*).

Par rapport à l'ancienne version, la taille du projet se trouve fortement diminuée.

### Contexte d'utilisation : (*Rappels et nouveautés*)

La fonction Dialogue() est une alternative à MsgBox(). Elle offre plus de possibilités en matière d'affichage et de paramétrage des messages. Comme par exemple afficher des messages "Furtifs", que s'affichent et disparaissent au bout d'un certain temps, du texte long sur une seule ligne, de la couleur dans le message, des caractères de tailles différentes, des fenêtres plus grandes, des icônes personnalisées, etc.

Elle permet au développeur de déterminer (*via l'application de gestion*) ses propres choix pour l'affichage des messages.

La fonction se charge, à partir des données du message (*et de ses paramètres*), de préparer le formulaire "Message" (*hauteur/largeur, etc.*), de placer le texte (*les labels*), de faire apparaître l'image associée et de positionner les boutons. Après validation par l'utilisateur final (*si validation*), elle retourne une valeur qui sera utilisée pour la suite du traitement.

### Exemples d'appels de la fonction :

M1 = Dialogue (N° Message)

If Dialogue (N° message) = 1 Then ...

Select case Dialogue (N° message) ...

Etc.

**M1** est une variable publique ; bien que réservée à la fonction, elle pourra servir à recevoir la réponse.

**N° message** est le numéro du message à afficher. La fonction extraira les données directement depuis le formulaire "Données\_Msg ».

Dans cette nouvelle version, le type de validation disparaît ; il est déterminé par la fonction (*selon le nombre de boutons positionnés*).

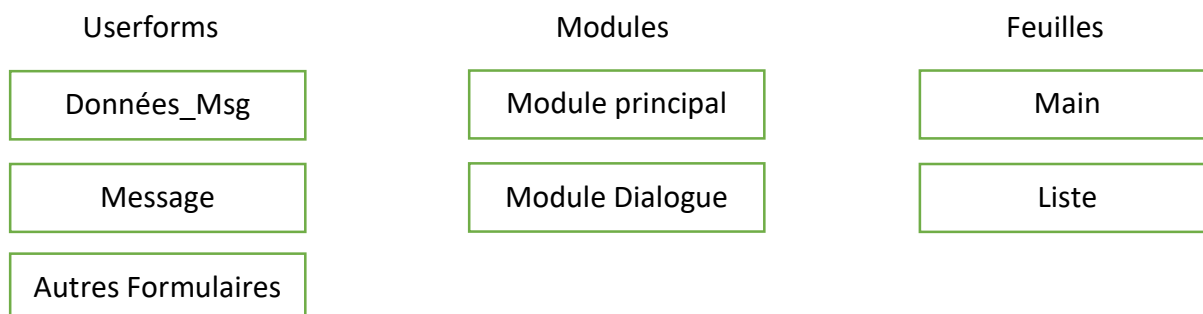
### Valeurs retournées par la fonction :

- 0 pour un message "Furtif" (*sans bouton*), utilisation de la touche Echap ou fermeture du formulaire par la croix,
- 1, 2 ou 3 selon le N° du bouton sélectionné,
- 99 si le message offre une possibilité d'aide et que l'utilisateur clique sur l'image (*point d'interrogation*) ou sur le label "Aide" présent dans ce cas.

## Le classeur de Gestion

Le classeur "Gestion des messages.xlsm" (*fourni en pièce jointe*) constitue l'axe principal pour la gestion des messages.

### Structure du classeur



Le module principal contient les différentes procédures liées à la gestion de l'ensemble.

Le module Dialogue contient la fonction ainsi que les déclarations nécessaires à son fonctionnement. Ce module est autonome ; il a été conçu pour fonctionner dans tous les projets. Il sera recopié en l'état.

Au moment de l'implantation, on s'assurera que les déclarations ne sont pas redondantes avec celles définies dans le projet.

Un secteur de la fonction (*avant l'affichage final*) est réservé pour la programmation des spécificités (*Traitements spécifiques ne pouvant pas être définis dans l'appli de gestion*). Ce codage pourra se faire dans le module source (*classeur de gestion*) ; le module sera ensuite recopié dans tous les projets (*voir le chapitre "Spécificités" pour les détails*).

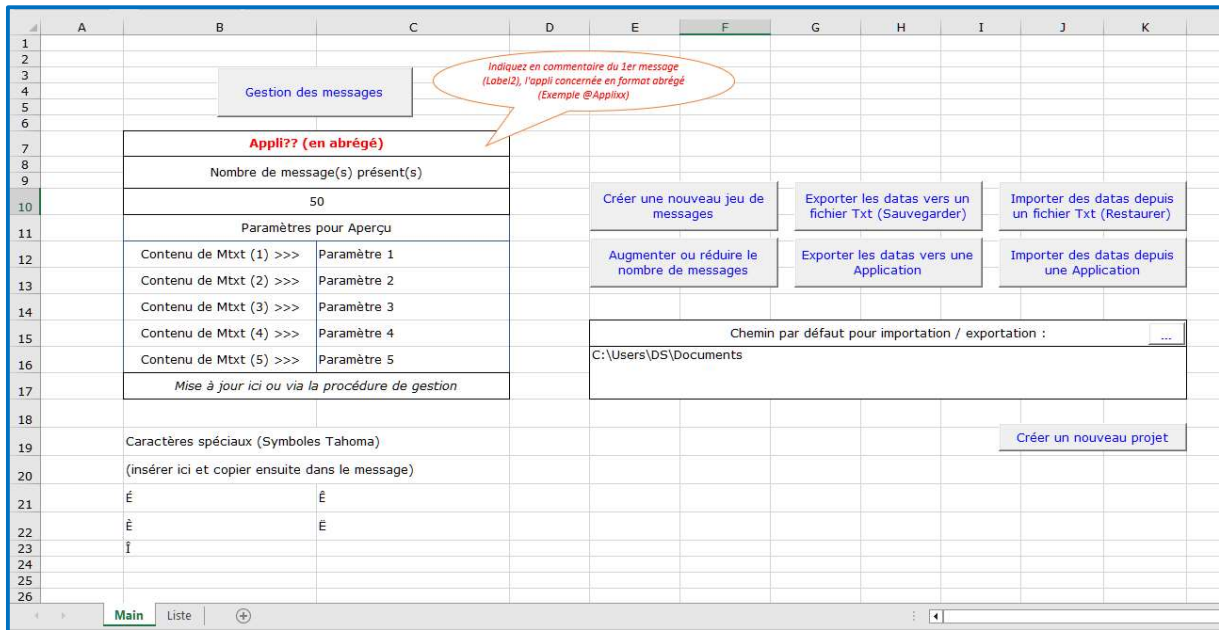
Le formulaire Données Msg sert au stockage des messages ; ceux-ci sont enregistrés au format texte sous une forme codée afin d'occuper le moins de place possible.

