



Manuel de référence de l'environnement

INSTALLATION	9
INSTALLATION EN RESEAU	9
L'ENVIRONNEMENT	10
VUE GENERALE	10
CHOIX DES CIBLES	11
PALETTES	11
MONTRER OU CACHER LA FENETRE DU PROJET OU LES FENETRE DE MESSAGES	12
AFFICHER L'ESPACE DE TRAVAIL EN MODE PLEIN ECRAN	12
RACCOURCIS CLAVIERS	12
LICENCES	13
ENREGISTRER UNE LICENCE	13
ENVOYER UN CODE UTILISATEUR A IRAI	14
<i>Envoyer un fichier par email (la meilleur solution)</i>	15
<i>Copier le code utilisateur dans le message d'un email</i>	15
<i>Par fax (déconseillé)</i>	15
<i>Par téléphone (fortement déconseillé)</i>	16
ENTRER LE CODE DE VALIDATION / DEBRIDAGE	16
<i>Débridage par un fichier reçu par email</i>	16
<i>Débridage par un code reçu dans le texte dans un email</i>	16
<i>Débridage par un code reçu par fax ou au téléphone</i>	16
MODIFIER UNE LICENCE	16
DEPLACER UNE LICENCE D'UN ORDINATEUR A UN AUTRE	16
LICENCES RESEAU	17
<i>Ajouter une licence réseau</i>	19
<i>Modifier une licence</i>	19
<i>Connexion des postes clients</i>	19
LE PROJET	20
IMPORTER UNE APPLICATION D'UNE ANCIENNE VERSION D'AUTOMGEN	20
IMPORTER UN PROJET CREE AVEC UN AUTRE ATELIER LOGICIEL	20
GENERER UN FICHIER EXECUTABLE DISTRIBUABLE GRATUITEMENT	20
MODIFIER LES PROPRIETES DU PROJET	21
<i>Modifier les options de sécurité</i>	21
<i>Options avancées</i>	21
<i>Interface utilisateur</i>	21
<i>Modèle</i>	21
<i>GO automatique</i>	21
LE NAVIGATEUR	23
FOLIOS	24
<i>Ajouter un nouveau folio</i>	24
<i>Importer des folios d'anciennes versions d'AUTOMGEN, importer des folios CADEPA</i>	25
<i>Modifier l'ordre de compilation des folios</i>	25
<i>Supprimer un folio de la liste</i>	26
<i>Exporter un folio vers un fichier « .GR7 »</i>	26
<i>Copier, Couper, Coller un folio</i>	26
<i>Renommer un folio</i>	26
<i>Modifier les propriétés d'un folio</i>	26
SYMBOLES	27
<i>Créer une table de symboles</i>	27
<i>Importer une table de symboles</i>	28
CONFIGURATION	28
<i>Post-processeurs</i>	28
<i>Options du compilateur</i>	28
DOCUMENTATION	28
FICHIERS GENERES	29
<i>Générer la liste des instructions en code pivot</i>	29

<i>Générer la liste des références croisées</i>	29
<i>Post-processeurs</i>	29
MISE AU POINT.....	29
<i>Voir et modifier une variable ou une table de variables</i>	29
OBJETS IRIS.....	30
<i>Ajouter un objet IRIS 2D</i>	30
<i>Supprimer un objet IRIS 2D</i>	31
<i>Montrer ou cacher un objet IRIS 2D</i>	31
<i>Copier, couper, coller un objet IRIS 2D</i>	32
<i>Ajouter un nouvel objet IRIS 2D sur un pupitre</i>	32
<i>Modifier les propriétés d'un objet IRIS 2D</i>	32
<i>Définir un modèle d'objet accessible dans l'assistant</i>	33
<i>Importer un objet IRIS 2D d'une version précédente d'AUTOMGEN</i>	33
<i>Créer un pupitre IRIS 3D</i>	34
RESSOURCES.....	34
<i>Ajouter un fichier dans les ressources</i>	34
<i>Supprimer un fichier contenu dans les ressources</i>	34
<i>Renommer un fichier contenu dans les ressources</i>	34
<i>Modifier un fichier contenu dans les ressources</i>	34
<i>Ajouter et convertir des fichiers 3D STUDIO dans les ressources</i>	34
MODULES EXTERNES.....	34
DESSINER DES PROGRAMMES.....	36
DESSINER AVEC L'ASSISTANT.....	36
DESSINER AVEC LE MENU CONTEXTUEL.....	37
DESSINER AVEC LA PALETTE.....	37
<i>Enrichir et personnaliser la palette</i>	37
DESSINER AVEC LES TOUCHES DU CLAVIER.....	37
<i>Bloc d'effacement</i>	37
<i>Blocs de liaison</i>	37
<i>Blocs Grafset</i>	38
<i>Blocs Logigrammes</i>	39
<i>Blocs Ladder</i>	40
<i>Blocs Action</i>	40
<i>Blocs Test</i>	41
<i>Blocs Organigramme</i>	41
<i>Blocs Bloc fonctionnel</i>	42
DOCUMENTER LES ELEMENTS DE PROGRAMME.....	42
AJOUTER DES SYMBOLES.....	43
EXECUTER UNE APPLICATION.....	45
<i>Exécuter facilement une application</i>	45
<i>Mettre fin à l'exécution</i>	45
<i>Uniquement compiler</i>	45
<i>Stopper la compilation</i>	45
<i>Se connecter à un automate ou installer l'exécuteur PC</i>	45
<i>Se déconnecter d'un automate ou désinstaller l'exécuteur PC</i>	45
<i>Mettre la cible en mode RUN</i>	45
<i>Mettre la cible en mode STOP</i>	45
<i>Initialiser la cible</i>	45
<i>Faire un cycle programme sur la cible (généralement non supporté sur les automates)</i>	46
<i>Activer la visualisation dynamique</i>	46
LE COMPILATEUR.....	47
MODIFIER LES OPTIONS DU COMPILATEUR.....	47
VISUALISER LES MESSAGES DE COMPILATION.....	47
LOCALISER UNE ERREUR.....	47
EXECUTION DES PROGRAMMES SUR PC.....	49
CONFIGURER LE NOMBRE DE VARIABLES.....	49

VARIABLES SYSTEME DE L'EXECUTEUR PC.....	50
MODIFIER LA PERIODE D'EXECUTION	51
PILOTER DES ENTREES / SORTIES	51
REFERENCES IRIS 2D.....	53
MODIFIER LA VISIBILITE DES OBJETS.....	53
MODIFIER LES CARACTERISTIQUES D'UN OBJET	54
<i>Effacer un objet.....</i>	54
<i>Dimensionner un objet.....</i>	54
<i>Déplacer un objet.....</i>	54
<i>Passer un objet en mode « Utilisateur ».....</i>	54
<i>Passer un objet en mode « Configuration ».....</i>	54
<i>Modifier les caractéristiques d'un objet.....</i>	54
VERROUILLEZ L'ACCES A LA CONFIGURATION DE TOUS LES OBJETS	55
OBJETS DE BASE, OBJETS PREDEFINIS	55
LISTE DES OBJETS DE BASE	55
<i>L'objet « Pupitre ».....</i>	55
<i>L'objet « Bouton et voyant ».....</i>	55
<i>L'objet « Objet ».....</i>	55
<i>L'objet « Valeur digitale ».....</i>	55
<i>L'objet « Ecran, clavier, liste de messages ».....</i>	56
<i>L'objet « Son ».....</i>	56
<i>L'objet « Archivage de données ».....</i>	56
<i>L'objet « Programme ».....</i>	56
<i>L'objet « Boîte de dialogue ».....</i>	56
<i>L'objet « Valeur analogique ».....</i>	56
PRISE EN MAIN	56
<i>Etape 1.....</i>	56
<i>Etape 2.....</i>	57
<i>Etape 3.....</i>	57
<i>Etape 4.....</i>	57
<i>Etape 5.....</i>	57
<i>Etape 6.....</i>	58
<i>Etape 7.....</i>	59
REALISER UNE APPLICATION DE SUPERVISION AUTONOME.....	60
SYNTAXE POUR L'ACCES A L'ETAT DES VARIABLES	60
<i>Etat booléen.....</i>	60
<i>Etat numérique.....</i>	61
<i>Modification d'état.....</i>	61
<i>Ordres spéciaux.....</i>	61
<i>Echanges entre objets.....</i>	62
DETAIL DE L'OBJET « PUPITRE »	62
<i>Onglet « Aspect ».....</i>	62
<i>Onglet « Bitmap ».....</i>	63
<i>Onglet « Liens ».....</i>	64
<i>Onglet « Options ».....</i>	64
<i>Onglet « Enfants ».....</i>	64
<i>Onglet « Externe ».....</i>	64
DETAIL DE L'OBJET « BOUTON VOYANT »	65
<i>Onglet « Aspect ».....</i>	65
<i>Onglet « Liens ».....</i>	65
<i>Onglet « Options ».....</i>	66
DETAIL DE L'OBJET « VALEUR DIGITALE »	68
<i>Onglet « Aspect ».....</i>	68
<i>Onglet « Textes ».....</i>	69
<i>Onglet « Liens ».....</i>	69
DETAIL DE L'OBJET « VALEUR ANALOGIQUE »	69
<i>Onglet « Aspect ».....</i>	69
<i>Onglet « Liens ».....</i>	70
<i>Onglet « Bornes ».....</i>	70

Onglet « Graduations »	71
DETAIL DE L'OBJET « ECRAN, CLAVIER, LISTE DE MESSAGES »	72
Liens avec l'application	72
Les classes de messages	73
Onglet « Aspect »	73
Onglet « Liens »	74
Onglet « Liste »	74
Onglet « Options »	75
Onglet « Messages »	76
DETAIL DE L'OBJET « ARCHIVAGE DE DONNEES »	76
Onglet « Aspect »	76
Onglet « Données »	77
Onglet « Options »	78
Onglet « Tables »	79
Onglet « Courbe »	79
Onglet « Graduations »	81
Onglet « Grille »	82
DETAIL DE L'OBJET « OBJET »	83
Onglet « Aspect »	83
Onglet « Liens »	84
Onglet « Formes »	85
Onglet « Bitmap »	85
Onglet « Wmf »	85
Onglet « Couleurs »	86
Onglet « Jauge »	87
Onglet « Capteur »	87
Onglet « Options »	88
Techniques avancées	88
DETAIL DE L'OBJET « SON »	89
Onglet « Aspect »	89
Onglet « Sons »	89
DETAIL DE L'OBJET « BOITE DE DIALOGUE »	89
Onglet « Aspect »	89
ONGLET « LIENS »	90
Onglet « Messages »	91
DETAIL DE L'OBJET « PROGRAMME »	91
Répartition du temps d'exécution	91
Affichage	92
Syntaxe	92
Déclaration des variables	92
Ecriture du programme	93
Constantes	93
Affectation	93
Calculs	93
Tests	94
Boucles	94
Adresse d'une variable ou d'un tableau de variables	95
Liste des fonctions	95
Messages d'erreurs	101
Onglet « Aspect »	102
Onglet « Programme »	103
EXEMPLES IRIS 2D.....	104
EXEMPLE D'OBJET COMPOSE	104
EXEMPLE D'UTILISATION DE L'OBJET « ECRAN, CLAVIER, LISTE A MESSAGES » COMME LISTE A MESSAGES ..	107
EXEMPLE D'UTILISATION DE L'OBJET ECRANCLA COMME TERMINAL	108
EXEMPLE D'APPLICATION COMPOSEE DE PLUSIEURS PAGES	109
EXEMPLE D'UTILISATION DE L'OBJET OBJET	109
EXEMPLES D'UTILISATION DE L'OBJET ARCHIVE	114
EXEMPLE D'UTILISATION DE L'OBJET PROG	115

EXEMPLES D'APPLICATION DE SUPERVISION 1	115
EXEMPLE D'APPLICATION DE SUPERVISION 2	116
EXEMPLE DE SIMULATION D'UNE PARTIE OPERATIVE 1	117
EXEMPLE DE SIMULATION D'UNE PARTIE OPERATIVE 2	118
REFERENCES IRIS 3D.....	119
CREER UN PUPITRE IRIS 3D	120
AJOUTER DES FICHIERS 3D AU PROJET	120
CONFIGURER LES OBJETS.....	122
AJOUTER DES OBJETS DANS LE MONDE 3D	123
ENLEVER UN FICHIER 3D DES RESSOURCES	123
ENLEVER UN OBJET DU MONDE 3D	123
APPLIQUER UN COMPORTEMENT A UN OBJET.....	123
<i>Nom des variables AUTOMGEN</i>	124
<i>Ajouter une translation</i>	125
<i>Ajouter une rotation</i>	127
<i>Ajouter un changement de couleur</i>	127
<i>Ajouter un lien</i>	128
<i>Ajouter un autre comportement</i>	129
EXEMPLE IRIS 3D.....	130

Installation

Si vous installez à partir du CD-ROM d'AUTOMGEN, placez celui-ci dans votre lecteur de CD-ROM.

L'installation se lance automatiquement.

Si ce n'est pas le cas, lancez l'exécutable « Setup.exe » qui se trouve à la racine du CD-ROM.

Le CD-ROM contient AUTOMGEN7, ACROBAT READER (pour l'accès à la documentation en ligne), CROSSROADS (un utilitaire de conversion 3D) et DIRECTX 8 (pour la gestion de l'affichage 3D).

Si vous installez à partir de fichiers téléchargés sur Internet, lancez l'exécution des exécutables téléchargés. Le site Internet permet également le téléchargement des modules ACROBAT READER, CROSSROADS et DIRECTX 8.

Installation en réseau

AUTOMGEN peut être installé sans problème en réseau.

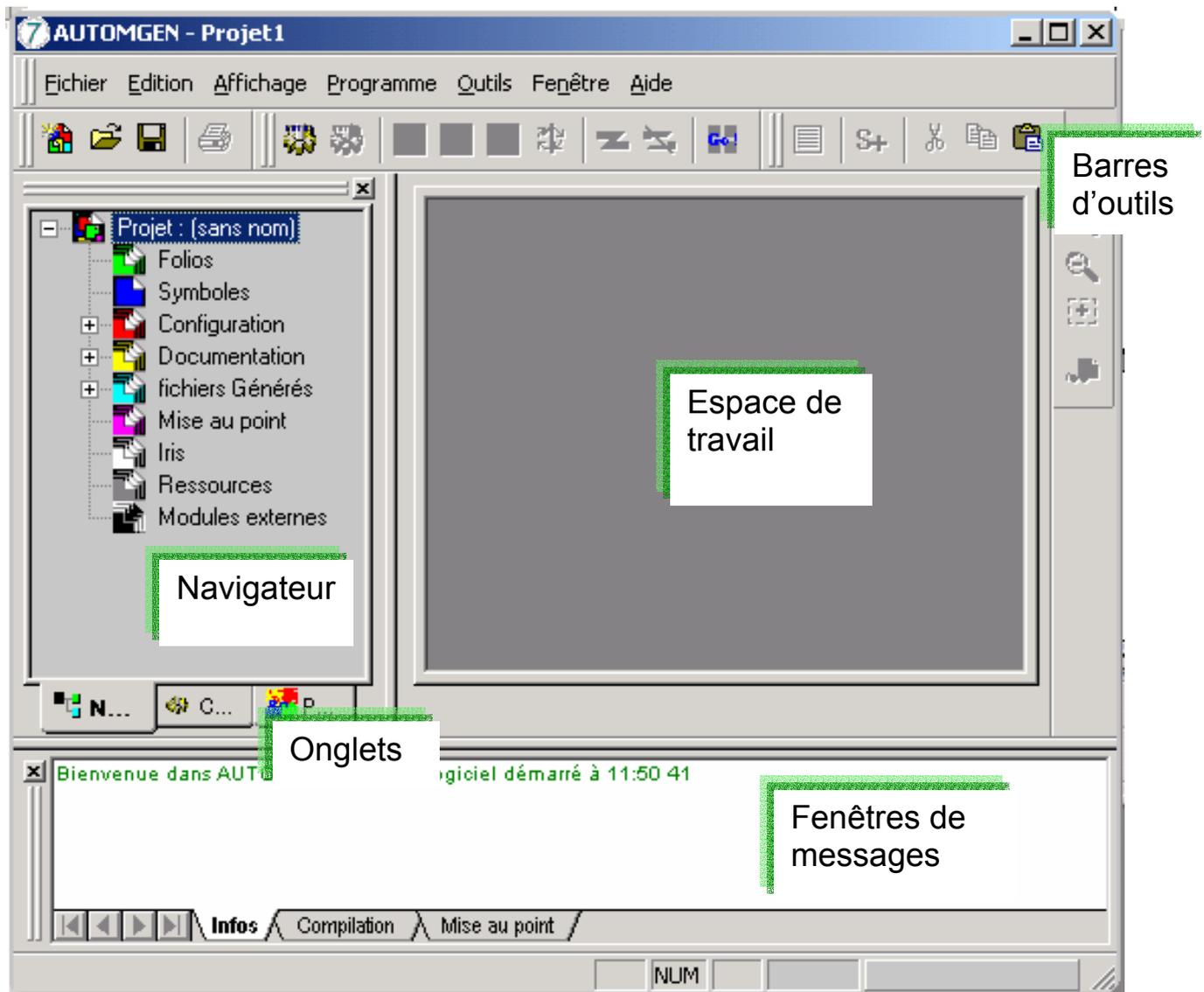
Exécutez la procédure d'installation sur le PC « serveur » (assurez vous d'avoir l'ensemble des droits d'accès au moment de l'installation).

Pour lancer AUTOMGEN, sur les PCs clients, créez un raccourci vers l'exécutable « autom7.exe » du répertoire d'installation d'AUTOMGEN sur le PC serveur.

Pour faire apparaître les post-processeurs dans l'onglet cible des PCs clients, installez les post-processeurs sur les PCs clients puis désinstallez AUTOMGEN sur les PCs clients (ceci afin de créer uniquement les lignes dans la fenêtrés « Cibles »).

L'environnement

Vue générale



La fenêtre principale d'AUTOMGEN

L'environnement est entièrement paramétrable. Les barres d'outils peuvent être déplacées (en les saisissant par ) et paramétrées (menu « Outils/Personnaliser l'environnement »).

L'état de l'environnement est sauvegardé lorsqu'on le quitte. Cet état peut aussi être sauvegardé dans un fichier projet (voir les options du projet).

Choix des cibles

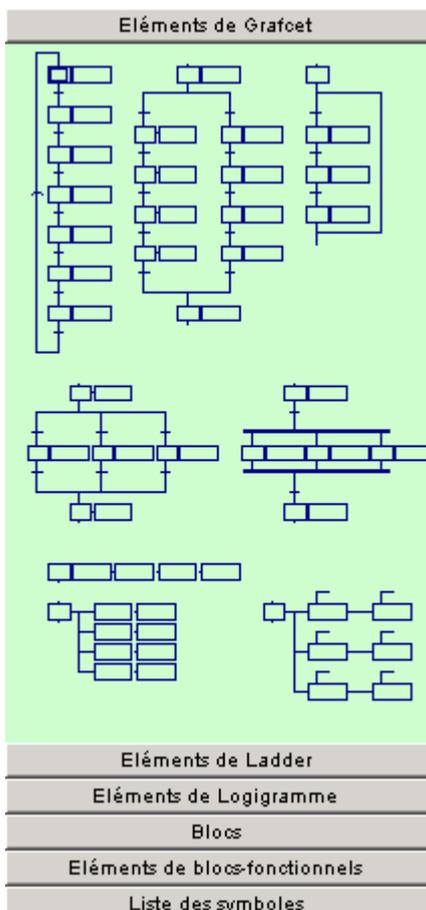
En bas de la fenêtre du navigateur se trouve un onglet « Cibles » permettant d'accéder à la liste des post-processeurs installés.

Actif	Nom	Version
	Exécuteur PC	7.000
	PL7 (Tsx 37 & Ts...	7.000
	PL72	7.000
	STEP7 (S7200)	7.000

La cible active est marquée d'un coche verte. L'accès aux cibles apparaissant en grisé n'est pas autorisé par rapport à la licence installée (voir le chapitre « Licences » pour plus de détails). Pour modifier la cible courante, double cliquez sur la ligne correspondante.

Palettes

En bas de la fenêtre du navigateur se trouvent un onglet « Palette » permettant d'accéder à des éléments de dessin de programmes.



La palette donne un ensemble d'éléments qui peuvent être sélectionnés et déposés sur les folios. Pour sélectionner un élément, cliquez avec le bouton gauche de la souris dans la palette, étirez la sélection, relâchez le bouton de la souris, cliquez dans la zone sélectionnée et déplacez la zone vers le folio.

La palette contient également la liste des symboles du projet. Vous pouvez les saisir et les faire glisser sur un test ou une action sur un folio.

Montrer ou cacher la fenêtre du projet ou les fenêtre de messages

Sélectionnez l'option « Projet » ou « Messages » dans le menu « Fenêtre ».

Afficher l'espace de travail en mode plein écran

Sélectionnez l'option « Plein écran » dans le menu « Affichage ». Cliquez sur  pour sortir du mode plein écran.

Raccourcis claviers

Les raccourcis claviers sont inscrits dans les menus. Des raccourcis « masqués » sont également utilisables :

CTRL + ALT + F8	Sauvegarder le projet sous la forme d'un exécutable
CTRL + ALT + F9	Sauvegarder le projet
CTRL + ALT + F10	Accéder aux propriétés du projet
CTRL + ALT + F11	Montrer ou cacher la fenêtre AUTOMGEN



L'environnement est entièrement paramétrable, son état est sauvegardé lorsqu'on quitte AUTOMGEN. On peut cacher les fenêtres de l'environnement. Le menu « Fenêtres » permet de les afficher de nouveau. L'espace de travail peut être affiché en mode plein écran. Les onglets en bas de la fenêtre du navigateur permettent d'accéder au choix du post-processeur courant ainsi qu'à la palette de dessin.

Licences

Une licence définit les droits d'utilisation d'AUTOMGEN. Les éléments suivants sont déterminés par une licence :

- le nombre d'entrées / sorties tout ou rien utilisables,
- les post-processeurs utilisables,
- le nombre d'utilisateurs (licence réseau uniquement).

Enregistrer une licence

Lorsque vous installez AUTOMGEN, vous pouvez l'utiliser gratuitement pendant une durée de 40 jours.

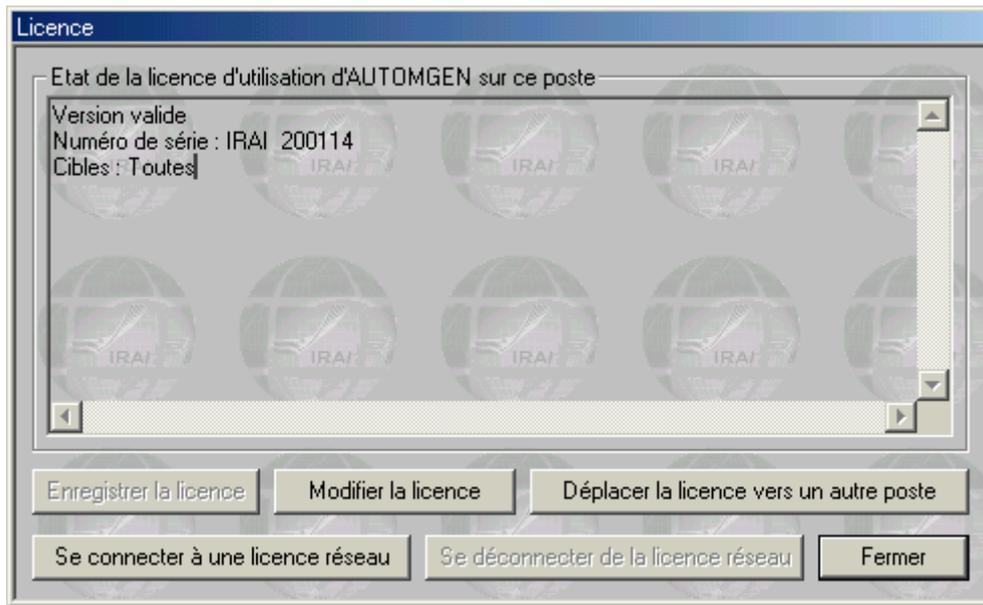
Pendant ces 40 jours, vous devez enregistrer votre licence.

Pour enregistrer votre licence, vous devez communiquer à IRAI :

- le numéro de série imprimé sur une étiquette collée sur la boîte du logiciel ou, à défaut, la référence de votre bon de livraison ou de votre commande,
- un code utilisateur fourni par le logiciel identifiant le PC sur lequel vous avez installé le produit.

Vous recevrez en retour un code de débridage (on parle aussi de code de validation).

L'option « Licence » du menu « Fichier » d'AUTOMGEN vous permet de visualiser le statut de votre licence et d'obtenir un code utilisateur (en cliquant sur « Enregistrer la licence »).



Etat de la licence.

Un code utilisateur est valide pendant une durée de 10 jours.

Il peut donc s'écouler un maximum de 10 jours entre le moment où vous communiquez un code utilisateur à IRAI et le moment où vous saisissez un code de débridage fourni par IRAI.

Envoyer un code utilisateur à IRAI

Plusieurs méthodes sont à votre disposition. La méthode d'échange des codes par email est fortement recommandée pour limiter les risques d'erreur.



Une seule erreur sur le code entraînera un échec de l'enregistrement d'une licence.

Envoyer un fichier par email (la meilleur solution)

Enregistrer ou modifier une protection

Vous êtes sur le point d'enregistrer ou de modifier votre licence d'utilisation (après acquittement des droits d'utilisation si nécessaire). Votre code utilisateur doit être fourni à la société IRAI qui vous fournira en retour le code de validation.

Vous pouvez communiquer votre code utilisateur :

- par Téléphone : 04 66 54 91 30,
- par Fax : 04 66 54 91 33
- ou par email : contact@irai.com

Les informations suivantes doivent être communiquées : vos coordonnées complètes et, le cas échéant, une référence de commande ou de bon de livraison.

Code utilisateur, attention : '0' est un ZERO et 'O' la lettre

FILEK	6P4MJ	1803G	0601I	UT7UR	N1803	G06G1	4UCL6	GLH8K	006IP	00
--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	-----------

Code de validation

--	--	--	--	--	--	--	--	--	--	--

La boîte de dialogue d'enregistrement d'une licence.

Pour générer un fichier contenant votre code utilisateur, cliquez sur « Sauvegarder le code utilisateur dans un fichier ». Vous pouvez ensuite transmettre ce fichier portant l'extension « .a7u » comme pièce jointe d'un email à envoyer à l'adresse contact@irai.com.

Copier le code utilisateur dans le message d'un email

En cliquant sur « Copier le code utilisateur vers le presse papier », vous pourrez ensuite coller le code dans le corps du message d'un email pour le transmettre à l'adresse email contact@irai.com.

Par fax (déconseillé)

En cliquant sur « Copier le code utilisateur vers le presse papier », vous pourrez ensuite coller le code dans un document et l'envoyer par fax au 04 66 54 91 33. Evitez si possible l'écriture manuscrite, prenez soin d'imprimer avec une fonte permettant de différencier la lettre « O » du chiffre zéro.

Par téléphone (fortement déconseillé)

En téléphonant au 04 66 54 91 30. Prenez soin de différencier la lettre « O » du chiffre zéro. Attention aux consonnes difficiles à différencier au téléphone (« S » et « F » par exemple).

Entrer le code de validation / débridage

Débridage par un fichier reçu par email

Si vous avez reçu un fichier « .a7v » par email, enregistrez le fichier reçu sur votre disque dur, cliquez sur « Lire un code de validation depuis un fichier », et sélectionnez le fichier.

Débridage par un code reçu dans le texte dans un email

Sélectionnez le code dans le texte du message (prenez soin de ne sélectionner que le code et de ne pas ajouter d'espaces à la fin). Cliquez sur « Coller un code de validation depuis le presse papier ».

Débridage par un code reçu par fax ou au téléphone

Saisissez le code dans les cases se trouvant sous l'intitulé « Code de validation ».

Modifier une licence

La modification d'une licence consiste à faire évoluer les éléments autorisés par la licence (ajout d'un post-processeur par exemple).

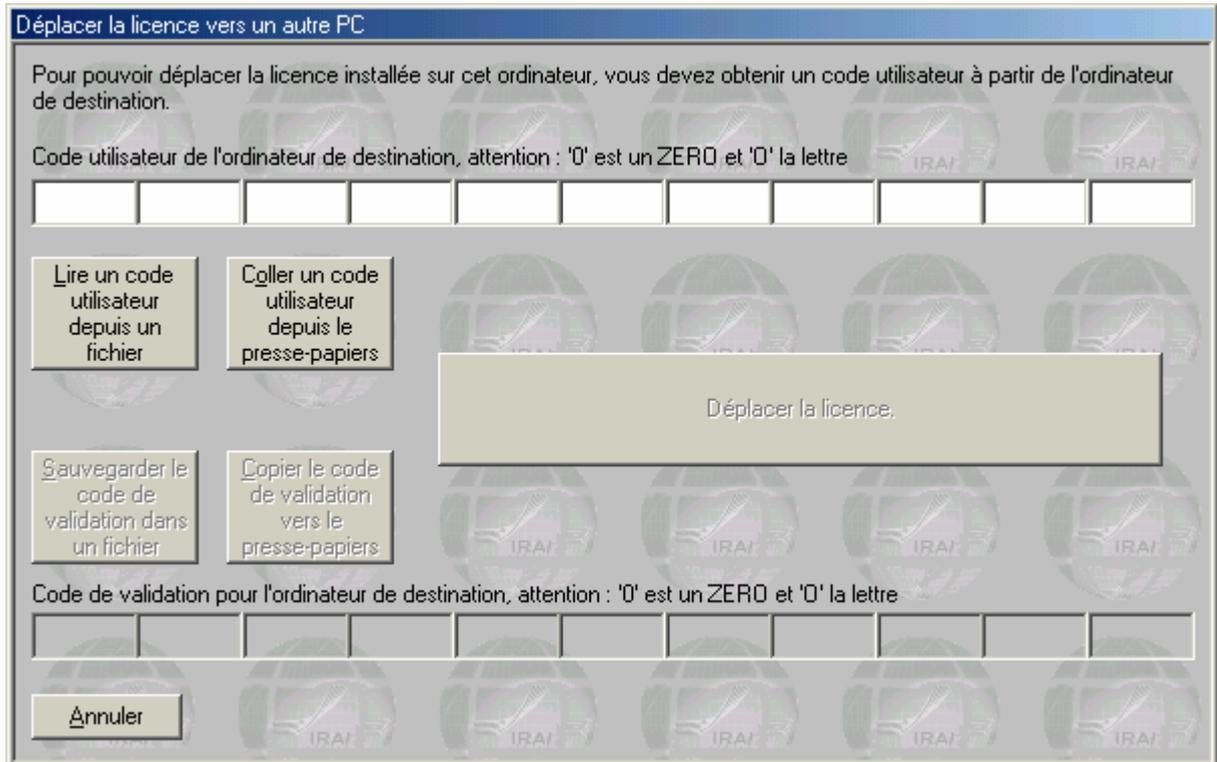
La procédure de modification d'une licence est en tout point identique à l'enregistrement.

Déplacer une licence d'un ordinateur à un autre

Cette procédure est la plus complexe. Les instructions qui suivent doivent être scrupuleusement respectées pour obtenir un bon résultat. Dans les explications qui suivent, le PC « source » désigne l'ordinateur où se trouve la licence et le PC « de destination » l'ordinateur où doit être déplacé la licence.

- 1- si ce n'est pas déjà fait, installez AUTOMGEN sur le PC de destination,
- 2- générez un fichier de code utilisateur « .a7u » sur le PC de destination et déplacer ce fichier vers le PC source (en le copiant sur une disquette par exemple),

- 3- sur le PC source, choisissez l'option « Déplacer la licence vers un autre poste »,

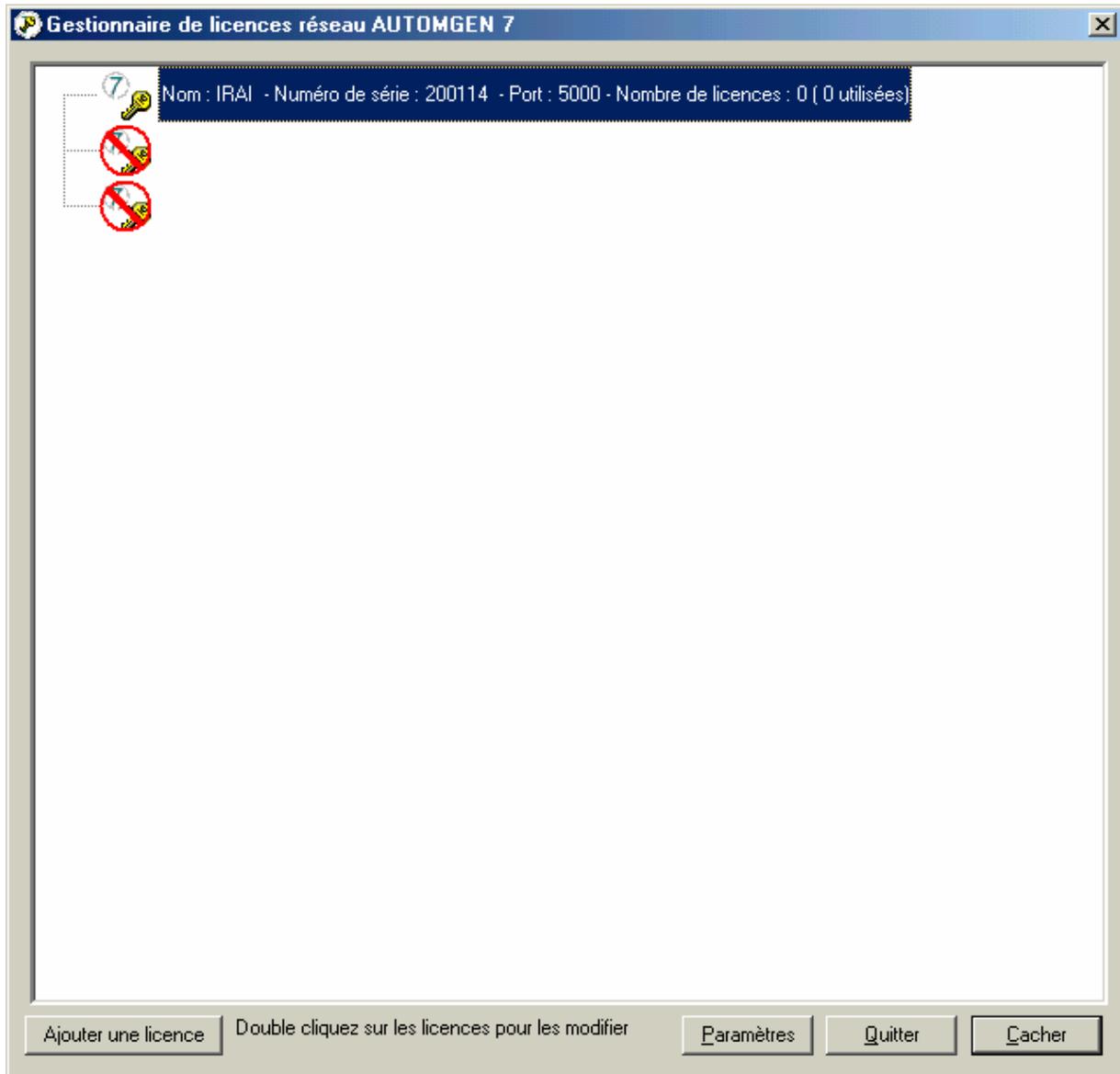


La boîte de dialogue de déplacement d'une licence

- 4- sur le PC source, cliquez sur « Lire un code utilisateur depuis un fichier » et sélectionnez le fichier « .a7u » provenant du PC de destination,
- 5- sur le PC source cliquez sur « Déplacer la licence »,
- 6- sur le PC source cliquez sur « Sauvegarder le code de validation dans un fichier », recopier le fichier « .a7v » généré vers le PC de destination,
- 7- sur le PC de destination cliquez sur « Lire un code de validation depuis un fichier » et sélectionnez le fichier « .a7v » en provenance du PC source.

Licences réseau

L'exécutable « akey7.exe » est le gestionnaire de licence réseau. Cet exécutable doit être lancé sur un des ordinateurs du réseau. Le réseau doit permettre l'utilisation du protocole TCP IP. Au lancement, le gestionnaire de licences réseau est masqué et seul une icône  apparaît dans la barre des tâches de WINDOWS. Pour visualiser la fenêtre du gestionnaire de licence réseau, double cliquez sur l'icône  dans la barre des tâches.