Le formulaire Message sera utilisé, par la fonction, pour l'affichage du message. Il reprend les mêmes principes que dans l'ancienne version avec quelques améliorations (*Petites et grandes images, un label supplémentaire pour le cadrage des sommes, etc.*).

Ces deux formulaires seront à recopier dans les différents projets utilisant la fonction. Concernant le formulaire "Données\_Msg", si celui-ci contient déjà un jeu de messages, il pourra être copié en l'état. La mise à jour pourra se faire à posteriori, via l'application de gestion (*exportation/changement du jeu*).

Les autres formulaires sont propres à l'application de gestion et n'ont pas à être recopiés.

## Fenêtre principale



A l'ouverture du classeur, la feuille "Main" est affichée. Les différents boutons permettent :

- De traiter le jeu de messages actuellement en mémoire,
- De créer un nouveau jeu,
- De créer un classeur de projet,
- De sélectionner le chemin par défaut.

Dans cette feuille, on trouve également :

- L'identification du jeu de messages, permettant de différencier les différents jeux,
- Le nombre de messages présents dans le jeu en mémoire,
- Les paramètres provisoires utilisés pour les aperçus,
- Le chemin par défaut pour l'import et l'export (*si vide, c'est le chemin du classeur qui sera utilisé*),
- Une zone libre permettant de copier/coller des caractères spéciaux.

La seconde feuille de calcul "Liste" permet de visualiser l'intégralité des messages présents dans le jeu avec possibilité d'afficher les caractères masqués et de sélectionner un message en particulier pour modification (*bouton "Aller au message" ou double-clic sur un numéro*). On pourra également y faire des recherches pour cibler une donnée précise.

A la fermeture du classeur (*avec enregistrement*), le jeu de messages en mémoire est sauvegardé.

### Définition et usage des boutons :

#### Gestion des messages

Pour la gestion des messages (*création, modification, aperçu, etc.*). Voir le chapitre spécifique "La gestion des messages".

#### Créer un nouveau jeu de messages

**Le nouveau jeu remplacera le jeu actuellement en mémoire.**

Par défaut, le nombre de messages proposé est de 50. Ce nombre est limité à 999.

Les nouveaux messages sont créés avec des valeurs par défaut.

A savoir :

- Titre (*Caption*) : = "Message Libre". Cette donnée permettra de repérer les messages disponibles.
- Label1 & 2 : Texte = vide, centré, noir, de taille 12 pts.  
(Pour le message n°1, un commentaire sera placé dans le Label2 permettant d'identifier le jeu).
- Boutons : seul le bouton 1 est paramétré (= "Vu").  
Hauteur = 25, Largeur = 50, Taille des caractères = 12, Couleur du texte = Bleu, Couleur du fond = Couleur par défaut (*gris*).
- Message (*Formulaire*) : couleur par défaut (*gris, valable également pour le fond des labels*). La police de caractères est celle par défaut (*Tahoma*). Elle n'est pas modifiable ici ; cependant, si souhaité, on pourra intervenir sur les propriétés des contrôles pour la modifier ou utiliser une spécificité.
- Temporisation : =0. Utilisée uniquement pour les messages "Furtifs" (*sans bouton*), la valeur sera exprimée en millisecondes (*valeur usuelle = 1800*).
- Bip Sonore : coché
- Avec Aide : décoché
- Images : sens interdit de petite taille

Les valeurs par défaut sont définies dans la procédure "Création\_Msg" de l'application et peuvent être adaptées.

#### Exporter les datas vers un fichier Txt (*Sauvegarder*)

Sauvegarde le jeu de messages dans un fichier texte. Un panneau précise les modalités d'enregistrement. Le nom du jeu (*1<sup>er</sup> commentaire du message n°1 dans le Label2*) est proposé comme nom de fichier (*ne pas ajouter d'extension*).

Le fichier sera enregistré à l'emplacement indiqué en cellule E16 de la feuille "Main" (*ou dans le dossier du classeur de gestion si vide*).

#### Importer les datas depuis un fichier Txt (*Restaurer*)

Récupère un jeu de messages préalablement sauvegardé.

Après confirmation, les messages sont importés et **remplacent** le jeu en mémoire.

Les fonctions Exporter/Importer vers/depuis un fichier texte, permettent de garder et de récupérer une copie valide des jeux de messages.

#### Augmenter ou réduire le nombre de messages

Selon le nouveau nombre indiqué, le jeu de messages sera réduit ou augmenté.

Réduction : les messages excédentaires sont supprimés de la mémoire.

Augmentation : les nouveaux messages sont initialisés avec les valeurs par défaut.

Une fois la programmation du projet terminé, on ajustera le nombre de messages en supprimant les messages "Libres" placés en fin de jeu. On limitera ainsi la taille de l'application.

#### Exporter les datas vers une application

Exporte le jeu de messages vers une application. Les classeurs déjà ouverts (*et répondant aux conditions d'export*) sont proposés dans une liste de choix. En l'absence de liste, le classeur sera sélectionné dans l'explorateur puis ouvert. Il devra répondre aux conditions d'exportation qui sont : posséder un formulaire "Données\_Msg" et ne pas être protégé.