Le gestionnaire de licences réseau

Jusqu'à 16 licences différentes peuvent être gérées par le gestionnaire de licences réseau. Une licence réseau est caractérisée par un nombre d'utilisateurs et un type de droit (nombre d'entrées / sorties tout ou rien et post-processeurs utilisables). Pour chaque licence est affiché le nombre d'utilisateur(s) possible(s), le nombre d'utilisateur(s) connecté(s) et la liste des utilisateurs connectés (en train d'utiliser AUTOMGEN) sous forme d'arborescence attachée à chaque licence. A chaque licence est associée un numéro de port (une valeur numérique à partir de 5000 par défaut). Le premier numéro de port utilisé peut être configuré en cliquant sur « Paramètres ».

Ajouter une licence réseau

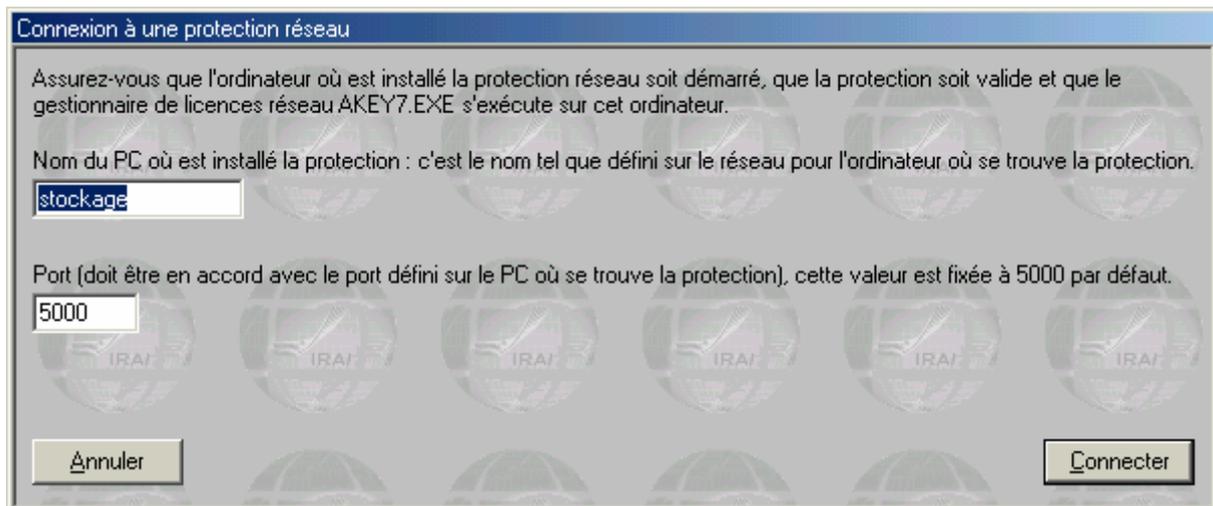
En cliquant sur « Ajouter une licence » vous pouvez ajouter une licence. Le principe d'enregistrement des licences est le même que pour les versions monopostes.

Modifier une licence

Double cliquez sur les licences pour les modifier. La procédure de modification des licences est identique à celle utilisée pour les licences monopostes.

Connexion des postes clients

Cliquez sur « Se connecter à une licence réseau » pour connecter un poste client à une licence réseau.



Connexion à une licence réseau

Le nom du PC (tel que vue sur le réseau) où a été lancé « akey7.exe » doit être fourni ainsi que le numéro du port correspondant à la licence souhaitée.



Vous devez enregistrer votre licence auprès de IRAI (contact@irai.com) en envoyant par email votre code utilisateur (menu « Fichier/Licence »). Le gestionnaire de licences réseau permet de gérer plusieurs licences sur un des PCs du réseau TCP IP.

Le projet

La notion de projet est très forte dans AUTOMGEN⁷. Un projet regroupe l'ensemble des éléments composant une application. Le navigateur (voir page 23) affiche sous forme arborescente tous les éléments d'un projet (folios, symboles, configuration, objets IRIS, etc ...).

Le nouveau format de fichier d'AUTOMGEN⁷ (fichiers portant l'extension « .AGN ») encapsule tous les éléments d'un projet.

Lorsque vous sauvegardez un fichier « .AGN » vous avez l'assurance de sauvegarder la totalité des éléments d'une application. Vous pouvez échanger facilement et efficacement les applications créées avec AUTOMGEN.

Les fichiers « .AGN » sont compactés avec la technologie « ZIP », nul n'est donc besoin de les compresser pour les échanger, leur taille est déjà optimisée.

Importer une application d'une ancienne version d'AUTOMGEN

Vous devez importer l'ensemble des folios (fichiers « .GR7 ») et l'éventuel fichier des symboles (fichier « .SYM »). Pour cela utilisez les procédures d'importation décrites dans les chapitres suivants.

Importer un projet créé avec un autre atelier logiciel

(disponible dans le courant du premier semestre 2002)

La commande « Importer » du menu « Fichier » permet d'importer des fichiers « .FEF » provenant des ateliers logiciels SCHNEIDER.

Générer un fichier exécutable distribuable gratuitement

La commande « Générer un exécutable » du menu « Fichier » permet de générer un exécutable à partir du projet en cours (un fichier « .EXE » exécutable sur PC sous WINDOWS). La « visionneuse » AUTOMGEN est intégrée automatiquement a l'exécutable généré (l'utilisateur de l'exécutable n'a pas besoin d'AUTOMGEN). Cette visionneuse permet d'utiliser l'application sans la modifier. Vous pouvez ainsi déployer vos applications facilement. L'exécutable généré est libre de droit. Cette technique est typiquement utilisée pour produire une application de supervision.

Modifier les propriétés du projet

Cliquez avec le bouton droit de la souris sur l'élément « Projet » dans le navigateur et choisissez « Propriétés » dans le menu.

Modifier les options de sécurité

Vous pouvez restreindre l'accès en lecture ou en modification du projet par des mots de passe.

Options avancées

« Sauver l'aspect de l'environnement avec le projet » : si coché, alors la position des fenêtres ainsi que l'aspect des barres d'outils sont sauvegardés dans le fichier « .AGN ». A l'ouverture du projet, ces éléments seront restitués.

« Cacher la fenêtre principale au lancement ... » : si coché, la fenêtre d'AUTOMGEN est cachée à l'ouverture du projet. Seuls les objets d'IRIS incorporés au projet sont visibles. Cette option est typiquement utilisée pour créer des applications « packagées » ne laissant apparaître que des objets IRIS. La combinaison de touche [CTRL] + [F11] permet de faire réapparaître la fenêtre AUTOMGEN.

Les autres options permettent d'agir sur l'affichage de la fenêtre AUTOMGEN à l'ouverture du projet.

Interface utilisateur

« Interdire la configuration des objets IRIS » : si coché, la configuration des objets IRIS ne peut être modifiée par l'utilisateur.

Les autres options permettent de modifier le comportement de l'interface utilisateur.

Modèle

« Ce projet est un modèle de document » : si coché, à son ouverture, toutes les options et documents qu'il contient servent de modèle à la création d'un nouveau projet. Cette fonctionnalité permet de créer des configurations standard pouvant être chargées au lancement d'AUTOMGEN (un fichier de symbole par défaut ou une configuration automate par défaut par exemple).

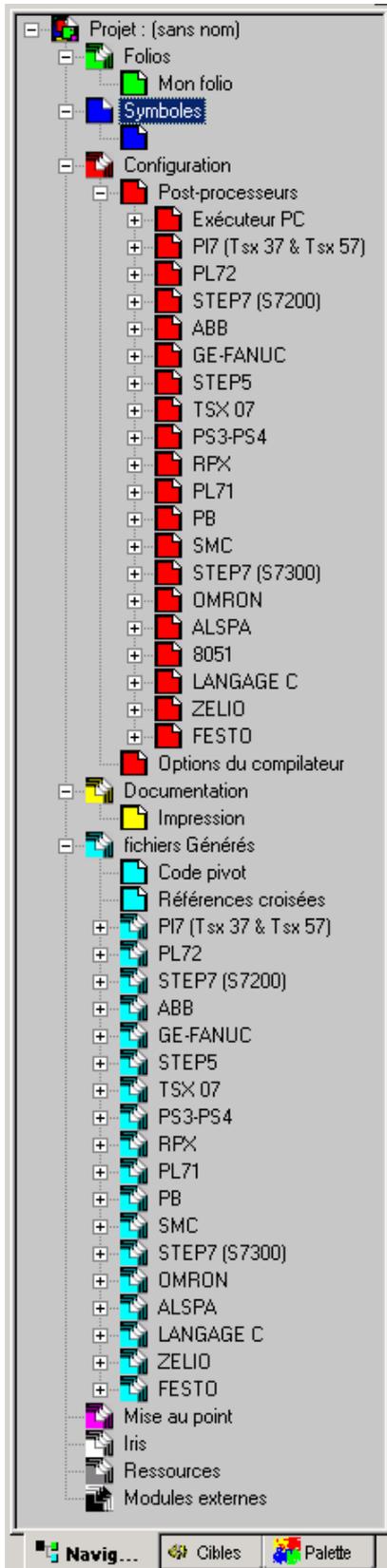
GO automatique

«Go automatique au lancement du projet » : si coché, l'exécution de l'application est automatique à l'ouverture du projet.



Le projet permet de regrouper l'ensemble des éléments d'une applications d'AUTOMGEN. Ainsi regroupés, les éléments ne forme plus qu'un fichier compacté qui porte l'extension « .AGN ». Les modèles de projet permettent de gérer facilement différentes configurations du logiciel. La génération d'exécutables rend aisé le déploiement des applications.

Le navigateur



Élément central de la gestion des applications, le navigateur permet un accès rapide aux différents éléments d'une application : folios, symboles, configuration, impressions, objets IRIS, etc ...

Les icônes « + » et « - » permettent de développer ou de rétracter les éléments du projet.

Les actions sur le navigateur sont réalisées en double cliquant sur les éléments (ouverture de l'élément) ou en cliquant avec le bouton droit (ajout d'un nouvel élément au projet, action spéciale sur un élément, etc...).

Certaines opérations sont réalisées en saisissant les éléments et en les déplaçant dans le navigateur (drag and drop).

Les couleurs (rappelées en général sur le fond des documents dans l'espace de travail) permettent d'identifier la famille des éléments.

L'arborescence du navigateur.

Folios

Un folio est une page sur laquelle est dessinée un programme ou une partie de programme.

La manipulation des folios est simplifiée à l'extrême dans AUTOMGEN⁷. Les ordres de chaînages de folios nécessaires dans les versions précédentes ne sont plus utilisés. Pour que plusieurs folios soient compiler ensemble, il suffit qu'ils se trouvent dans le projet. Les icônes associées aux folios dont les suivantes :

-  folio normal,
-  folio normal (exclu de la compilation),
-  folio contenant une expansion de macro-étape,
-  folio contenant un programme de bloc-fonctionnel,
-  folio contenant une tâche,
-  folio contenant une tâche (exclu de la compilation).

Des icônes sont barrés d'une croix indiquent un folio fermé (non visible dans l'espace de travail). Double cliquer sur une icône de ce type ouvre (montre) le folio associé.

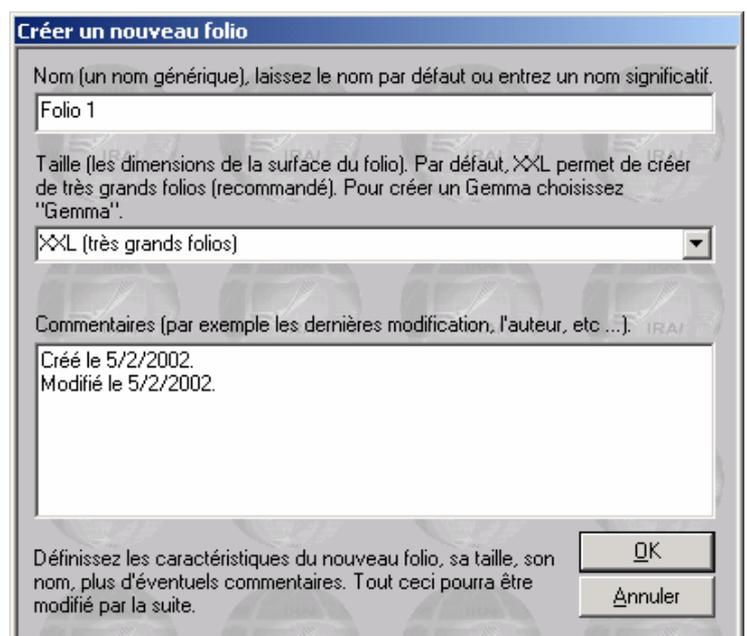
Ajouter un nouveau folio

Cliquer avec le bouton droit sur de la souris sur l'élément « Folios » dans le navigateur puis choisissez « Ajouter un nouveau folio ».

Choisissez la taille du folio (XXL est le format recommandé, les autres formats sont ceux des anciennes versions d'AUTOMGEN, GEMMA doit uniquement être utilisé pour créer un modèle GEMMA).

Le nom du folio peut être quelconque mais doit rester unique pour chaque folio du projet.

La zone commentaire est laissée à votre discrétion pour l'évolution des modifications ou autres informations relatives à chacun des folios.



Créer un nouveau folio

Nom (un nom générique), laissez le nom par défaut ou entrez un nom significatif.
Folio 1

Taille (les dimensions de la surface du folio). Par défaut, XXL permet de créer de très grands folios (recommandé). Pour créer un Gemma choisissez "Gemma".
XXL (très grands folios)

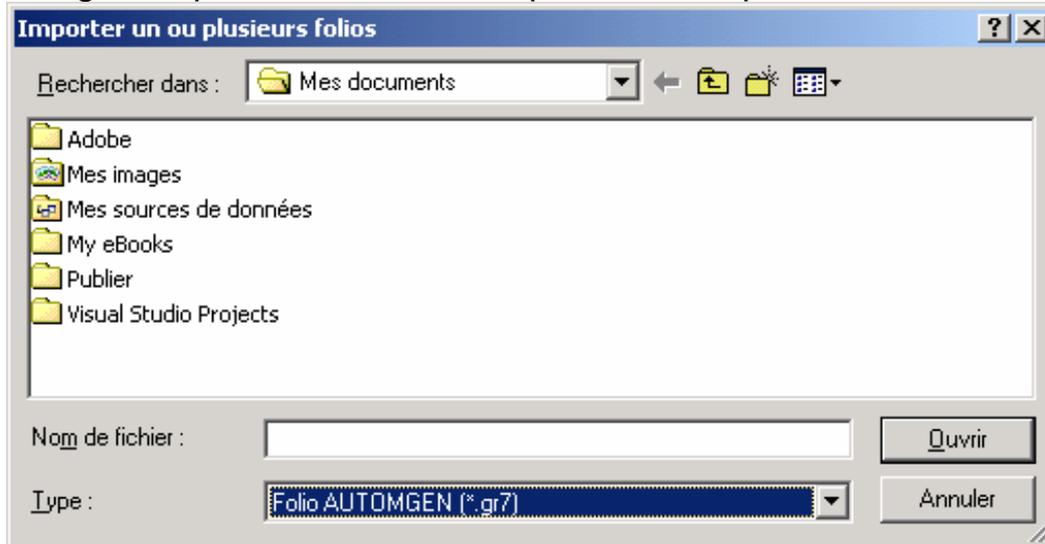
Commentaires (par exemple les dernières modification, l'auteur, etc ...). IRAI
Créé le 5/2/2002.
Modifié le 5/2/2002.

Définissez les caractéristiques du nouveau folio, sa taille, son nom, plus d'éventuels commentaires. Tout ceci pourra être modifié par la suite.

OK
Annuler

Importer des folios d'anciennes versions d'AUTOMGEN, importer des folios CADEPA

Cliquez avec le bouton droit de la souris sur l'élément « Folios » dans le navigateur puis choisissez « Importer un ou plusieurs folios existants ».



Sélection d'un ou plusieurs folios à importer.

Choisissez dans la liste « Type » le type du folio à importer « AUTOMGEN » ou « CADEPA » puis cliquez sur OK.



Certaines restrictions sont applicables à l'importation des folios CADEPA :

- les numéros d'étapes doivent être uniques (pas d'utilisation du même numéro d'étape sur plusieurs folios),
- les renvois doivent être convertis en liens dans CADEPA avant de pouvoir être importés.

En maintenant la touche [CTRL] enfoncée, vous pouvez sélectionner et importer plusieurs folios.

Modifier l'ordre de compilation des folios

Les folios sont compilés dans l'ordre dans lequel ils sont listés dans le projet. Pour modifier cet ordre, cliquez sur un folio avec le bouton gauche de la souris dans le navigateur et déplacez-le dans la liste.

Supprimer un folio de la liste

Cliquez avec le bouton droit de la souris sur le folio à supprimer dans le navigateur et choisissez « Supprimer » dans le menu.

Exporter un folio vers un fichier « .GR7 »

Cliquez avec le bouton droit de la souris sur le folio à supprimer dans le navigateur et choisissez « Exporter » dans le menu.

Copier, Couper, Coller un folio

Cliquez avec le bouton droit de la souris sur le folio dans le navigateur et choisissez « Copier / couper » dans le menu. Pour coller, cliquez avec le bouton droit de la souris sur l'élément « Folios » dans le navigateur et choisissez « Coller ».

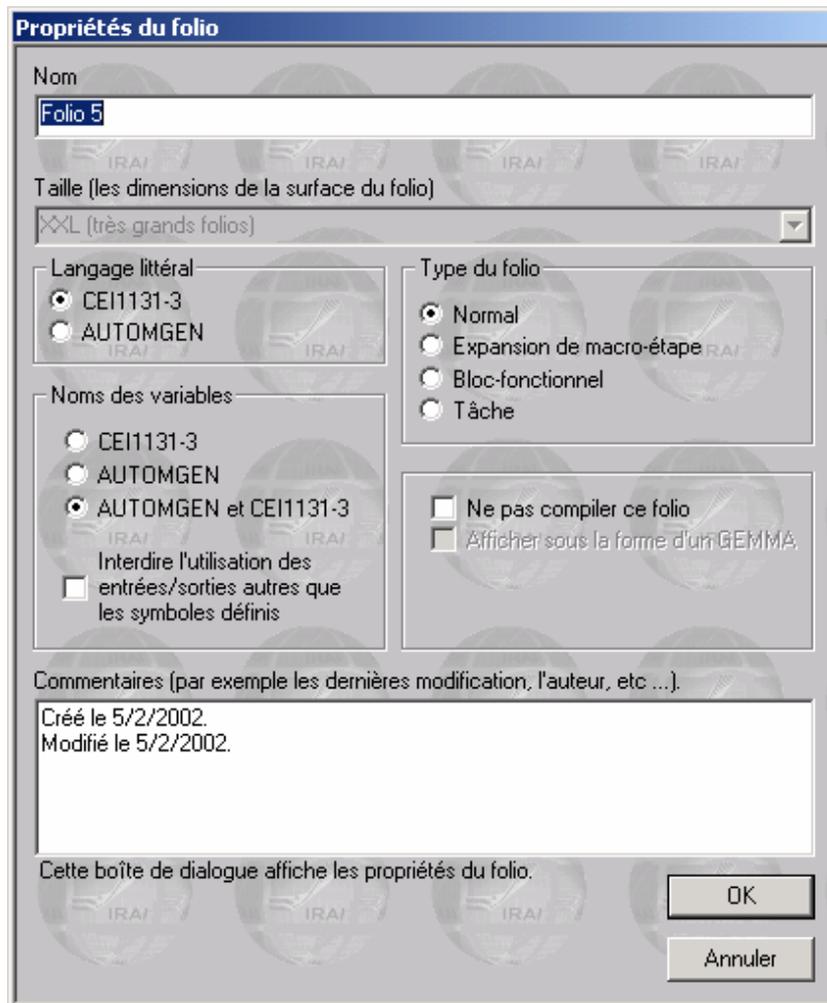
Cette option permet de copier ou de transférer des folios d'un projet à un autre.

Renommer un folio

Voir ci-après « Modifier les propriétés ».

Modifier les propriétés d'un folio.

Cliquez avec le bouton droit de la souris sur le folio dans le navigateur et choisissez « Propriétés » dans le menu.



Vous pouvez modifier le nom du folio, la syntaxe utilisée pour le langage littéral et le nom des variables. L'option « Ne pas compiler ce folio » permet d'exclure le folio de la compilation. L'option « Afficher sous la forme d'un GEMMA » disponible uniquement si le format du folio est GEMMA permet d'afficher et de modifier le folio sous la forme d'un GEMMA. L'option « Interdire l'utilisation des entrées / sorties autres que les symboles définis » interdit l'utilisation des variables i , $\%i$, o $\%q$ non attribuées à des symboles. La zone « commentaires » est laissée à votre discrétion.

Symboles

La liste des symboles donne la correspondance entre des noms « symboliques » et des noms de variables. Un projet ne peut contenir qu'une seule table de symboles.

Créer une table de symboles

Cliquez avec le bouton droit de la souris sur l'élément « Symboles » dans le navigateur et choisissez « Créer une table de symboles » dans le menu.

Importer une table de symboles

Cliquez avec le bouton droit de la souris sur l'élément « Symboles » dans le navigateur et choisissez « Importer une table de symboles » dans le menu.

Configuration

Post-processeurs

Sous cette rubrique se trouve tous les éléments de configuration des post-processeurs (voir le manuel de référence des post-processeurs pour plus d'informations).

Options du compilateur

Double cliquez sur cet élément pour modifier le réglage des options du compilateur.

Documentation

Permet d'accéder à la fonction d'impression de dossier (double clic sur l'élément « Impression »). Vous pouvez imprimer un dossier complet composé d'une page de garde, de la table de références croisées, de la liste des symboles et des folios. La fonction d'aperçu avant impression permet de visualiser l'ensemble de ces éléments.

Fichiers générés

Générer la liste des instructions en code pivot

En double cliquant sur l'élément « Code pivot » vous générez un listing en langage littéral bas niveau (le code pivot d'AUTOMGEN). L'observation du code généré est généralement réservée à des spécialistes soucieux de comprendre les méthodes de traductions utilisées par le compilateur.

Générer la liste des références croisées

Un double clic sur l'élément « Références croisées » génère et affiche la liste des variables utilisées dans l'application avec leurs éventuelles variables automates associées ainsi que le nom du ou des folios où elles sont utilisées.

Post-processeurs

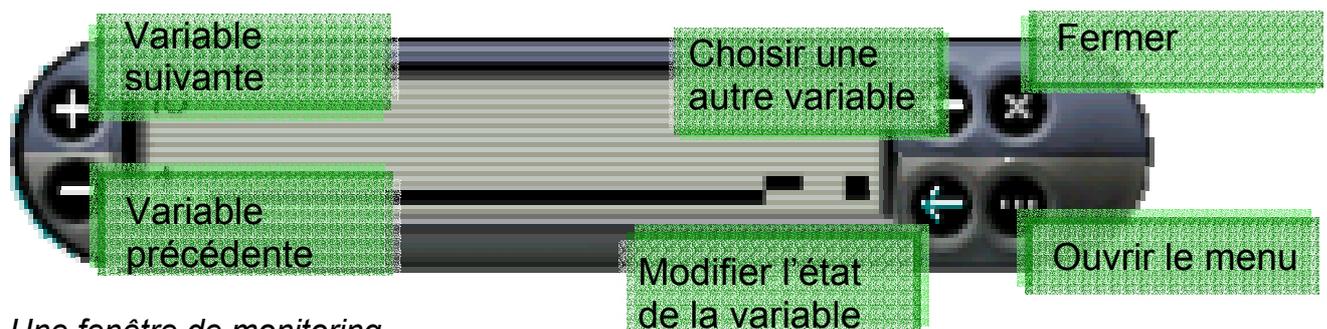
Les autres éléments concernent les fichiers générés par les post-processeurs : listes d'instructions en langage automate.

Mise au point

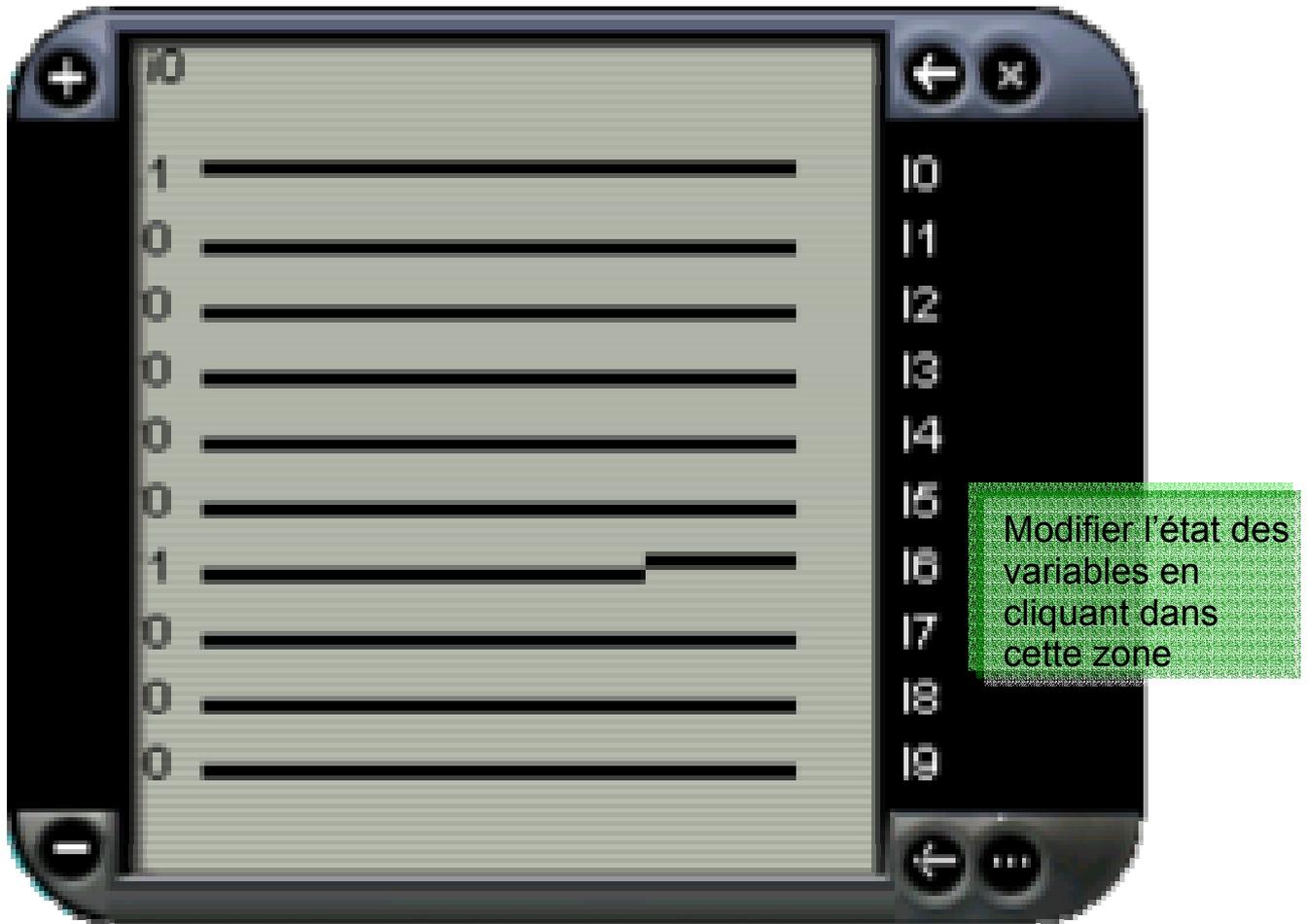
Regroupe des outils permettant la visualisation et la modification de l'état des variables.

Voir et modifier une variable ou une table de variables

En cliquant avec le bouton droit sur l'élément « Mise au point » et en choisissant « Monitoring », vous ouvrez un élément vous permettant d'observer l'état d'une variable ou d'une table de variables.



Une fenêtre de monitoring.



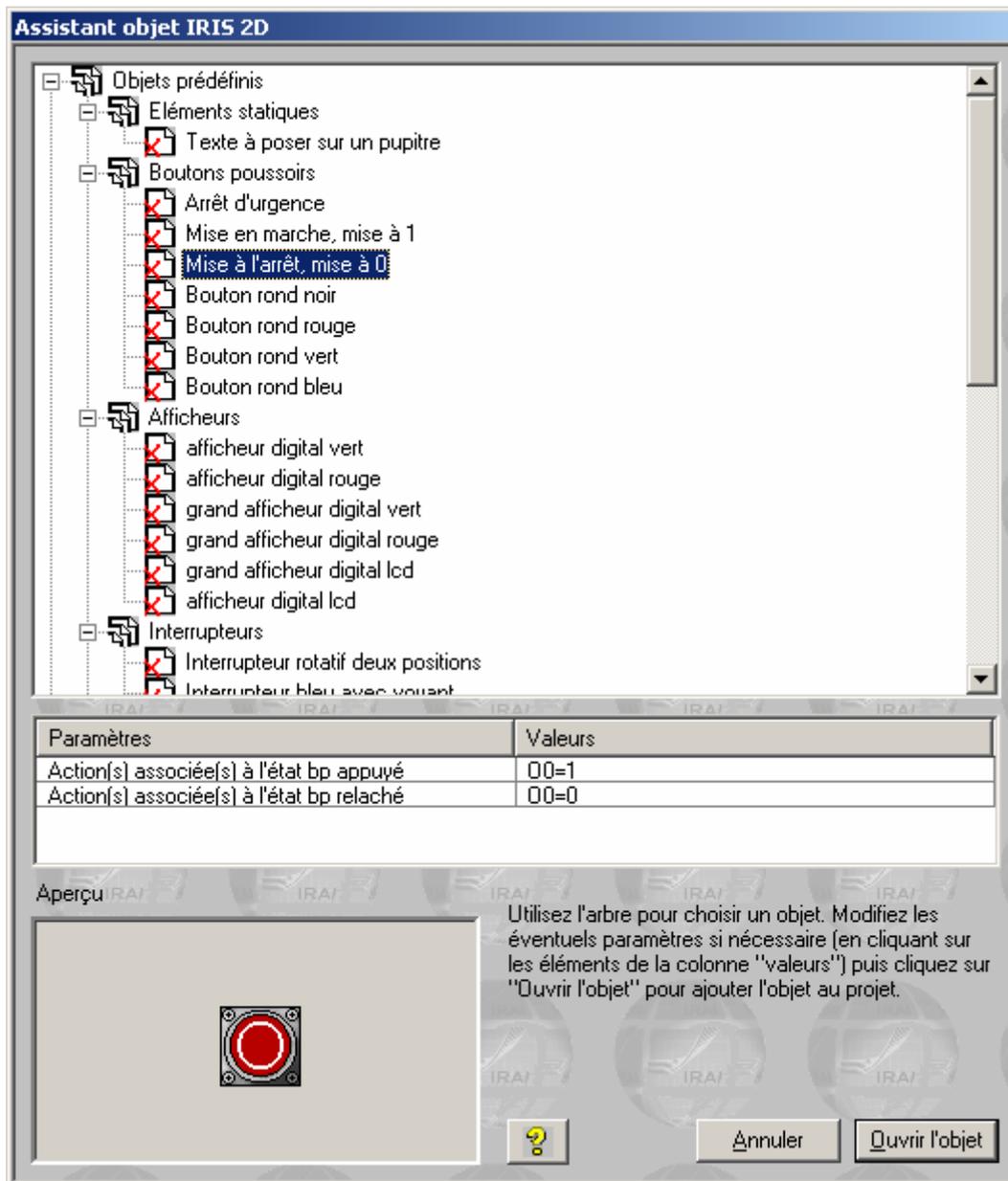
La fenêtre monitoring en mode « Table de variables ».

Objets IRIS

Les objets IRIS 2D permettent de créer des pupitres, des applications de supervision et des applications de simulation de parties opératives 2D). IRIS 3D permet de créer des applications de simulation de parties opératives 3D. Chaque objet IRIS 2D apparaît dans l'arborescence du projet (voir les chapitres Références IRIS 2D et Références IRIS 3D pour plus de détails).

Ajouter un objet IRIS 2D

En cliquant avec le bouton droit de la souris sur l'élément « Ajouter un objet IRIS 2D ». L'assistant de sélection d'un objet vous permet de le choisir et de le paramétrer.



L'assistant de sélection d'un objet IRIS 2D.

Supprimer un objet IRIS 2D

Cliquez avec le bouton droit de la souris sur l'objet IRIS dans le navigateur et choisissez « Supprimer » dans le menu.

Montrer ou cacher un objet IRIS 2D

Cliquez avec le bouton droit de la souris sur l'objet IRIS dans le navigateur et choisissez « Montrer/cacher » dans le menu.

Copier, couper, coller un objet IRIS 2D

Cliquez avec le bouton droit de la souris sur l'objet IRIS dans le navigateur et choisissez « Copier » ou « Coller » dans le menu.

Pour coller un objet IRIS, cliquez avec le bouton droit de la souris sur l'élément « IRIS » dans le navigateur et choisissez « Coller ».

Pour coller un objet IRIS dans un pupitre, choisissez « Coller » dans le menu du pupitre ou cliquez avec le bouton droit de la souris sur le pupitre dans le navigateur et choisissez « Coller ».

Ajouter un nouvel objet IRIS 2D sur un pupitre

Choisissez « Ajouter un objet » dans le menu du pupitre ou cliquez avec le bouton droit de la souris sur le pupitre dans le navigateur et choisissez « Ajouter un objet sur le pupitre » dans le menu (pour plus de détails sur le pupitre, reportez vous au chapitre L'objet « Pupitre »)

Modifier les propriétés d'un objet IRIS 2D

Cliquez avec le bouton droit de la souris sur l'objet IRIS dans le navigateur et choisissez « Propriétés ». Pour les objets de plus haut niveau (les objets parents), des propriétés spéciales sont accessibles :

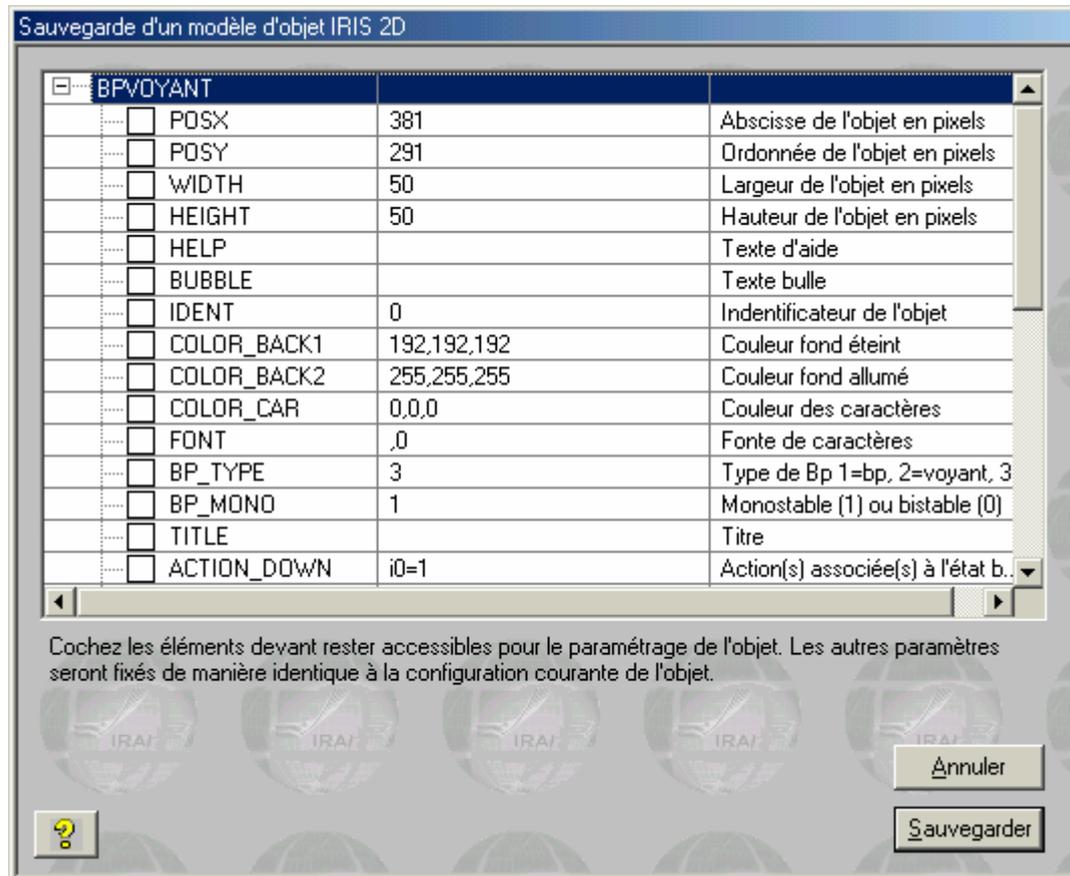


Les propriétés des objets de haut niveau.