Après confirmation, le jeu de messages est copié dans l'application. Le classeur reste ouvert pour les tests et les vérifications.

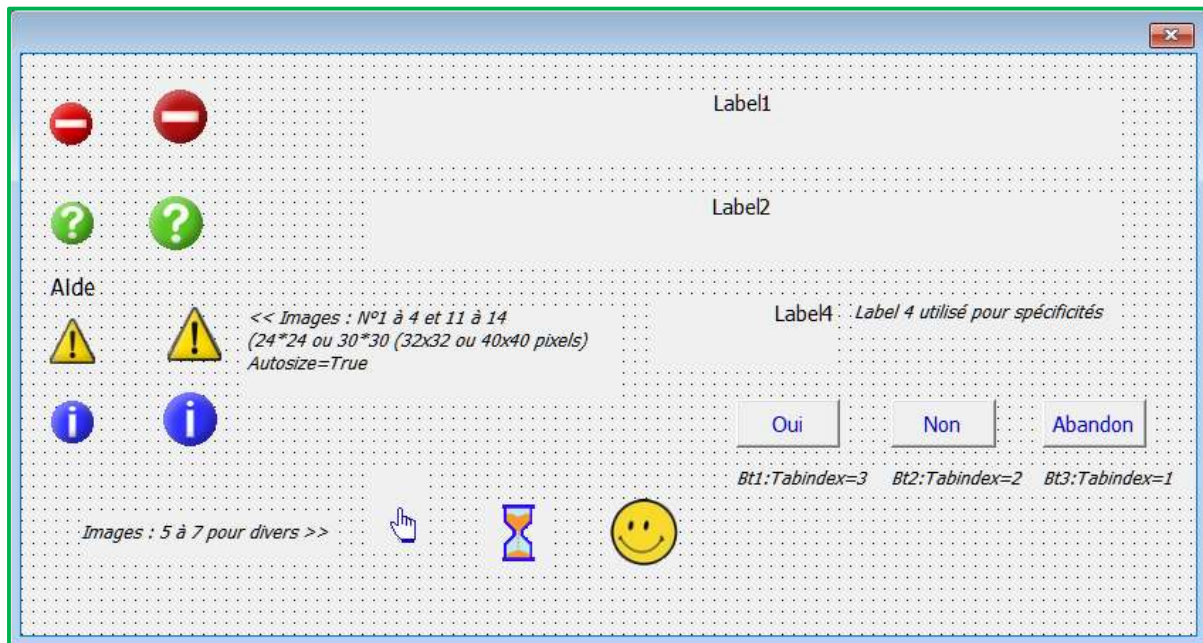


Un seul contrôle est présent dans ce formulaire. Le Textbox nommé "Datas" recevra les données du jeu de message. Ici le jeu, composé de 50 messages, est pour l'instant paramétré à des valeurs par défaut.

Côté programmation, seul l'évènement "Activate" est codé. Le formulaire est refermé aussitôt après ouverture. Phase obligatoire pour pouvoir enregistrer "en Dur" les données dans le contrôle.

Ce formulaire sera recopié en l'état dans les différents projets.

### Le Formulaire Message



On y trouve tous les contrôles utilisés pour l'affichage. Les propriétés de chaque contrôle ont été fixées par défaut. La taille et le positionnement n'ont ici aucune importance (*y compris pour le formulaire*) puisque c'est la fonction Dialogue qui se charge de cette tâche.

#### Les images :

4 images de base, en 2 tailles, seront proposées pour les messages (*images 1 à 4 & 11 à 14*). Selon le type de message, on sélectionnera une grande ou petite image. Par exemple, pour les messages de plusieurs lignes, on choisira une grande image (*pour le visuel*).

La "Main" (*image5*) sera utilisée pour les messages offrant une possibilité d'aide (*Propriété MouseIcon*).

Le sablier (*image6*) est utilisé par l'application (*Panneau "Traitement en cours"*).

La dernière image (*image7*) sera utilisée pour un exemple de message avec spécificité (*voir le chapitre "Spécificités" pour les détails*).

#### Les labels :

Les labels 1 & 2 serviront au texte du message. Ici on utilise 2 labels pour pouvoir varier la présentation (*par exemple le Label1 en noir, le Label2 en rouge et aligné à gauche*). Lors du paramétrage, on pourra se servir uniquement du Label1, en incluant au texte des retours chariot pour une présentation sur plusieurs lignes.

Le label 3 "Aide" sera utilisé pour les messages offrant une possibilité d'aide. Dans ce cas le label sera positionné au-dessous de l'image qui sera dans ce cas le point d'interrogation.

Enfin le label 4, sera réservé aux spécificités. Il permettra notamment de cadrer correctement des valeurs numériques dans le message (*voir le chapitre "Spécificités" pour les détails*).

### Les boutons :

Comme indiqué précédemment, ils sont au nombre de trois. Le texte et la couleur placés ici n'ont aucune importance. Ils seront paramétrés via l'application de gestion.

Ce formulaire sera recopié en l'état dans les différents projets. L'ajout d'images supplémentaires, ou le remplacement des images proposées, reste possible. La taille et la numérotation devront être respectées.

### La Fonction Dialogue

Elle est intégrée à un module spécifique. Ce qui permet une recopie plus facile dans les projets.

Les différentes phases de la fonction : *(voir le code pour les détails)*

#### Entrée :

Après définition des variables locales, le classeur actif est testé. Le traitement sera différent selon que l'on est dans l'application de gestion ou dans une autre application.

#### Extraction des données du message :

Pour l'application de gestion (*messages internes, aperçu, diaporama*) les données messages sont déjà en place (*Tabl\_Msg()* & *Nbre\_Msg*).

Pour les autres applications, les données du message (*paramètre d'appel Msg*) sont extraites depuis le formulaire "Données\_Msg".

Un test est ensuite réalisé sur la validité du n° de message et des données extraites. En cas d'anomalie, un rejet est affiché et l'application est stoppée.