La visibilité détermine sous quelle condition l'objet est visible ou caché. L'option de réinitialisation permet de replacer un objet dans son état initial à l'enclenchement de la visualisation dynamique (typiquement utilisé pour les applications de simulation de PO).

Définir un modèle d'objet accessible dans l'assistant

Cliquer sur l'objet IRIS avec le bouton droit de la souris dans le navigateur et choisissez « Enregistrer comme modèle » dans le menu.



La sélection des paramètres modifiables par les utilisateurs de vos modèles.

Vous pouvez sélectionner la liste des paramètres devant rester accessibles à l'utilisateur dans l'assistant. En cliquant sur « Sauvegarder », vous enregistrez votre modèle d'objet. Le répertoire de stockage des modèles d'objet est « <répertoire d'installation d'AUTOMGEN>\i2d\lib ». Un sous répertoire nommé « mes objets » vous est réservé pour sauvegarder vos propres modèles.

Importer un objet IRIS 2D d'une version précédente d'AUTOMGEN

Cliquez avec le bouton droit de la souris sur l'élément « IRIS » dans le navigateur et choisissez « Importer des objets IRIS 2D ». Sélectionnez un ou plusieurs fichiers « .AOF ».

Créer un pupitre IRIS 3D

Cliquez avec le bouton droit de la souris sur l'élément « IRIS » dans le navigateur et choisissez « Ajouter un pupitre IRIS 3D » (voir le chapitre consacré à IRIS 3D pour plus de détails).

Ressources

Cet élément du projet permet d'ajouter tout type de fichier au projet. Les fichiers ainsi ajoutés feront partie intégrante du projet et seront sauvegardés avec les autres éléments. Pour faire référence au pseudo répertoire où se trouve les ressources, le mot clé « <RESDIR> » peut être utilisé dans le nom des répertoires spécifiés dans AUTOMGEN. Les objets IRIS peuvent par exemple faire référence à des bitmaps se trouvant dans les ressources.

Ajouter un fichier dans les ressources

Cliquez avec le bouton droit de la souris sur l'élément « Ressources » dans le navigateur et choisissez « Ajouter » dans le menu.

Supprimer un fichier contenu dans les ressources

Cliquez avec le bouton droit de la souris sur le fichier ressource dans le navigateur et choisissez « Supprimer ».

Renommer un fichier contenu dans les ressources

Cliquez avec le bouton droit de la souris sur le fichier ressource dans le navigateur et choisissez « Renommer ».

Modifier un fichier contenu dans les ressources

Cliquez avec le bouton droit de la souris sur le fichier ressource dans le navigateur et choisissez « Modifier ».

Ajouter et convertir des fichiers 3D STUDIO dans les ressources

Les fichiers 3D STUDIO peuvent être convertis en fichiers .x et ajoutés dans les ressources en cliquant avec le bouton droit de la souris sur l'élément « Ressources » du navigateur et en choisissant « Importer un ou des fichiers 3D » (voir le chapitre Références IRIS 3D pour plus de détails).

Modules externes

Ces éléments sont réservés à des modules exécutables développés par des tiers et interfacés avec AUTOMGEN.



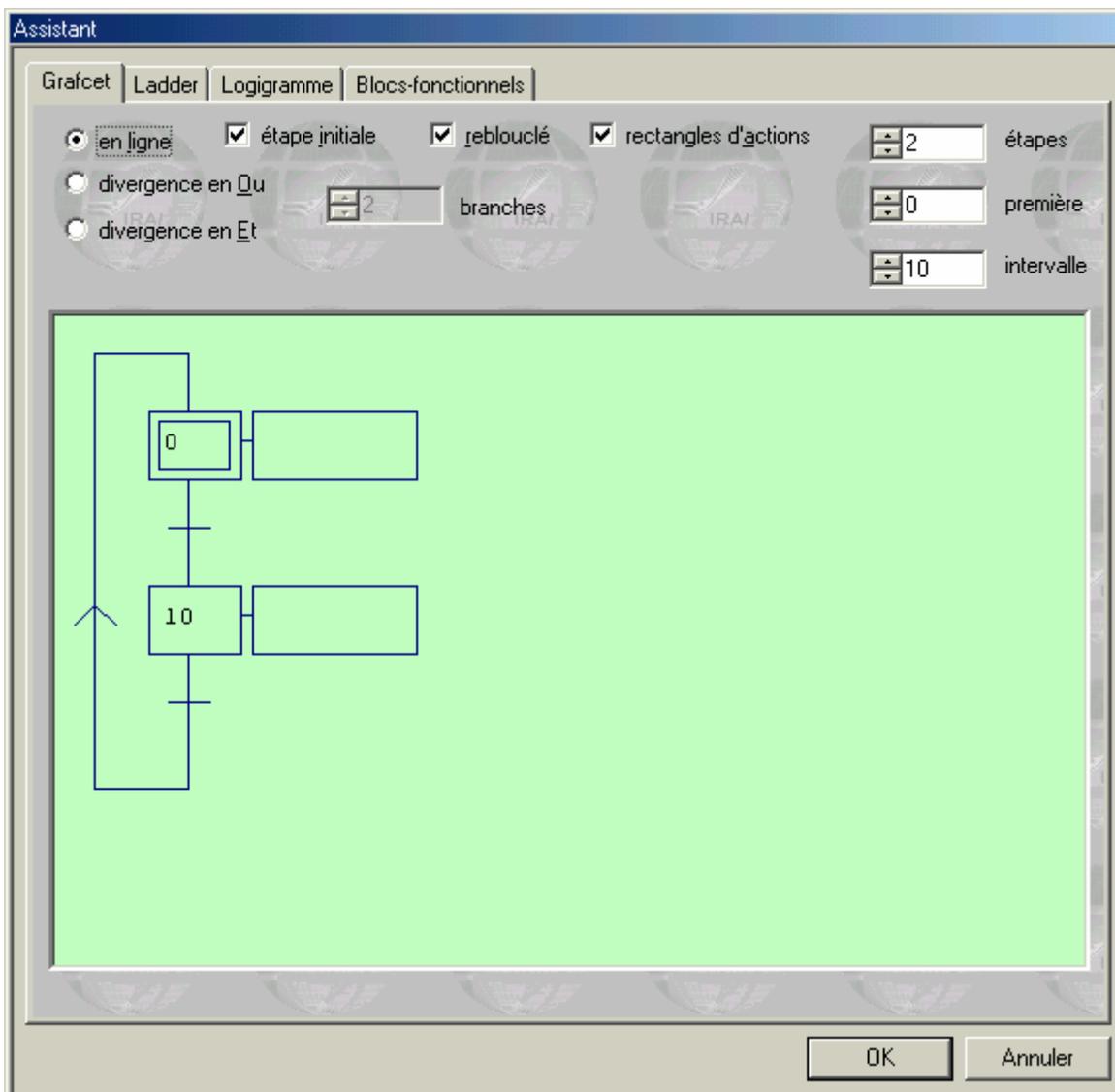
Le navigateur permet de voir et de manipuler l'ensemble des éléments d'un projet. En double cliquant sur les éléments ou en cliquant avec le bouton droit de la souris, on accède aux différentes fonctions applicables à chaque éléments.

Dessiner des programmes

Pour dessiner des programmes, plusieurs outils sont à votre disposition :

Dessiner avec l'assistant

C'est sans doute le plus simple lorsqu'on débute avec AUTOMGEN. Cliquez avec le bouton droit de la souris sur un folio ouvert dans l'espace de travail et choisissez « Assistant » dans le menu. Laissez vous ensuite guider dans les choix. Lorsque vous avez fini, cliquez sur « OK » et poser le dessin sur le folio en cliquant avec le bouton gauche de la souris.



L'assistant.

Dessiner avec le menu contextuel

En cliquant avec le bouton droit de la souris sur un folio ouvert dans l'espace de travail, le menu vous propose une série d'éléments que vous pouvez poser sur le folio. C'est un mode de création instinctif et rapide.

Dessiner avec la palette

En sélectionnant des éléments dans la palette, vous pouvez créer rapidement des programmes à partir d'éléments déjà créés.

Enrichir et personnaliser la palette

La définition de la palette est composée de fichiers « .GR7 » se trouvant dans le répertoire « <répertoire d'installation d'AUTOMGEN>\pal ». Vous pouvez effacer, modifier, renommer ou ajouter des fichiers. Pour générer des fichiers « .GR7 », utiliser la commande « Exporter » en cliquant avec le bouton droit de la souris sur un folio dans le navigateur. Les noms affichés dans la palette sont ceux des fichiers « .GR7 ». Relancer AUTOMGEN pour que le nouvel élément apparaisse dans la palette.

Dessiner avec les touches du clavier

Chaque touche est associée à un des blocs de dessin. L'élément « Blocs » de la palette donne également accès à l'ensemble de ces blocs. Le tableau ci-dessous liste les blocs et leur utilisation.

Bloc d'effacement

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[A]	Effacement	Permet de remettre à blanc une cellule	Tous

Blocs de liaison

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[E]	Liaison verticale	Liaison haut vers bas ou bas vers haut	Tous
—	[F]	Liaison horizontale	Liaison droite vers gauche ou gauche vers droite	Tous

	[G]	Coin supérieur gauche	Liaison bas vers droite ou gauche vers bas	Tous
	[H]	Coin supérieur droit	Liaison bas vers gauche ou droite vers bas	Tous
	[I]	Coin inférieur gauche	Liaison haut vers droite ou gauche vers haut	Tous
	[J]	Coin inférieur droit	Liaison haut vers gauche ou droite vers haut	Tous
	[Z]	Croisement	Croisement de deux liaisons	Tous

Blocs Grafcet

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[B]	Étape	Étape normale	Grafcet
	[C]	Étape initiale sans activation	Étape initiale sans activation	Grafcet
	[D]	Étape initiale	Étape initiale	Grafcet
		Macro-étape	Accessible dans le menu contextuel uniquement	Grafcet
	[T]	Transition	Transition	Grafcet
	[K]	Bord gauche d'une divergence en « Et »	Obligatoirement à gauche des divergences en « Et »	Grafcet
	[L]	Branche supplémentaire d'une divergence en « Et » ou convergence en « Et »	Ne pas utiliser comme bord gauche ou droit d'une divergence en « Et »	Grafcet

	[M]	Bord droit d'une divergence en « Et »	Obligatoirement à droite d'une divergence en « Et »	Grafcet
	[N]	Prolongation d'une divergence en « Et »	Se place entre les blocs [K], [L], [M], [P] ou [O],[P],[Q], [L]	Grafcet
	[O]	Bord gauche d'une convergence en « Et »	Obligatoirement à gauche d'une convergence en « Et »	Grafcet
	[P]	Branche supplémentaire d'une convergence en « Et » ou divergence en « Et »	Ne pas utiliser comme bord gauche ou droit d'une convergence en « Et »	Grafcet
	[Q]	Bord droit d'une convergence en « Et »	Obligatoirement à droite d'une convergence en « Et »	Grafcet
	[R]	Divergence en « Ou »	Ne pas utiliser comme bord d'une convergence en « Ou »	Grafcet
	[S]	Convergence en « Ou »	Ne pas utiliser comme bord d'une divergence en « Ou »	Grafcet
	[U]	Saut ou reprise d'étape à gauche	Convergence ou divergence en « Ou »	Grafcet
	[V]	Saut ou reprise d'étape à droite	Convergence ou divergence en « Ou »	Grafcet
	[ESPACE] sur un bloc [E]	Liaison orientée vers le haut	Pour les rebouclages et les reprises d'étapes	Grafcet

Blocs Logigrammes

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[0] (zéro)	Affectation logigramme	Sépare la zone « test » de la zone « action »	Logigrammes

	[1]	Fonction « Pas »	Complémente le signal d'entrée du bloc	Logigrammes
	[2]	Fonction « Et »	Combine les entrées en un « Et » logique	Logigrammes
	[3]	Fonction « Ou »	Combine les entrées en un « Ou » logique	Logigrammes
	[4]	Milieu de bloc	Agrandit un bloc fonction « Et » ou « Ou »	Logigrammes
	[5]	Bas de bloc	Termine un bloc fonction « Et » ou « Ou »	Logigrammes

Blocs Ladder

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[C]	Partie gauche bobine	Débute une action	Ladder
	[D]	Partie droite bobine	Termine une action	Ladder
	[U]	Bord gauche	Termine le schéma	Ladder
	[V]	Bord droit	Débute le schéma	Ladder
	[R]	Connexion	Fonction « Ou »	Ladder
	[S]	Connexion	Fonction « Ou »	Ladder

Blocs Action

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[W]	Bord gauche rectangle d'action	Débute une action	Grafcet et Logigrammes

	[X]	Milieu d'un rectangle d'action	Prolonge une action	Grafcet et Logigrammes
	[Y]	Bord droit d'un rectangle d'action	Termine une action	Grafcet et Logigrammes
	[S]	Divergence Action	Permet de juxtaposer des rectangles d'action verticalement	Grafcet et Logigrammes
	[V]	Divergence Action	Permet de juxtaposer des rectangles d'action verticalement	Grafcet et Logigrammes

Blocs Test

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[7]	Bord gauche d'un test	Début un test	Logigrammes et Ladder
	[6]	Bord droit d'un test	Termine un test	Logigrammes et Ladder

Blocs Organigramme

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[<]	Entrée d'organigramme	Marque l'entrée dans un rectangle	Organigrammes
	[=]	Sortie « Faux »	Sortie si faux d'un rectangle de test	Organigrammes

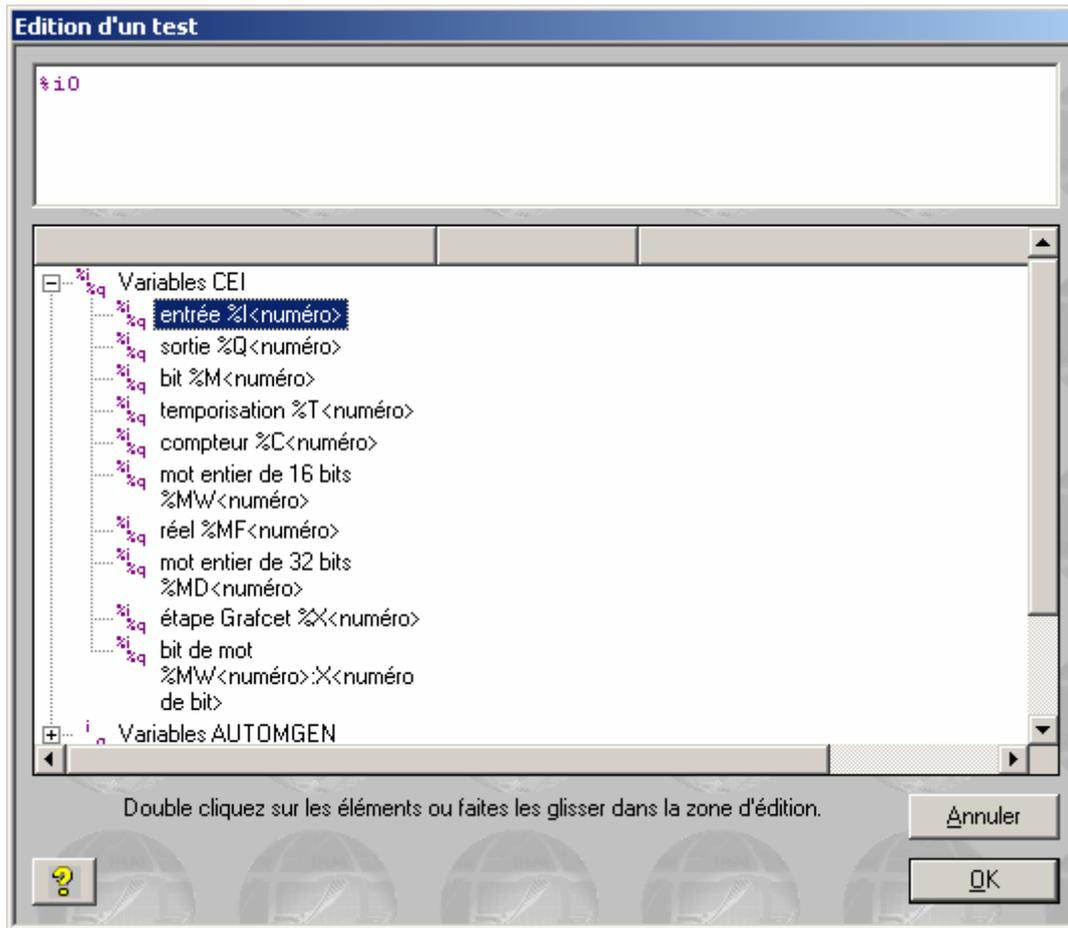
Blocs Bloc fonctionnel

Aspect	Touche associée	Nom générique	Commentaires	Langages
	[8]	Coin supérieur gauche d'un bloc fonctionnel	Début le nom du bloc fonctionnel	Bloc fonctionnel
	[9]	Coin supérieur droit d'un bloc fonctionnel	Termine le nom du bloc fonctionnel	Bloc fonctionnel
	[:]	Coin inférieur gauche d'un bloc fonctionnel	Ajoute une entrée au bloc fonctionnel	Bloc fonctionnel
	[;]	Bord gauche d'un bloc fonctionnel	Ajoute une entrée au bloc fonctionnel	Bloc fonctionnel
	[>]	Bord droit d'un bloc fonctionnel	Ajoute une sortie au bloc fonctionnel	Bloc fonctionnel
	[?]	Coin inférieur droit d'un bloc fonctionnel	Ajoute une sortie au bloc fonctionnel	Bloc fonctionnel

Documenter les éléments de programme

Pour documenter les éléments de programme, cliquez dessus avec le bouton gauche de la souris. Pour créer des commentaires, cliquez sur un espace vide du folio. Pour valider les modifications, appuyez sur la touche [Entrée] ou cliquez en dehors de la zone d'édition avec le bouton gauche de la souris. Pour annuler les modifications, appuyez sur la touche [Echap] ou cliquez avec le bouton droit de la souris en dehors de la zone d'édition.