#### Mise en forme du formulaire Message :

- Détermination du nombre de boutons,
- Préparation du formulaire (*labels, boutons, images, etc.*),
- Mise en place des paramètres optionnels :
  - Cette phase consiste à remplacer les termes "Mtxt(?)" présents dans les données messages par le contenu de la variable publique tableau Mtxt(?). Sont concernés, les champs Titre, Label 1 & 2, le texte et les infos bulles des boutons (*voir chapitre "Gestion des messages" pour les détails*).
- Suppression des commentaires présents dans le Label2 (*si existent*),
- Dimensionnement et placement des différents contrôles,
- Pour les messages avec possibilité d'aide (*paramètre "Aide" coché*) :
  - Placement du label "Aide" sous l'image point d'interrogation, ajout de l'icône "Main" et de l'info-bulle pour la sélection.
- Emission d'un bip sonore (*si paramétré*),
- Traitement des spécificités :
  - Ici on vérifie si le Label2 (*seul concerné*) comporte une spécificité. Les spécificités permettent un traitement particulier des messages ne pouvant être paramétrés via l'application de gestion. Elles sont signalées par la présence du terme "@SP=xx" dans le texte. "xx" correspondant au numéro de spécificité et donc à la partie de code à exécuter (*Voir le chapitre "Spécificités" pour la mise en œuvre*).

#### Affichage du formulaire Message :

Absence de bouton : le formulaire est affiché pendant un certain temps (*paramètre Temporisation*) puis déplacé vers le haut avant effacement.

Présence de bouton(s) : l'utilisateur effectue son choix et le formulaire est effacé (*idem sans bouton*).

## Retour à la procédure appelante :

Selon le type de message et l'action de l'utilisateur, la réponse est retournée à la procédure appelante.

= 0 pour un message "Furtif", utilisation de la touche Echap ou fermeture par la croix,

= 1, 2 ou 3 pour sélection du bouton 1, 2 ou 3

= 99 si message avec possibilité d'aide et sélection de l'image ou du label "Aide"

## La gestion des messages

### Formulaire de gestion et fonctionnalités

The screenshot shows a software window titled "MISE À JOUR DES DONNÉES MESSAGES. Nbre de messages présents : 50". The main content area is titled "Message N° 001" and contains several input fields and controls:

- Caption:** A text box containing "Message Libre".
- Label 1:** A large empty text area.
- Label 2:** A text box containing "@Appli?? (en abrégé)".
- Bouton 1:** A text box containing "Vu".
- Bouton 2:** An empty text box.
- Bouton 3:** An empty text box.
- Info Bulle:** Three yellow rectangular buttons corresponding to Bouton 1, 2, and 3.
- Message:** A section with fields for "Couleur" (a color picker), "Temporisation" (set to 0), "Bip Sonore" (checked), and "Avec Aide" (unchecked).
- Dimensions and Style:** Fields for "Hauteur" (25,00), "Largeur" (50,00), "Taille des car." (12,00), and "Couleur" (A).
- Alignement:** Radio buttons for "Gauche", "Centré" (selected), and "Droite".
- Texte:** Fields for "Couleur" (black), "Taille" (12,00), and checkboxes for "Italique" and "Gras".
- Images:** A section with radio buttons for "Image 40x40 pixels" and "Pas d'image", and icons for a red circle with a slash, a yellow triangle with an exclamation mark, a green question mark, and a blue circle with an 'i'.

At the bottom, there are several buttons: "Paramètres", "Aperçu", "Annuler les modifications", "Diaporama", "Réinitialiser le message", and "Quitter".

A l'entrée dans la procédure de gestion, le 1<sup>er</sup> message du jeu est affiché.

Chaque contrôle, présent dans le formulaire, dispose d'une info-bulle indiquant son rôle ou ses limites.

Ici le message est disponible (*Caption* = "Message Libre") ; les données sont celles définies par défaut.

### Modification d'un message

Cette phase consiste à modifier les valeurs par défaut (*message Libre*) pour créer un message définitif ou rectifier un message déjà paramétré.

Complétez chaque zone du formulaire et sélectionnez les différentes options proposées.

Cliquez sur les boutons de couleur pour modifier la couleur des données.

Le champ "Caption" sera utilisé pour le titre du formulaire message.

Dans les champs "Label 1" & "Label 2" on indiquera le texte du message. Utilisez la touche "Enter" pour placer des retours de lignes et la touche "Tab" pour décaler le texte vers la droite. Le champ "Label 2" est facultatif ; selon le message, le Label 1 pourra être utilisé seul.

Les champs "Bouton 1" à "Bouton 3" et "Info-bulle" seront utilisés pour définir les boutons. Les champs hauteur, largeur et taille des caractères ainsi que les boutons "Couleur" définiront les propriétés des boutons (*variables pour les 3 boutons*).



## Importance du paramétrage des boutons & placement :

Le placement des boutons s'effectuant dans l'ordre, on s'attachera à définir le premier comme bouton de confirmation et le dernier comme bouton d'abandon ou d'annulation. Le focus étant toujours attribué au dernier bouton (*le plus à droite lors de l'affichage*), on évitera ainsi une validation par inadvertance (*touche Enter par exemple*), et cela simplifiera la programmation des réponses.

Ils seront automatiquement centrés et placés en bas du formulaire avec un espace entre chaque.

- 1 seul bouton (*CommandButton1*). En général "Vu" ou "Ok". Il sera au centre.
- 2 boutons (*CommandButton1 & 2*). Par exemple "Oui" / "Non", le bouton 2 sera placé à droite du bouton 1 et aura le Focus.
- 3 boutons (*Comm..1 2 3*). Par exemple "Accepter" / "Refuser" / "Annuler", le 3<sup>ème</sup> bouton sera placé à droite du bouton 2 et aura le Focus.

Comme dans la version précédente, le nombre de boutons est limité à 3. Ils doivent être paramétrés dans l'ordre. Pas de bouton 2 sans bouton 1 ni de bouton 3 sans bouton 2.

La présence d'un bouton, dans le message, est déterminée par son texte (*caption*). En l'absence de texte, le bouton n'est pas affiché et en l'absence de bouton, le message est considéré comme "Furtif" ; il disparaîtra après un certain temps (*valeur de temporisation*).

Le champ "Temporisation" (*temps d'attente pour les messages "Furtifs"*) restera à 0 dès lors qu'au moins 1 bouton sera positionné (*texte <> vide*).

L'activation de la case à cocher "Avec Aide" sélectionnera automatiquement l'image point d'interrogation. Avec cette option, le message offrira un choix supplémentaire permettant de cliquer sur l'image ou le label "Aide". Si sélectionné, la fonction retournera la valeur 99 indiquant une demande d'aide (*à coder dans le projet*). La sélection d'une autre image (*ou aucune image*) désactivera l'option.

Le choix de l'image dépendra du type de message : sens interdit pour une interdiction, point d'interrogation pour une question (*ou aide*), point d'exclamation pour une anomalie, icône Info pour une information ou aucune image si non utile. La taille par défaut sera de 32x32 pixels, ou 40x40 pixels si la case "Image 40x40 pixels" est cochée. Selon le message on sélectionnera l'une des 2 tailles (*pour le visuel*).

## **Boutons de commandes en bas du formulaire :**

### Paramètres

Permet de modifier les paramètres provisoires d'aperçu (*voir chapitre "Ajout de paramètres"*).

### Aperçu

Utilisez ce bouton pour visualiser le résultat du paramétrage. Le numéro du message s'affiche en barre de tâche suivie, après effacement, de la réponse issue de la fonction dialogue. Cette information permet de s'assurer du retour attendu pour le codage définitif dans le projet.

### Annuler les modifications

Restaure (*après confirmation*) le message à son état précédant l'entrée dans la procédure de gestion.

### Diaporama

Permet d'enchaîner un aperçu de chaque message présent dans le jeu. Il débute par le message actif et se termine par le dernier. Le message de lancement précise les modalités d'arrêt et permet d'exclure les messages inutilisés (*Caption="Message Libre"*).  
(*Important : les messages avec paramètres utilisent les paramètres provisoires*)

## Réinitialiser le message

Après confirmation, les valeurs par défaut sont appliquées au message ; le message redevient disponible.

## Quitter (ou fermeture du formulaire par la croix)

Ferme la fenêtre de gestion. Si des modifications ont été apportées au jeu de messages, une confirmation sera demandée pour la mise à jour.

Réponse "Oui" : le jeu de messages est enregistré dans le formulaire Données\_Msg.

Réponse "Non" : **toutes les mises à jour**, effectuées depuis l'entrée dans la procédure de gestion, sont annulées (*une deuxième confirmation sera demandée*).

Les modifications ne seront définitives qu'une fois le classeur de gestion enregistré. Il est possible d'annuler toutes modifications en refermant le classeur sans l'enregistrer.

## Ajout de paramètres

Les paramètres permettent d'insérer dans le message des informations issues de l'application hôte. Dans cette nouvelle version, ils deviennent pratiquement obligatoires puisque les messages sont construits hors projet et que les informations sont, à priori, inconnues à ce stade.

L'ajout de paramètres permet également de limiter le nombre de messages. Un même message pourra être utilisé plusieurs fois avec des paramètres différents.

L'insertion se fait en ajoutant au texte le nom du paramètre sous la forme Mtxt(élément).

Mtxt() est le nom donné à la variable tableau de 5 éléments (*texte*) déclarée dans la fonction Dialogue (*Public Mtxt(5) As String*).

Tous les champs avec texte peuvent contenir des paramètres (*y compris info-bulles des boutons*) et le nombre n'est pas limité. Lors de l'affichage du message, la chaîne "Mtxt(?)" sera remplacée par le contenu de la variable tableau Mtxt(?).

Lors de l'aperçu (*ou Diaporama*), l'application de gestion utilise les paramètres provisoires définis dans la feuille "Main". Le bouton "Paramètres" permet de les modifier sans quitter le formulaire de mise à jour. On adaptera les valeurs en fonction du résultat attendu (*texte, chiffres, etc.*).

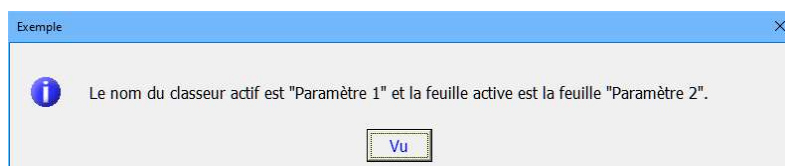
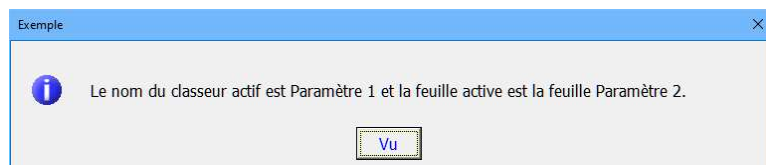
Exemple d'insertion dans le Label 1 :

Le nom du classeur actif est Mtxt(1) et la feuille active est la feuille Mtxt(2).

*Ou entre guillemets*

Le nom du classeur actif est "Mtxt(1)" et la feuille active est la feuille "Mtxt(2)".

Aperçus côté application de gestion : (*paramètres par défaut*)



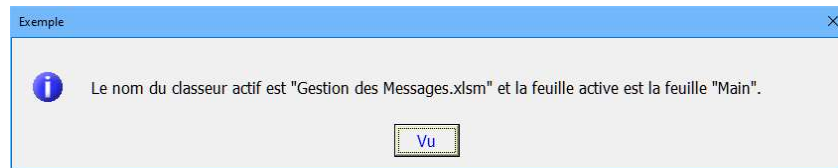
Codification de l'appel dans l'application hôte (N° de message =9 pour l'exemple & guillemets) :

*Mtxt(1) = ActiveWorkbook.Name*

*Mtxt(2) = ActiveSheet.Name*

*M1 = Dialogue(9)*

*Donnera :*



Dans le projet, les paramètres doivent être prêts à l'emploi, c'est-à-dire contenir toutes les informations utiles (*texte, ponctuation éventuelle, etc.*) et être formatées s'il s'agit de valeurs numériques. La fonction Dialogue ne fait qu'un simple remplacement.