En édition des tests et des actions, un bouton « ... » apparaît sous la zone d'édition. En cliquant dessus, vous accédez à un assistant de création de tests ou d'actions.



L'assistant de création des tests.

Ajouter des symboles

Pour créer un symbole, cliquez avec le bouton droit de la souris sur la table des symboles dans l'espace de travail et choisissez « Ajouter ». Ou cliquez sur le bouton  dans la barre d'outils. Vous pouvez également lancer la compilation d'un programme contenant des symboles non définis. Les variables correspondantes aux symboles vous seront alors demandées pendant la compilation.

Propriétés d'un symbole

Nom

Variable associée

Commentaires associés



Le nom peut contenir n'importe quel caractère à l'exception de '.' La longueur est limitée à 512 caractères. Le nom de la variable doit respecter la syntaxe CEI-1131-3 ou AUTOMGEN.

Attribution des symboles pendant la compilation.



Pour dessiner facilement un programme, créez un nouveau folio, puis cliquez avec le bouton droit de la souris sur le fond du folio. Choisissez « Assistant » dans le menu puis laissez vous guider par l'assistant.

Exécuter une application

Exécuter facilement une application

Le bouton  de la barre d'outils, représente la méthode la plus rapide pour observer le résultat de l'exécution d'une application. Ce bouton poussoir active les mécanismes suivants :

- compilation de l'application si elle n'est pas à jour (pas déjà compilée depuis les dernières modifications),
- installation du module d'exécution (avec téléchargement si la cible courante est un automate et suivant les options de connexions),
- passage de la cible en RUN,
- activation de la visualisation dynamique.

Mettre fin à l'exécution

Cliquez sur . Sur cible automate, le programme continu à s'exécuter sur la cible. Sur exécuteur PC, le programme est stoppé.

Uniquement compiler

Cliquez sur .

Stopper la compilation

Cliquez sur .

Se connecter à un automate ou installer l'exécuteur PC

Cliquez sur .

Se déconnecter d'un automate ou désinstaller l'exécuteur PC

Cliquez sur .

Mettre la cible en mode RUN

Cliquez sur .

Mettre la cible en mode STOP

Cliquez sur .

Initialiser la cible

Cliquez sur .

Faire un cycle programme sur la cible (généralement non supporté sur les automates)

Cliquez sur .

Activer la visualisation dynamique

Cliquez sur .



Pour exécuter une application, cliquez sur le bouton « GO ». Pour mettre fin à l'exécution, cliquez de nouveau sur le même bouton.

Le compilateur

Le compilateur traduit les folios en un ensemble d'équations de langage pivot (visualisables en double cliquant sur l'élément « Code généré / langage pivot » dans le navigateur).

Le langage pivot est ensuite traduit en langage exécutable par un post-processeur (le post-processeur courant est visible et sélectionnable par un double clic dans le volet « Cibles » accessible en cliquant sur l'onglet « Cibles » en bas de la fenêtre où se trouve le navigateur).

Modifier les options du compilateur

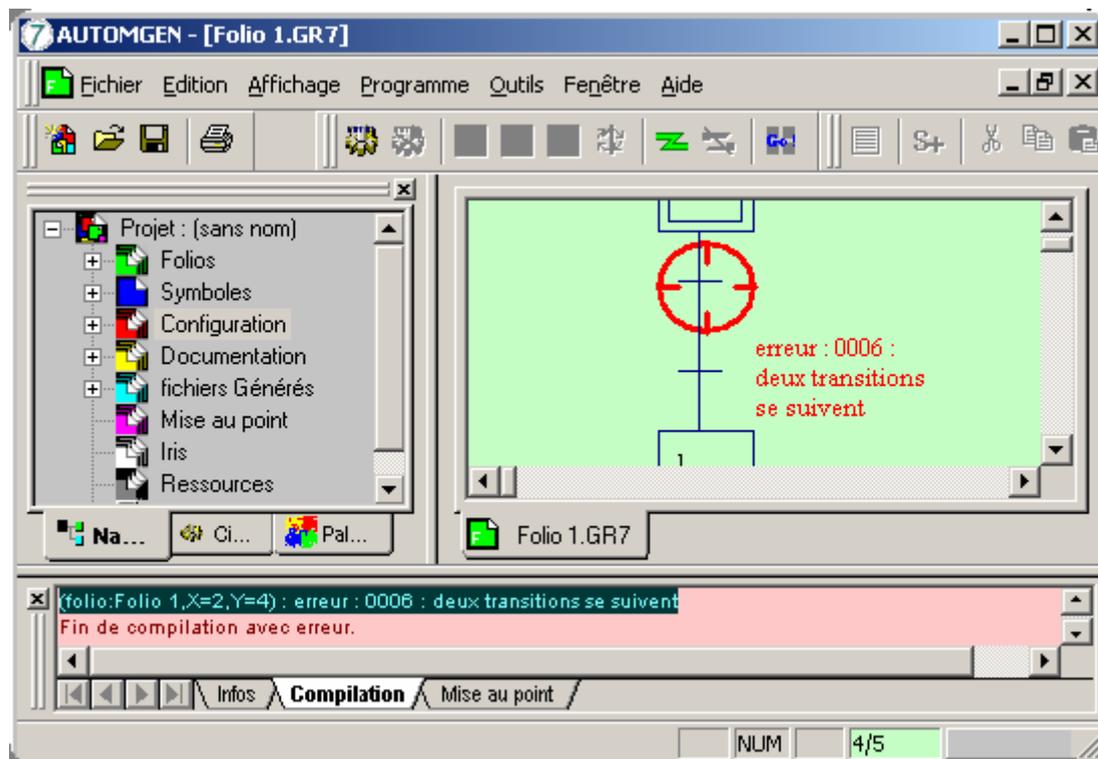
Double cliquez sur l'élément « Configuration / Options du compilateur ».

Visualiser les messages de compilation

Le volet « Compilation » de la fenêtre des messages contient les comptes rendus de la dernière compilation.

Localiser une erreur

En double cliquant sur messages d'erreurs, vous pouvez en localiser la source.



Un message d'erreur et sa source.

Si les fenêtres de messages sont cachées et qu'une ou plusieurs erreurs sont détectées par le compilateur, alors une boîte de dialogue annonce la première erreur détectée (pour faire apparaître les fenêtres de messages : utilisez la commande « Messages » du menu « Fenêtres »).



A la fin de la compilation, la fenêtre « Compilation » donne la liste des éventuelles erreurs. En double cliquant sur les messages d'erreur, l'emplacement du programme qui a provoqué l'erreur est affiché.

Exécution des programmes sur PC

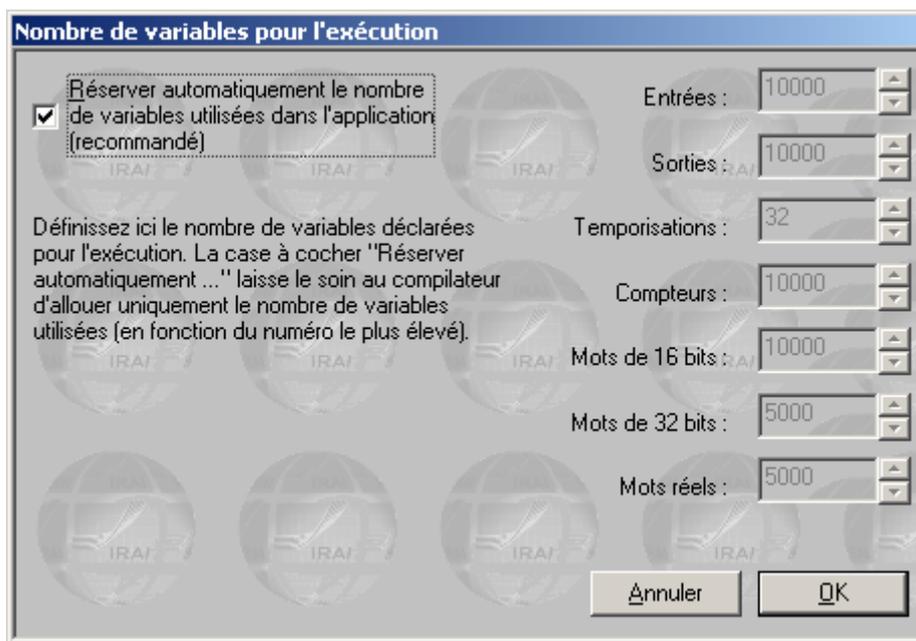
La cible « Exécuteur PC » est un véritable automate embarqué dans votre PC.

Vous pouvez :

- tester vos applications,
- piloter une partie opérative virtuelle réalisée avec IRIS 2D ou 3D,
- piloter des cartes d'entrées / sorties connectées au PC.

Configurer le nombre de variables

Double cliquez sur l'élément « Configuration / Post-processeurs / Exécuteur PC / Variables ».



Sélection du nombre de variables

Par défaut l'espace nécessaire pour les variables utilisées dans l'application est automatiquement réservé. Vous pouvez sélectionner manuellement la quantité de mémoire réservée pour chaque type de variables. Ceci peut être nécessaire si un adressage indexé est utilisé pour accéder à une table de variable.

Variables système de l'exécuteur PC

Les bits 0 à 99 et les mots 0 à 199 sont des variables système à ne pas utiliser comme variable utilisateur dans vos applications. Les deux tableaux ci-dessous donnent le détail des variables systèmes de l'exécuteur PC.

Bits	Utilisation
0	actif au premier cycle, activation des étapes Grafcet initiales
1 à 4	réservés aux drivers d'E/S
5 à 7	réservés aux erreurs pour les drivers d'E/S
8	erreur sur débordement de chien de garde si égal à 1
9 et 10	erreur : défaillance générale de l'exécuteur
11	mode de marche : 1=RUN, 0=STOP
12	arrêt d'urgence : passe à 1 en cas d'erreur ou peut être forcé à 1 pour bloquer le programme
13 à 29	réservés aux drivers
30	bit associé au timer 1
31	bit associé au timer 2
32	bit associé au timer 3
33	bit associé au timer 4
34	bit associé au timer 5
35	bit de reprise secteur (passe à 1 sur reprise secteur, la r.a.z. est à la charge du programmeur)
36	la mise à 1 de ce bit provoque une lecture de l'horloge temps réel et un transfert dans les mots Système 4, 5, 6, 7, 8, 51 et 52
37	la mise à 1 de ce bit provoque l'écriture des mots Système 4, 5, 6, 7, 8, 51 et 52 dans l'horloge temps réel
38 à 55	réservés
56	division par zéro
57 à 67	réservés pour les versions futures
68 à 99	réservés pour la pile des traitements booléens

Mots	Utilisation
0	réservé pour la partie haute du résultat de la multiplication ou pour le reste de la division
1 à 3	timers en millisecondes
4	timer en 1/10 seconde
5	timer en secondes
6	timer en minutes
7	timer en heures
8	timer en jours

9 à 29	réservés aux drivers E/S
30	compteur du timer 1
31	compteur du timer 2
32	compteur du timer 3
33	compteur du timer 4
34	compteur du timer 5
35	consigne du timer 1
36	consigne du timer 2
37	consigne du timer 3
38	consigne du timer 4
39	consigne du timer 5
40	partie basse de la référence horloge
41	partie haute de la référence horloge
42 à	réservés aux drivers E/S
50	
51	timer en mois
52	timer en années

Modifier la période d'exécution

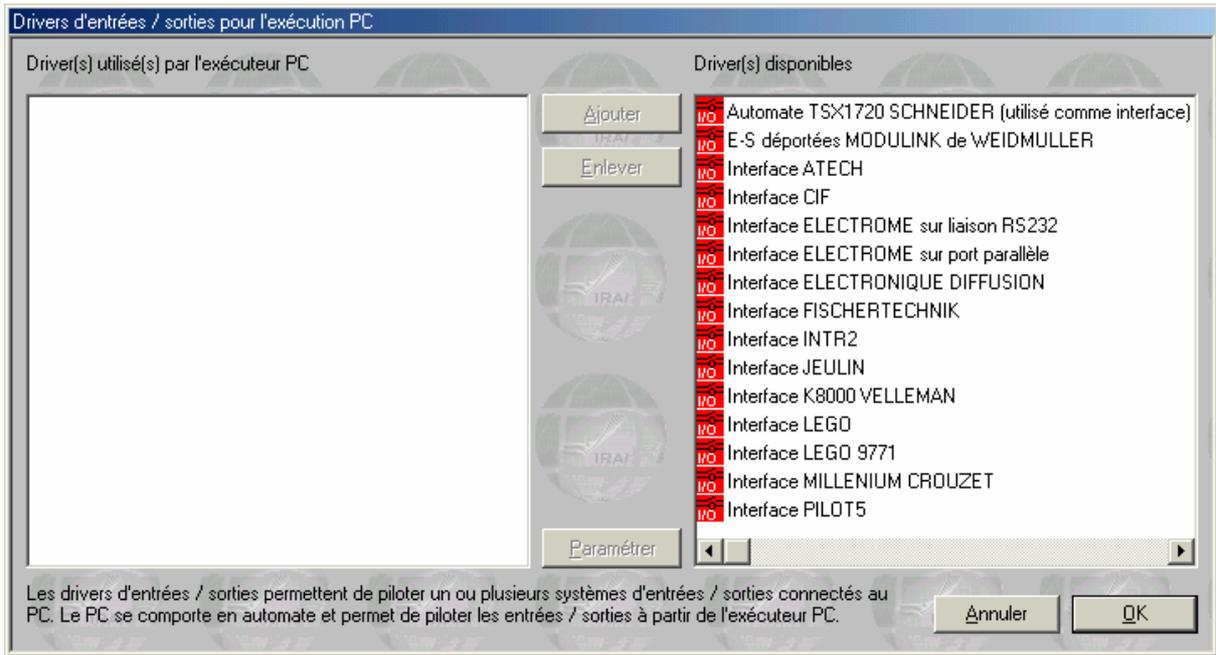
Double cliquez sur « Post-processeur / Exécuteur PC / Exécution ».



Réglage de la période d'exécution.

Piloter des entrées / sorties

Double cliquez sur « Configuration / Post-processeur / Exécuteur PC / Drivers d'E/S ».



Ajout d'un driver d'E/S

Sélectionnez un driver dans la liste de droite, puis cliquez sur « Ajouter ».

« Paramétrer » permet de configurer certains drivers.



L'exécuteur PC transforme votre PC en automate programme, il permet de piloter des entrées / sorties connectées directement à votre ordinateur.

Références IRIS 2D

Les objets IRIS 2D vous permettent de créer des applications de supervision et de simulation de parties opératives en 2D.

Le lien entre les objets et l'application d'automatisme est toujours réalisé par échange de l'état des variables.

Les objets IRIS 2D sont encapsulés dans des fenêtres WINDOWS.



Un objet IRIS 2D

Les objets IRIS 2D ont deux états possibles : le mode « Configuration » (on peut modifier les caractéristiques de l'objet) et le mode « Utilisateur » (on utilise l'objet). Le mode « Utilisateur » est également appelé mode « Exploitation ».

Modifier la visibilité des objets

Les objets peuvent être cachés ou visibles. Cette propriété peut être spécifiée à l'ouverture de l'objet ou sur changement d'état de la visualisation dynamique dans l'environnement. Seuls les objets de plus haut niveau (pas les objets posés sur un pupitre) peuvent être montrés ou cachés. Les objets posés sur un pupitre sont montrés ou cachés en même temps que le pupitre.

Pour modifier en dynamique la visibilité des objets, cliquez avec le bouton gauche de la souris sur les objets dans le navigateur et choisissez « Montrer / Cacher ».

Pour modifier les propriétés de visibilité, cliquez avec le bouton gauche de la souris sur les objets dans le navigateur et choisissez « Propriétés ».



Les propriétés de visibilité d'un objet.

Modifier les caractéristiques d'un objet

Effacer un objet

Méthode 1 : cliquez sur le bouton  qui se trouve sur la surface de l'objet.

Méthode 2 : cliquez avec le bouton droit de la souris sur l'objet dans le navigateur et choisissez « Supprimer » dans le menu.

Dimensionner un objet

En saisissant l'objet par un de ses bords, vous pouvez l'étirer ou le rétrécir (on peut modifier précisément la taille d'un objet en accédant à ses propriétés, voir plus loin).

Déplacer un objet

Saisissez l'objet en cliquant avec le bouton gauche de la souris sur la mini barre se trouvant en haut de sa surface.

Passer un objet en mode « Utilisateur »

Méthode 1 : cliquez sur le bouton  qui se trouve sur l'objet avec le bouton gauche de la souris.

Méthode 2 : cliquez avec le bouton droit de la souris sur l'objet.

Passer un objet en mode « Configuration »

Cliquez avec le bouton droit de la souris sur l'objet.

Modifier les caractéristiques d'un objet

Méthode 1 : cliquez sur le bouton .

Méthode 2 : enfoncez la touche [CTRL] du clavier et cliquez avec le bouton droit de la souris sur l'objet, relâchez ensuite la touche [CTRL].

Méthode 3 : cliquez avec le bouton droit de la souris sur l'objet dans le navigateur et choisissez « Propriétés » dans le menu.

Verrouillez l'accès à la configuration de tous les objets

Cliquez avec le bouton droit de la souris sur « Projet » dans le navigateur, choisir « Propriétés » et cochez la case « Interdire la configuration des objets IRIS 2D » dans l'onglet « Avancé ».

Objets de base, objets prédéfinis

Les objets de bases définissent des grands types de fonctionnalités. Les objets prédéfinis s'appuient sur un type de base et une configuration pour répondre à un besoin spécifique. Par exemple, un bouton poussoir d'arrêt d'urgence est un objet dérivé de l'objet de base permettant de réaliser des boutons poussoirs et des voyants. Pour accéder aux objets prédéfinis, utiliser l'assistant en cliquant avec le bouton droit de la souris sur l'élément « IRIS » dans le navigateur et en choisissant « Ajouter un objet IRIS 2D ».

Liste des objets de base

L'objet « Pupitre »

L'objet pupitre est le seul qui peut contenir d'autres objets sur sa surface. Vous l'utiliserez pour créer pupitres de commandes et surfaces d'animation pour parties opératives virtuelles. Cet objet possède un bouton poussoir  qui permet de gérer les objets sur sa surface : ajout, déplacement, suppression, etc ...