*Important : en sortie de la fonction Dialogue, la variable tableau Mtxt() est inchangée. Elle peut donc être réutilisée pour un autre message (utilisant le même paramètre). On évitera ainsi des répétitions de définition comme c'était le cas dans l'ancienne version.*

*Exemple :*

*Sub Traitement1 ()*

*Mtxt(1) = "Titre de mon traitement" ' Titre des messages*

*If Dialogue(5) = 1 then*

*' Ok traite*

*...*

*M1 = Dialogue(7) ' Fin (Même titre)*

*Else*

*M1 = Dialogue(10) ' Abandon (Même titre)*


*End if*

*End Sub*

## Ajout de commentaires

Les commentaires apportent une information supplémentaire dans le message. C'est le cas pour le message n° 1 de chaque jeu. Un commentaire précisera le nom de l'application destinataire (*ou source*). Il sera égal à "@Appli??" (en abrégé)" pour un nouveau jeu.

Les commentaires ne sont possibles que dans le Label2 et doivent commencer par le caractère "@". Tout le texte, à partir de ce code, sera ignoré à l'affichage.

Le bouton  présent à gauche du champ "Label 2" affiche un panneau indiquant les modalités pratiques d'insertion.

## Ajout d'une spécificité

Dans certains cas, le paramétrage des messages (*par l'application de gestion*) ne permet pas d'obtenir les résultats attendus. C'est le cas notamment pour le cadrage de plusieurs valeurs numériques ou l'ajout d'une image dans le corps du message.

Les spécificités permettent de traiter les messages concernés en ajoutant du code dans la fonction Dialogue. Au moment de l'affichage, si une spécificité est rencontrée, le code correspondant sera exécuté. Le numéro de message n'entre pas en ligne de compte, seul le numéro de spécificité est concerné. De cette façon, pour deux messages de même numéro (*appartenant donc à des jeux différents*), avec pour l'un une spécificité et pour l'autre aucune, seul le premier sera concerné.


Comme pour les commentaires, l'insertion d'une spécificité ne peut se faire que dans le Label2. On utilisera le code "@SP=xx" pour la signaler suivi d'un éventuel commentaire sur sa description. On ne peut insérer qu'un seul item par message (*si plusieurs, seul le 1<sup>er</sup> sera traité*).

"xx" détermine le numéro de la spécificité à exécuter. Valeurs autorisées : 1 à 99 sous la forme "01" ... "99". Le code spécifique étant exécuté au sein de la fonction Dialogue via une instruction "Select Case", les doublons ne sont pas autorisés et en l'absence de codification, la spécificité sera ignorée.

Avec 99 possibilités, on pourra codifier toutes les spécificités (*quel que soit le projet*) directement dans le Module Dialogue de l'application de gestion (*pour effectuer les tests*) puis, une fois au point, réexporter le Module dans les projets concernés (*ou tous*).

Pour s'y retrouver, on précisera en commentaire la ou les applications concernées :

```
Select Case i
  Case 1
    ' Pour Appli xx, Appli zz ..
  ...
```

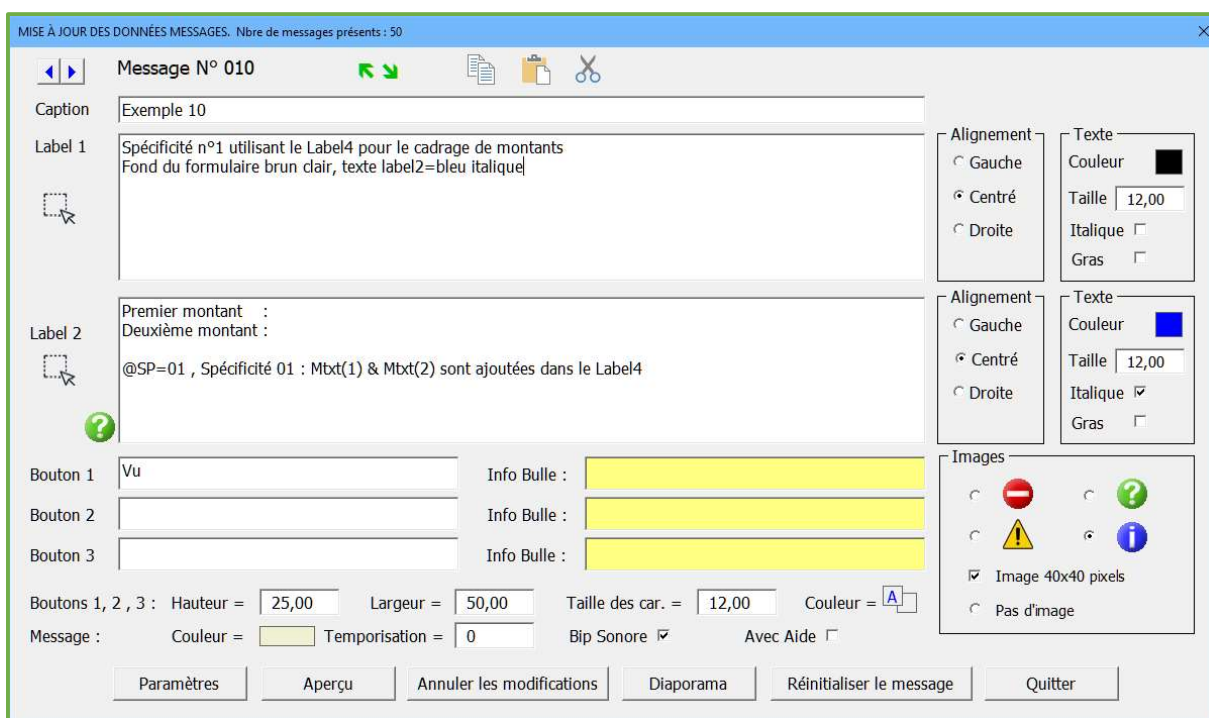
Le bouton  présent à gauche du champ "Label 2" affiche un panneau indiquant les modalités pratiques d'insertion.