L'objet « Bouton et voyant »

Vous l'utiliserez pour créer boutons poussoirs et voyants en interactions avec des variables de l'application d'automatisme.

L'objet « Objet »

C'est un élément polymorphique que vous utiliserez principalement pour simuler des parties opératives.

L'objet « Valeur digitale »

Vous l'utiliserez pour afficher des valeurs numériques de l'application d'automatisme sous forme de nombres.

L'objet « Ecran, clavier, liste de messages »

Vous l'utiliserez pour afficher des informations de l'application d'automatisme sous formes de textes.

L'objet « Son »

Vous l'utiliserez pour produire des sorties sonores sur changement de l'état des variables de l'application d'automatisme.

L'objet « Archivage de données »

Vous l'utiliserez pour afficher sous forme de tables ou de courbes et sauvegarder dans le mémoire de l'ordinateur ou sur le disque des données de l'application d'automatisme.

L'objet « Programme »

Vous l'utiliserez pour effectuer des traitements qui s'exécuteront indépendamment de l'application d'automatisme.

L'objet « Boîte de dialogue »

Vous l'utiliserez pour afficher des messages sous forme de fenêtres surgissant sur changement d'état de variables de l'application d'automatisme.

L'objet « Valeur analogique »

Vous l'utiliserez pour afficher sous forme d'éléments analogiques (barres, cadrans, etc ...) des variables numériques de l'application d'automatisme.

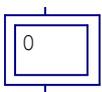
Prise en main

Ce chapitre va vous permettre de créer très rapidement votre première application IRIS 2D. Nous allons créer un pupitre, y déposer un bouton poussoir et lier l'objet aux variables de l'application d'automatisme.

Etape 1

Créons une application minimale avec AUTOMGEN voir chapitre Dessiner des programmes.

Il s'agit d'un Grafcet comportant une étape tel que montré ci-après.



Etape 2

Lancez l'exécution de l'application AUTOMGEN (cliquez sur le bouton « Go » dans la barre d'outils).

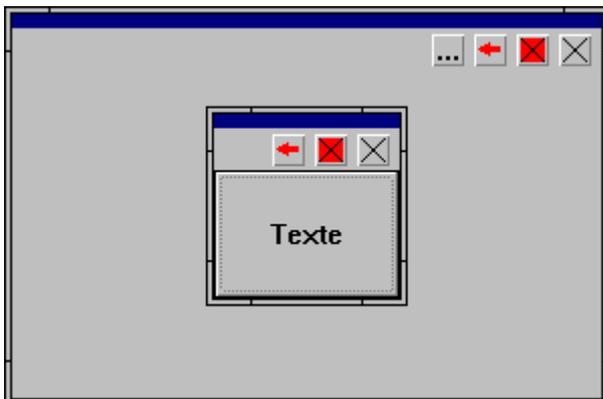
Etape 3

Cliquez avec le bouton droit de la souris dans le navigateur sur l'élément « IRIS » puis choisissez « Ajoutez un objet IRIS 2D » dans le menu. Dans la catégorie « Objets de base », double cliquez sur « Pupitre ». A ce stade l'objet doit apparaître à l'écran sous cette forme :



Etape 4

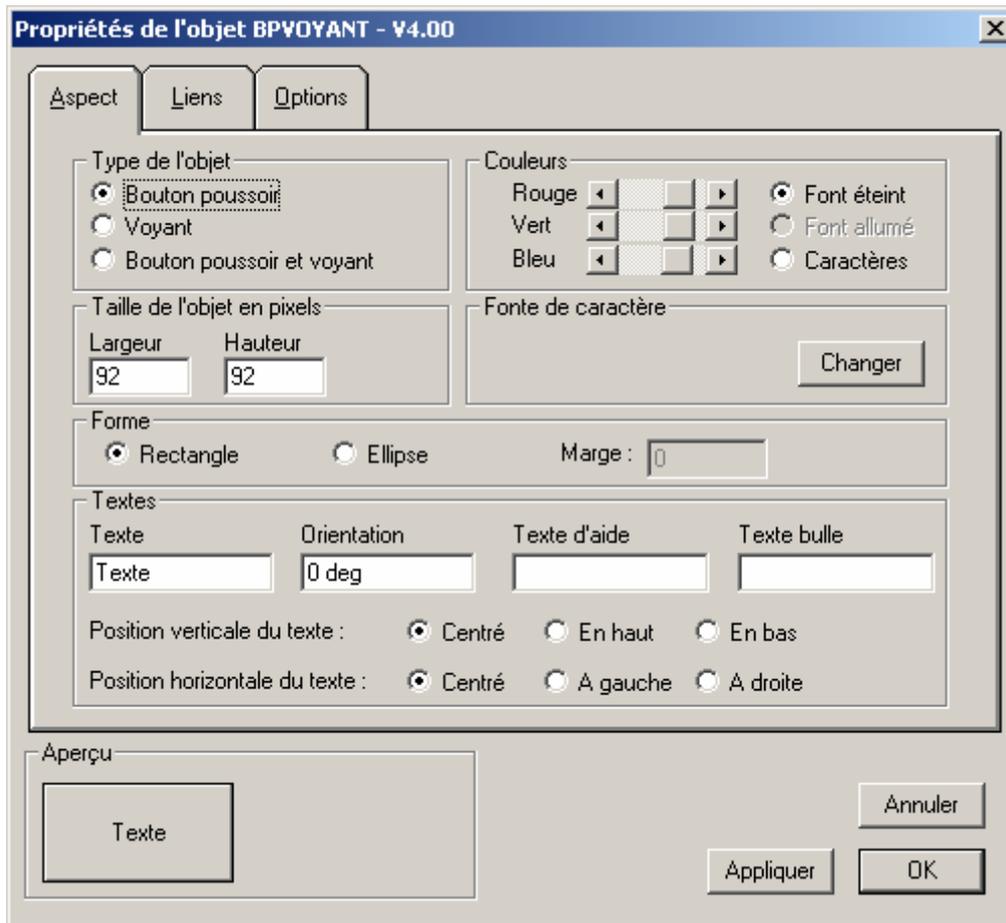
Pour ajouter un bouton poussoir sur le pupitre vous devez maintenant cliquer sur l'icône  du pupitre (accès au menu) et sélectionner l'option « Ajouter un objet ». Dans la catégorie « Objets de base », double cliquez sur « Bouton voyant ». L'objet doit alors apparaître sur le pupitre :



Etape 5

Nous allons associer le bouton poussoir à une sortie de l'application d'automatisme, %Q4 par exemple. Cliquez sur l'icône  du bouton

poussoir (pas du pupitre). La boîte de dialogue des propriétés du bouton poussoir s'ouvre :



Cliquez sur l'onglet « Liens » (en haut de la fenêtre de dialogue). Sous la rubrique « Action lorsque le bouton est enfoncé » entrez « %Q4=1 ». Sous la rubrique « Action lorsque le bouton est relâché » entrez « %Q4=0 ». Cliquez ensuite sur le bouton poussoir « OK » en bas de la fenêtre de dialogue. Les actions sur le bouton poussoir ont pour effet de piloter la sortie 4 de l'application d'automatisme. Vous pouvez ouvrir une fenêtre « Monitoring » dans le menu « Mise au point » en cliquant avec le bouton droit de la souris dans le navigateur. Visualisez l'état de la sortie 4 pendant que vous cliquez puis relâchez le bouton poussoir.

Etape 6

Nous allons associer un voyant à l'objet « Bouton Voyant », ce voyant sera associé à une entrée de l'application d'automatisme (I2 par exemple). Cliquez à nouveau sur l'icône  du bouton poussoir. Dans l'onglet « Aspect » cliquez sur le bouton radio « Bouton poussoir et voyant ». Cliquez sur l'onglet « Liens » et entrez « %i2 » sous la rubrique « Etat du voyant ». Cliquez sur le bouton poussoir « OK » en bas de la

fenêtre de dialogue des propriétés. Vous pouvez maintenant modifier l'état de la variable « %i2 » (avec une fenêtre « Monitoring » ou en modifiant l'état de l'entrée physique si elle existe).

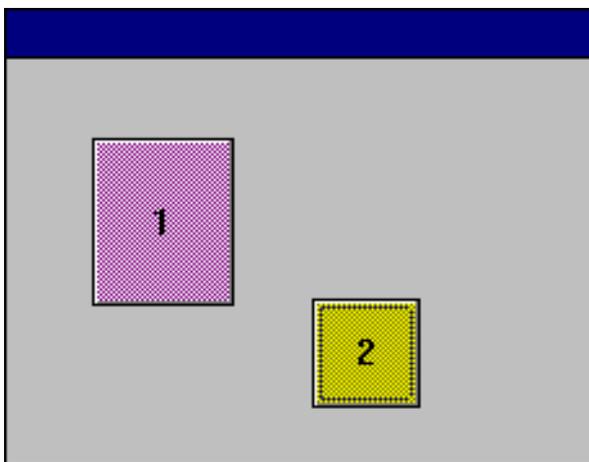
Etape 7

Nous allons dupliquer l'objet « Bouton Voyant » présent sur le pupitre pour obtenir un deuxième bouton poussoir dont nous modifierons les propriétés. Cliquez sur le bouton poussoir avec le bouton gauche de la souris en maintenant la touche [SHIFT] enfoncée. Des carrés noirs apparaissent autour de l'objet sélectionné. Cliquez sur l'icône  du pupitre et choisissez l'option « Copier ». Cliquez sur l'icône  du pupitre et choisissez l'option « Coller ». Il y a maintenant deux objets « Bouton Voyant » superposés. Saisissez celui du dessus (c'est le seul accessible) par sa barre supérieure et déplacez le en dehors de l'autre bouton poussoir. L'objet ainsi dupliqué possède les mêmes propriétés que le premier. Vous pouvez maintenant paramétrer le deuxième objet pour qu'il soit par exemple lié à la sortie 5 et l'entrée 3.

Vous pouvez également personnaliser l'aspect des boutons poussoirs en utilisant l'onglet aspect pour les deux objets. Vous pouvez aussi modifier la taille des objets en les saisissant par un de leurs bords.

Les trois objets présents à l'écran (le pupitre et les deux boutons poussoirs) sont en mode « Configuration », c'est à dire qu'ils laissent apparaître une mini barre en haut de leur surface, des icônes et une bordure permettant de les dimensionner. Les objets ont un autre mode nommé « Exploitation » dans lequel ils ont un aspect définitif : plus de barre en haut, plus d'icône et plus de bordure pour les dimensionner. Pour basculer un objet d'un mode à l'autre, il suffit de cliquer dessus avec le bouton droit de la souris.

A ce stade vous devez avoir créé un objet qui ressemble à ceci :



Réaliser une application de supervision autonome

Pour réaliser une application de supervision autonome (sans développer l'application d'automatisme avec AUTOMGEN), suivez la démarche suivante :

- créez les correspondances de variables entre AUTOMGEN et les variables de l'automate en double cliquant sur l'élément « Configuration / Post-processeur / <nom du post-processeur> / Correspondance des variables » (reportez-vous au manuel de référence des post-processeurs pour plus de détails),
- compilez l'application en cliquant sur le bouton  de la barre d'outils (ceci a pour effet de valider la correspondance des variables),
- configurez le mode de connexion sur « Seulement connecter » en double cliquant sur l'élément « Configuration / Post-processeur / <nom du post-processeur> / Option de connexion ».

Remarques :

- l'option « Go automatique » du projet vous permet d'obtenir une application qui se connecte automatiquement à la cible au démarrage.
- la commande « Générer un exécutable » du menu « Fichier » vous permet d'obtenir une application de supervision auto compactée et libre de droit sous la forme d'un seul fichier exécutable.

Syntaxe pour l'accès à l'état des variables

Vous pouvez utiliser les noms de variables en syntaxe AUTOMGEN, CEI 1131-3 ou un symbole. Les boutons poussoirs « ... » se trouvant à proximité des zones de saisies dans les objets permettent d'accéder à l'assistant de sélection d'un nom de variable.

Etat booléen

Cette syntaxe est à utiliser dans les rubriques « états » des objets.

Pour tester l'état d'une variable booléenne, utilisez le nom de la variable, par exemple : « i0 », « %q0 », « vanne ouverte ».

Pour tester l'état complémenté d'une variable booléenne, ajouté un caractère « / » avant le nom de la variable, par exemple : « /i4 », « /%M100 », « /niveau haut ».

Pour tester l'égalité d'une variable numérique avec une constante, utilisez le nom de la variable numérique, suivi de « = », « < », « > » et d'une constante, par exemple : « %MW200=4 », « vitesse>2 ».

L'état complémenté permet de réaliser les tests « si différent », « si inférieur ou égal » et « si supérieur ou égal », par exemple : « /%MW201<300 ».

L'opérateur '&' permet de tester un bit d'une variable numérique, par exemple M200&4 teste le troisième bit (4 = 2 puissance 3) du mot m200.

Etat numérique

Cette syntaxe est à utiliser dans les rubriques « états » des objets.

Pour lire l'état d'une variable numérique, utilisez le nom de la variable, par exemple : « %MW300 », « m400 », « pression », « _+V_ ».

Modification d'état

Cette syntaxe est à utiliser dans les rubriques « ordre » des objets.

Pour réaliser la modification de l'état d'une variable, ajoutez à la suite du nom de la variable le signe « = » suivi d'une constante.

Pour les variables booléennes, les constantes suivantes sont utilisables : « 0 », « 1 », « F1 » (forçage à 1), « F0 » (forçage à 0), « UF » (fin de forçage), par exemple : « %Q0=1 », « %I10=F1 », « %I12=UF ».

Pour les variables numériques, la constante est un nombre, par exemple : « M200=1234 », « vitesse=4 ».

Ordres spéciaux

Les mots clés suivants peuvent être utilisés dans les rubriques ordres des objets :

« RUN » : passe la cible en mode RUN,

« STOP » : passe la cible en stop,

« INIT » : initialise la cible,

« STEP » : effectue un pas sur la cible,

« GO » : identique à la commande GO de l'environnement,

« ENDGO » : termine la commande GO,

« EXIT » : quitte l'environnement,

« UCEXIT » : quitte l'environnement sans demander de confirmation,

« OPENAOF(<objet>) » : montre un objet. « <objet> » désigne un objet soit par le titre de l'objet soit par son numéro d'identificateur (configuré dans les propriétés des objets) avec la syntaxe « #identificateur ».

« CHAINAOF(<objet>) » : montre un objet et cache l'objet courant. « <objet> » désigne un objet soit par le titre de l'objet soit par son numéro d'identificateur (configuré dans les propriétés des objets) avec la syntaxe « #identificateur ».

Echanges entre objets

« PARENTPARAM(paramètre {+n} {-n}) »

Permet à un objet enfant d'accéder à un paramètre du pupitre parent. Le paramètre doit être défini dans la rubrique « Liens / Données pour les objets enfants » du pupitre parent. Voir chapitre L'objet « Pupitre »

SISTERPARAM(identificateur , paramètre)

Utilisé par l'objet OBJET, cette syntaxe permet de lire une valeur propre à un objet. Voir l'objet « Objet ».

SETPARAM(identificateur , paramètre , valeur)

Permet de modifier le paramètre d'un objet.

Pour accéder à la liste des paramètres modifiables, cliquez sur le bouton droit de la souris pendant l'édition des zones actions d'un objet « Bouton Voyant », puis choisissez la commande « Paramètres ».

Détail de l'objet « Pupitre »

Onglet « Aspect »

Fenêtre

Permet de définir l'aspect de la fenêtre composant le pupitre : présence d'une bordure, d'une barre de titre (dans ce cas un titre peut être précisé), présence des icônes de fermeture et de réduction. La case à

cocher « Affichage des messages d'aides » permet de définir une zone de message en bas de la fenêtre, la taille de cette zone est déterminée automatiquement en fonction de la fonte choisie (voir ci-après). Si une telle zone n'est pas définie, alors les messages provenant des enfants seront affichés soit sur le pupitre parent du pupitre, soit sur en bas de la fenêtre de l'environnement AUTOMGEN (si l'objet n'a pas de parent).

Fond du pupitre

Détermine l'aspect du fond du pupitre : coloré (voir ci-après), transparent (accessible seulement si le pupitre est l'enfant d'un autre pupitre), bitmap (le fond est défini par un fichier « .BMP » créé avec PAINTBRUSH par exemple).

Couleurs

Permet de choisir la couleur pour le fond du pupitre (si un fond coloré est sélectionné - voir plus haut), le fond et les caractères de la zone d'affichage des messages d'aides (si cette zone est validée - voir plus haut).

Fonte de caractères pour l'aide

Détermine la fonte utilisée pour afficher les messages d'aide en bas du pupitre.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Textes

Le texte d'aide et le texte bulle.

Onglet « Bitmap »

Bitmap

Si le fond du pupitre contient un bitmap (voir onglet « Aspect ») la zone d'édition doit contenir un nom d'accès complet vers un fichier « .BMP » (les formats 16 couleurs, 256 couleurs et 24 bits sont supportés).

Les boutons poussoirs « PARCOURIR » et « EDITER » permettent respectivement de rechercher un fichier « .BMP » et d'éditer un fichier avec le logiciel PAINTBRUSH de WINDOWS.

Onglet « Liens »

Données pour les objets enfants

Cette zone d'édition permet de définir des paramètres auxquels pourront accéder les objets enfants avec le mot clé « PARENTPARAM ». Il faut écrire une définition par ligne. Chaque définition doit respecter la syntaxe suivante : « PARAMETRE=VALEUR ».

Onglet « Options »

Grille

Permet de définir une grille (invisible) pour le positionnement des objets. Seule la commande « Déplacer » du menu intégré au pupitre utilise la grille. Les valeurs pour la grille sont exprimées en nombre de pixels. Les valeurs 0 et 1 annulent l'effet de la grille. Cette fonction doit être utilisée pour aligner parfaitement les objets.

Rafraîchissement des enfants

La case à cocher « Continuer à rafraîchir les enfants ... » détermine si les enfants doivent continuer à être mis à jour lorsque le pupitre est mis en icône. Cette option permet, lorsqu'elle n'est pas sélectionnée, d'augmenter les performances du système lorsqu'un pupitre mis en icône ne contient que des éléments visuels.