### Exemple pour le cadrage de montants

Lorsque l'on souhaite afficher plusieurs montants dans un label, l'alignement vertical des chiffres est toujours compliqué. Chaque caractère ayant sa propre taille, le cadrage n'est jamais satisfaisant ; surtout si l'on souhaite centrer les informations (*texte + montants*).

A moins d'écrire une procédure spécifique pour formater les nombres, se basant sur les twips de chaque caractère, l'utilisation d'un Label dédié aux montants (*ici le Label4*) reste la solution la plus simple pour obtenir un alignement correct.

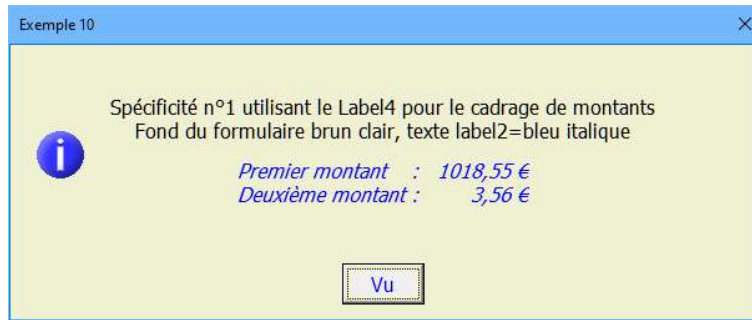
#### Paramétrage du message 10



Modification des paramètres provisoires : Mtxt(1)="1018,55 €", Mtxt(2)="3,56 €"

Codification dans la fonction Dialogue : la spécificité est déjà en place dans la fonction. Voir le code.

## Aperçu



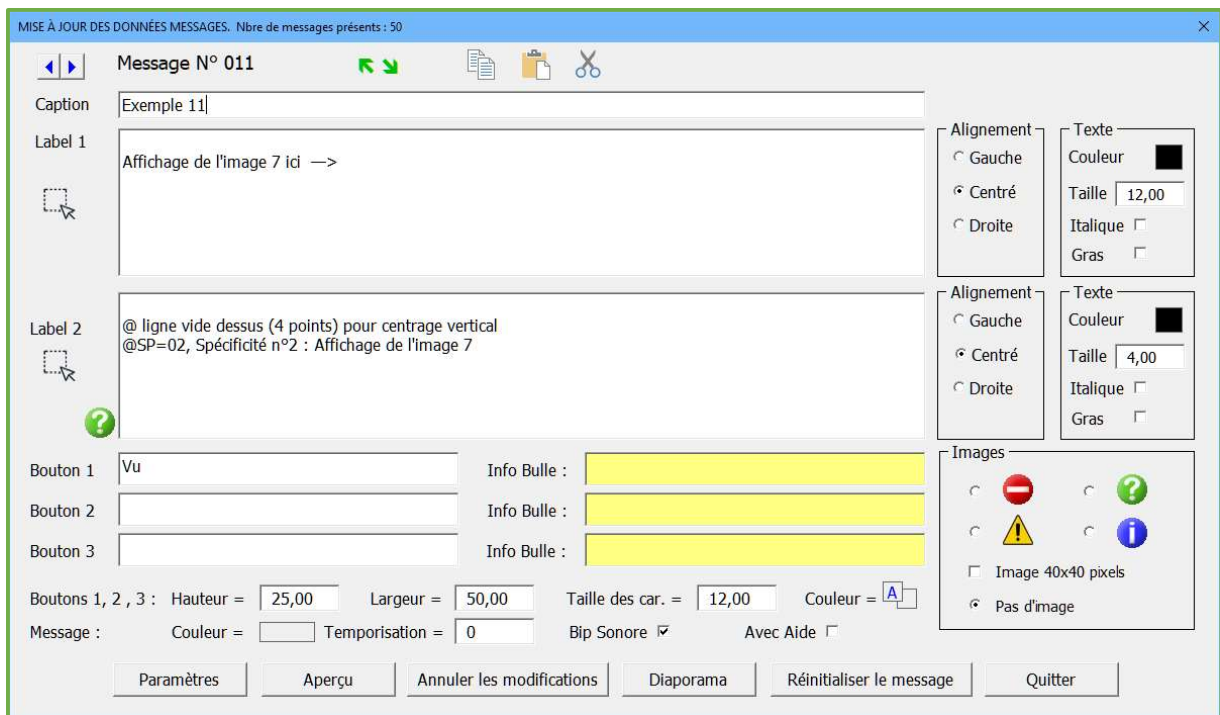
Appel de la fonction depuis une application hôte

```
Mtxt(1) = Format(Montant1, Fmt€) ' Au format Euros  
Mtxt(2) = Format(Montant2, Fmt€)  
M1 = Dialogue(10)
```

(Fmt€ est une constante déclarée dans le Module1  
Public Const Fmt€ As String = " ### ##0.00 €; -# ### ##0.00 €")

## Exemple pour l'ajout d'une image

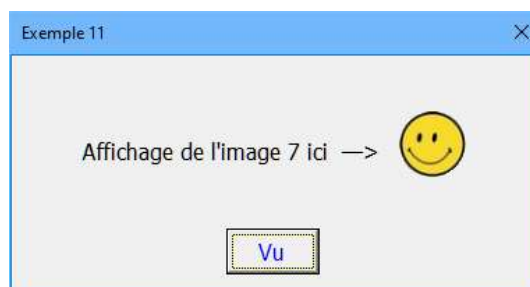
Paramétrage du message 11



Prévoir l'image dans le Formulaire Message : Image7, Propriété Picture.

Codification dans la fonction Dialogue : la spécificité est déjà en place dans la fonction. Voir le code.

## Aperçu



Appel de la fonction depuis une application hôte

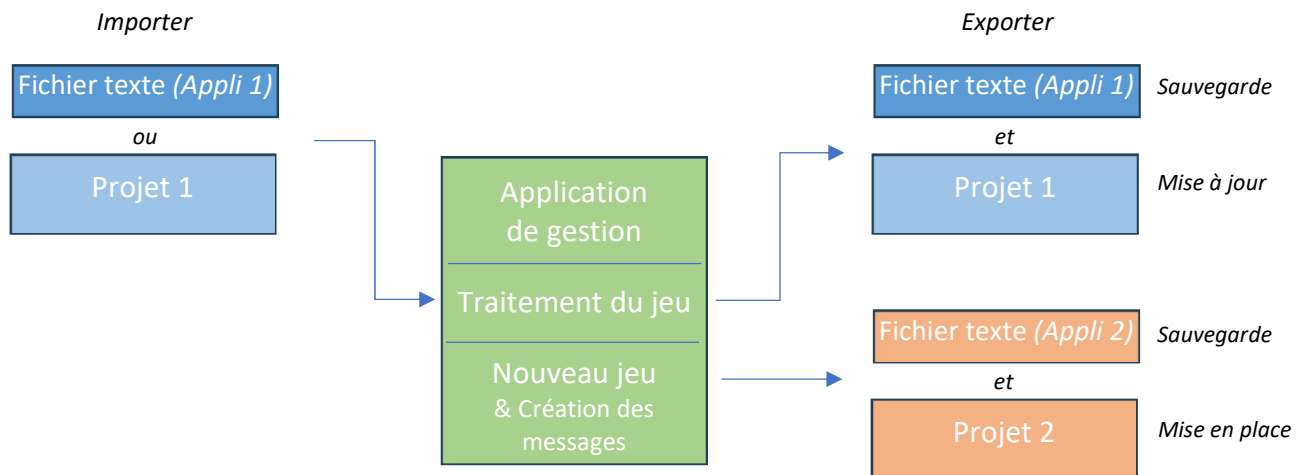
```
M1 = Dialogue(11)
```

Le fichier texte "Datas Msg Exemples de messages ...", fourni en pièce jointe, présente différents exemples de paramétrage. Les messages 10 & 11 correspondent aux exemples précédents.

Importer ce fichier dans l'application de gestion pour les visualiser (*Bouton "Importer Datas"*).

## Utilisation pratique

### Schéma général



#### Créer un nouveau projet

Utilisez le bouton "Créer un nouveau projet" pour créer un classeur prêt à l'emploi. **L'enregistrer au format Xlsm.**

Ou dupliquez, depuis l'explorateur VBA, le "Module Dialogue" et les formulaires "Données\_Msg" & "Message" dans votre classeur.

Créez un nouveau jeu de messages (*si non encore créé*), procédez au paramétrage puis l'exporter vers le nouveau projet. Une fois mis au point, créez une sauvegarde du jeu de messages (*export vers un fichier texte*).

#### Mettre à jour un projet (*Modifier, supprimer ou ajouter un ou plusieurs messages*)

Si le jeu n'est pas chargé en mémoire, l'importer à partir de l'application (*ou à partir d'un fichier de sauvegarde si restauration*).

Faites les mises à jour puis :

- Exportez vers un fichier texte (*nouvelle sauvegarde*),
- Exportez vers l'application (*mise à jour*).

#### Migrer de l'ancienne version à la nouvelle

Commencez par la création du jeu de messages. Pour cela imprimez (*ou placez dans le bloc-notes*) les messages présents dans l'ancienne fonction Dialogue de votre projet. Indiquez dans le message n°1, en commentaire, le nom de votre Appli (*en abrégé*).

Programmez, si nécessaire, les spécificités dans la fonction Dialogue du classeur de gestion.

Testez chaque message en utilisant, si besoin, les paramètres provisoires.

Enregistrez le classeur de gestion sans le fermer.

Ouvrez l'application à modifier : *(faire une sauvegarde au préalable)*

- Supprimez le formulaire "Message" *(ancienne version)*,
- Renommez l'ancienne procédure Dialogue *(Dialogue2 par exemple)*,
- Importez depuis l'application de gestion : *(glissez les via la fenêtre explorateur de projet)*
  - o Le formulaire "Message",
  - o Le formulaire "Données\_Msg" *(il contient le nouveau jeu de messages)*,
  - o Le module Dialogue.
- Supprimez les déclarations *(Publiques)* redondantes *(ne garder que celles présentes dans le Module Dialogue)*.
- Dans le menu Débogage, lancez la compilation. Une erreur sera signalée pour chaque instruction "Dialogue(?,?)". Corrigez l'instruction en supprimant le deuxième paramètre puis relancez la compilation. Vous pouvez aussi procéder par recherche des instructions "Dialogue(".
- Une fois tous les appels à la fonction Dialogue corrigés, supprimez l'ancienne fonction *(que vous avez renommée)*.
- Relancez une compilation pour vérifier que tout est ok.
- Enregistrez le classeur.
- Testez l'application pour vérifier chaque message :
  - o Pour corriger un message :
    - Repassez dans la l'appli de gestion,
    - Modifiez le message *(ou le codage de la spécificité dans le module)*,
    - Réexportez le jeu de messages,
    - Réexportez le module Dialogue si vous avez modifié la partie spécificités,
    - Refaire les tests.
- Sauvegardez votre jeu de messages *(Export vers un fichier texte)*.

## En conclusion

Avec cette nouvelle version, la programmation des messages devient encore plus simple, ils sont stockés, sous une forme réduite, dans un formulaire spécifique, rendant ainsi l'application plus légère.

L'autonomie du module Dialogue permet une implantation plus aisée dans les nouveaux projets.

Les commentaires, présents dans le code de l'application de gestion, permettent de comprendre les mécanismes de traitement. Vous pouvez adapter l'application à vos besoins en y ajoutant ou en modifiant des éléments *(procédures, objets, etc.)*.

Bien que le programme ait été testé maintes fois, il peut s'y trouver encore quelques coquilles résiduelles *(erreurs non traitées par exemple)*. Un retour de votre part sera le bienvenu.

Bonne utilisation