Onglet « Enfants »

Enfants

Cette rubrique contient la liste des objets enfants du pupitre. Le bouton poussoir « Propriétés » permet d'ouvrir directement la boîte de dialogue des propriétés de l'enfant sélectionné dans la liste. Le bouton poussoir « Détruire » élimine l'objet sélectionné. Les zones d'édition « Positions » permettent de régler la position des objets.

Onglet « Externe »

Nom de l'exécutable

Nom d'un fichier exécutable fonctionnant dans le pupitre.

Paramètres

Paramètres fournis sur la ligne de commande à l'exécutable.

Détail de l'objet « Bouton Voyant »

Onglet « Aspect »

Type de l'objet

Permet de choisir la nature de l'objet : bouton poussoir, voyant ou bouton poussoir intégrant un voyant.

Couleurs

Permet de choisir les couleurs de l'objet. Si l'objet est un bouton poussoir, le réglage « Fond éteint » représente la couleur du bouton poussoir. Si l'objet est un voyant ou un bouton poussoir intégrant un voyant le réglage « Fond allumé » détermine la couleur du fond lorsque le voyant est allumé et « Fond éteint » la couleur lorsque le voyant est éteint. Seule la couleur des caractères est réglable si l'aspect de l'objet est déterminé par un bitmap.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet. Ceci est nécessaire si l'aspect de l'objet est déterminé par un bitmap.

Fonte de caractères

Permet de choisir la fonte et la taille des caractères. Le fichier de fonte utilisé devra être présent sur le PC ou s'exécute le programme.

Texte

Permet de spécifier le texte affiché sur l'objet, sa position, son orientation ainsi qu'un texte d'aide qui apparaît lorsque le bouton est pressé et un texte bulle qui apparaît lorsque le curseur est placé sur l'objet.

Onglet « Liens »

Action lorsque ...

Permet de définir les actions à réaliser lorsque le bouton est enfoncé et lorsqu'il est relâché.

Une action peut être le forçage de l'état d'une variable, par exemple :

```
00=1, m200=4, _depart cycle_=3
```

Ou un mot clé prédéfini.

Exemple de configuration pour que l'entrée i10 reflète l'état du bouton poussoir (i10 à 0 si le bouton est relâché, i10 à 1 si le bouton est enfoncé) :

Action lorsque le bouton est enfoncé : `i10=1`

Action lorsque le bouton est relâché : `i10=0`

Etat du voyant

Détermine l'état du voyant. Cette rubrique doit contenir le nom d'une variable qui pilotera le voyant : 0 = voyant éteint, 1 = voyant allumé.

Par exemple :

```
b31, o4, _voyant init_, m200=50, m400<8, m500&16
```

Identificateur

Permet de référencer l'objet par rapport aux autres objets.

Condition de désactivation

Permet de désactiver le voyant. Si cette rubrique contient le nom d'une variable, alors cette variable désactive l'objet si elle est vraie.

Onglet « Options »

Type du bouton poussoir

Détermine si le bouton poussoir est bistable (il reste enfoncé), monostable ou une combinaison des deux : monostable sur simple clic et bistable sur double clic.

Clavier

Permet d'associer une touche au bouton poussoir. Si cette touche ou cette combinaison de touche est présente au clavier alors le bouton poussoir sera enfoncé.

Pour préciser le code de la touche différentes syntaxes sont utilisables :

- un simple caractère : par exemple A, Z, 2,
- le caractère \$ suivi du code de la touche en hexadécimal,
- le nom d'une touche de fonction, par exemple F5.

Pour les combinaisons de touches il faut ajouter au début CTRL+ ou SHIFT+.

Par exemple : CTRL+F4 ou SHIFT+Z.

Bitmap

Permet de spécifier un fichier bitmap qui contiendra le dessin de l'objet.

L'option « Redimensionner l'image » permet d'étendre le bitmap sur toute la surface de l'objet.

Le fichier bitmap contient les quatre aspects possibles de l'objet : bouton relâché voyant éteint, bouton enfoncé voyant éteint, bouton relâché voyant allumé et bouton enfoncé voyant allumé.

Même si le fichier est un bouton poussoir sans voyant ou un voyant il y a toujours quatre aspects dont seuls deux sont utilisés.

Le fichier bitmap est découpé en quatre horizontalement.

Exemple :



L'option « Aspect différent si le curseur est sur l'objet ... » permet d'afficher une image différente lorsque l'objet passe sur l'objet.

Si cette option est cochée, le fichier bitmap contient 8 aspects, 4 aspects supplémentaires sont ajoutées à droite du bitmap pour contenir le dessin de l'objet lorsque le curseur est sur l'objet.

Exemple :



Sons

En choisissant des fichiers .WAV, l'objet peut produire des sons si l'objet est enfoncé, relâché, ou si le curseur passe sur l'objet.

Détail de l'objet « Valeur digitale »

Onglet « Aspect »

Format

Permet de définir le type d'affichage :

- Toujours montrer le signe : affiche le signe '+' pour les valeurs signées positives,
- Valeur signée : définit le mode signé ou non signé pour les entiers 16 ou 32 bits (base 10 uniquement),
- Afficher tous les digits : affiche des 0 au début de la valeur si nécessaire.

Base

- Détermine la base d'affichage pour les entiers 16 et 32 bits.

Couleurs

Permet de choisir les couleurs du fond de l'objet (s'il n'est pas transparent) et des caractères.

Fonte de caractères

Permet de choisir la fonte et la taille des caractères. Le fichier de fonte utilisé devra être présent sur le PC ou s'exécute le programme.

Nombre de digits

Définit la longueur de la partie entière et décimale.

Fond

Permet de choisir entre un fond coloré ou transparent (si l'objet est posé sur un pupitre seulement).

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Onglet « Textes »

Texte Bulle

Texte affiché dans une bulle lorsque l'utilisateur déplace le curseur sur l'objet.

Texte affiché avant et après la valeur

Permet d'afficher des informations à gauche et à droite de la valeur numérique.

Onglet « Liens »

Variable ou symbole

Désigne la variable à afficher. Pour accéder au compteur ou à la consigne d'une temporisation il faut utiliser la syntaxe suivante :

- pour le compteur : COUNT(temporisation), exemple : COUNT(t3),
- pour la consigne : PRED(temporisation), exemple : PRED(t7),

L'état de la variable est modifiable ...

Si cette case est cochée alors l'utilisateur peut modifier l'état de la variable en cliquant sur l'objet.

Détail de l'objet « Valeur analogique »

Onglet « Aspect »

Objets

Permet de définir le type d'affichage.

Orientation

Détermine l'orientation : horizontale ou verticale.

Couleurs

Permet de choisir les couleurs du fond de l'objet (s'il n'est pas transparent) et de l'objet.

Fond

Permet de choisir entre un fond coloré ou transparent (si l'objet est posé sur un pupitre seulement).

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Textes

Le texte bulle.

Onglet « Liens »

Variable ou symbole

Désigne la variable liée à l'objet (un mot ou un compteur).

Les actions de l'utilisateur ...

Détermine si l'état de la variable peut être modifié par l'utilisateur.

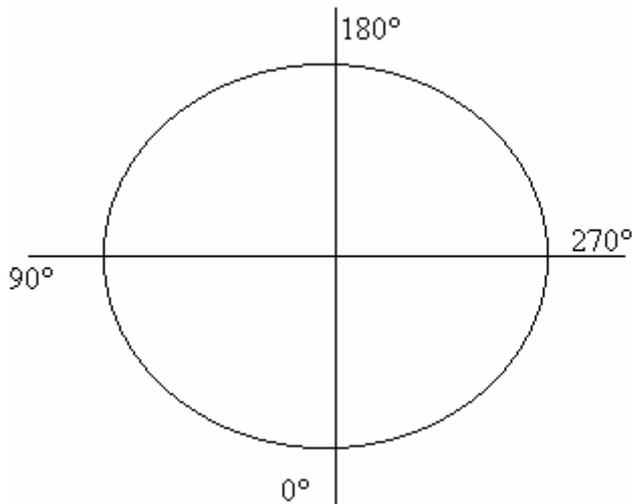
Onglet « Bornes »

Minimum, maximum

Valeurs minimales et maximales.

Angle de départ, angle de fin

Pour l'affichage de type cadran détermine l'angle de départ et l'angle d'arrivée. Les valeurs sont précisées en degrés :



Onglet « Graduations »

Utiliser les graduations

Valide ou invalide l'utilisation des graduations

Valeur de départ, valeur de fin

Valeurs affichées pour les graduations, ces valeurs peuvent être des nombres signés et/ou des nombres à virgule.

Pas des petites graduations, pas des grandes graduations

Pas des graduations (deux niveaux) par rapport aux valeurs de départ et de fin. Ces valeurs peuvent être des nombres à virgule.

Police de caractère

Définit les caractères utilisés pour les graduations.

Zone N°1, zone N°2 et zone N°3

Permet de définir des zones colorées. « Valeur de départ » et « Valeur de fin » définissent chaque zone. La couleur pour chaque zone est spécifiée par trois composantes le rouge, le vert et le bleu comprises entre 0 et 255.

Couleurs

Détermine la couleur des caractères et des graduations. Les couleurs sont ici aussi exprimées par leurs trois composantes : le rouge, le vert et le bleu.

Détail de l'objet « Ecran, clavier, liste de messages »

Liens avec l'application

Le lien entre l'objet et l'application se fait par des tables de mots.

Pour envoyer des données au type d'objet (avec ou sans clavier) il faut placer les données à partir du deuxième mot de la table de réception puis la longueur des données dans le premier mot de la table (la longueur maximale est 255). Chaque mot contient une donnée.

Les données peuvent être : un caractère ASCII, un numéro de message prédéfini + 8000 hexa, ou une commande spéciale : 100 hexa efface la fenêtre, 200 hexa affiche la date, 300 hexa affiche l'heure, 400 affiche le numéro du message.

Lorsque l'objet a relu les données disponibles dans une table il remet le premier mot à 0 pour signaler que l'opération a été effectuée.

Pour les objets « avec clavier » le principe est le même : le premier mot de la table d'émission contient le nombre de caractères entrés au clavier, les mots suivants contiennent les caractères (un par mot). L'application doit remettre à 0 le premier mot lorsqu'elle a utilisé les données.

Pour l'objet « Boîte à messages, liste d'alarmes » la table d'échange à une longueur fixe de 10 mots. Comme pour le type « Ecran » le premier mot déclenche l'affichage de message. S'il est différent de 0 il désigne un numéro de message à afficher. Seuls des messages enregistrés peuvent être affichés. Le premier mot peut également prendre la valeur ffff hexa pour vider la boîte à messages.

Description des 10 mots utilisés pour les échanges avec le type « Boîte à messages » :

Mot 0 représente le premier mot de la table, Mot 1 le deuxième, etc ...

Mot 0 : numéro de message à afficher ou 0 si pas de message ou ffff hexa pour tout effacer,

Mot 1 : numéro de classe pour le message (voir chapitre Les classes de messages pour plus d'explications),

Les mots qui suivent déterminent la date et l'heure qui peuvent être affichées pour chaque message. Une valeur égale à ffff hexa demande à l'objet d'utiliser la date et l'heure courante de l'ordinateur (cela ne concerne pas les millisecondes).

Mot 2 : jour

Mot 3 : mois

Mot 4 : année

Mot 5 : heures

Mot 6 : minutes

Mot 7 : secondes

Mot 8 : millisecondes

Mot 9 : réservé (mettre 0)

Les classes de messages

Les classes de messages permettent de classifier les messages en famille qui partageront les caractéristiques suivantes : la couleur du fond, la couleur des caractères et une icône.

Il existe deux classes prédéfinies :

- la classe message d'information : caractères bleus sur fond blanc, icône , elle porte le numéro -1,
- la classe message d'alarme : caractères blancs sur fond rouge, icône , elle porte le numéro -2.

D'autres classes peuvent être définies par l'utilisateur.

Un texte bulle peut être associé à l'objet.

Onglet « Aspect »

Type de l'objet

Permet de définir un des types d'objet. Voir le chapitre Liens avec l'application.

Couleurs

Permet de choisir les couleurs du fond de l'objet et des caractères.

Fonte

Permet de choisir la fonte de caractères utilisée pour afficher les textes.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Textes

Le texte bulle.

Onglet « Liens »

Réception, émission

Définissent les premières variables de la table de réception et d'émission. Ces zones peuvent contenir un nom de variable ou un symbole.

Onglet « Liste »

Ces rubriques ne concernent que les objets de type « Boîte à messages ».

Icônes

Cette case à cocher détermine s'il faut afficher une icône avant les messages.

Classes

Cette case à cocher détermine s'il faut afficher le numéro de la classe du message.

Jours, Mois, Années, Heures, Minutes, Secondes, 1/1000 secondes

Ces cases à cocher déterminent s'il faut afficher chacun de ces éléments.

Messages

Cette case à cocher détermine s'il faut afficher un message.

Numéros

Cette case à cocher détermine s'il faut afficher le numéro d'apparition des messages.

Classes de messages

Cette zone d'édition permet de définir de nouvelles classes de messages. Chaque ligne définit une classe. Sur chaque ligne on doit trouver dans l'ordre et séparés par des virgules : la couleur du fond, (trois composantes rouge, vert et bleu) la couleur des caractères (trois composantes rouge, vert et bleu), le nom de la classe, le nom d'un fichier bitmap pour l'icône associé à la classe.

Par exemple :

255,0,0,0,0,0,ALARME,alarme.bmp

Signifie :

Couleur du fond rouge, couleur des caractères noir, nom de la classe ALARME, fichier contenant l'icône : « alarme.bmp ».

Onglet « Options »

Afficher les codes hexadécimaux des caractères

Cette option permet d'afficher le code hexadécimal pour chaque caractère à la place de sa représentation ASCII. Elle est utilisable pour les objets de type « Ecran ... » et permet notamment la mise au point des programmes.

Barre de défilement horizontale, verticale

Montre ou cache les barres de défilements.

Convertir les caractères OEM -> ANSI

Si cette case est cochée, alors les caractères provenant de l'application d'automatisme sont automatiquement convertis du jeu de caractères OEM (MS-DOS) au jeu de caractères ANSI (WINDOWS). La conversion inverse est appliquée aux caractères allant de l'objet à l'application d'automatisme.

Dupliquer les messages vers ...

Cette rubrique peut recevoir un nom de fichier ou de périphérique (« LPT1 » par exemple pour l'imprimante). Il est possible de spécifier plusieurs fichiers et/ou périphériques en les séparant par une virgule. Les affichages seront automatiquement dupliqués : impression dite « au fil de l'eau ».

Associer un fichier de stockage des messages ...

Permet de définir un fichier qui sera associé à l'objet et utilisé pour le stockage des informations. Si un tel fichier existe alors les messages sont sauvegardés (à concurrence du nombre défini sous la rubrique « nombre de lignes mémorisées », si ce nombre est atteint les données les plus anciennes sont effacées). Lorsque l'objet est ouvert, et si un fichier de stockage existait lors de sa dernière utilisation, alors les données contenues dans le fichier sont transférées dans l'objet.

Ecrire les messages périmés vers ...

Permet de définir un fichier ou un périphérique qui recevra les messages périmés (les messages qui sont éliminés du fichier de stockage pour libérer de la place).

Nombre de lignes mémorisées ...

Détermine la capacité du fichier de stockage des messages en nombre de lignes. La valeur 0 attribue le maximum d'espace mémoire utilisable (pas de limite fixe).

Onglet « Messages »

Messages prédéfinis

Cette boîte d'édition permet de documenter les messages prédéfinies (un par ligne).

Détail de l'objet « Archivage de données »

Onglet « Aspect »

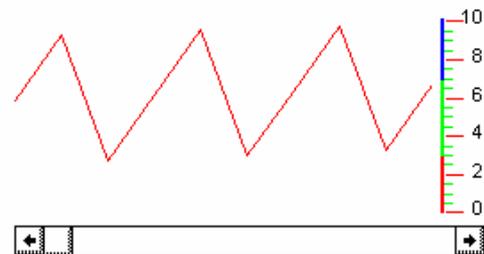
Objets

Permet de définir le type d'affichage.

L'objet peut être représenté sous la forme d'une table (*figure 1.1*) ou sous la forme de tracés de courbes (*figure 1.2*).

Date	Heure d'acquisition	Valeur
23/07/96	16.52.52.443	-28043
23/07/96	16.52.53.541	-6059
23/07/96	16.52.54.640	16477
23/07/96	16.52.55.738	-26575
23/07/96	16.52.56.837	-4091
23/07/96	16.52.57.935	18441
23/07/96	16.52.59.034	-24579
23/07/96	16.53.00.132	-2067

(figure 1.1)



(figure 1.2)

Couleurs

Permet de choisir la couleur de la police de caractère lorsque l'objet est sous la forme d'une table ainsi que la couleur de marquage des valeurs sur les courbes.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Texte

Le texte bulle associé à l'objet.

Onglet « Données »

Première variable à lire

Permet de sélectionner la première variable qu'il faudra archiver.

Nombre de variables à lire

Signale à l'objet ARCHIVE le nombre de variables consécutives à la « Première variable à lire » qu'il faudra archiver .

Nombre d'enregistrements mémorisés

Permet de dimensionner la base de données en mémoire .

Un enregistrement représente une acquisition de « n » variables (« n » étant le nombre de variables à lire).

Lecture périodique

L'acquisition des variables se fera à intervalle fixe des l'exécution de l'objet ARCHIVE

Lecture déclenchée

Une acquisition des variables sera effectuée dès que le « Mot de contrôle » en aura donné l'ordre .

Période

Permet de fixer le temps qui sépare deux acquisitions. Le temps est au format Jour(s) /Heure(s) /Minute(s) /Seconde(s) /Milliseconde(s) :

J pour les heures

H pour les heures

M pour les minutes

S pour les secondes

MS pour mes millisecondes

Ex : 2J

Ex : 2H10M15S

Contrôle

Permet de définir une variable (un mot) qui contrôlera l'objet ARCHIVE. Dès la valeur prise en compte, son contenu est forcé à 0 par l'objet ARCHIVE.

Valeur

0

1

Action

Rien

Déclenchement d'une acquisition (Lecture déclenchée)

- 2 Geler les acquisitions
- 3 Reprendre l'archivage (après avoir gelé)
- 4 Vider la base de données en mémoire
- 5 Détruire le fichier d'archivage
- 6 Activer le mode « Suivi des dernières acquisitions »
- 7 Annuler le mode « Suivi des dernières acquisitions »

Onglet « Options »

Utiliser le fichier image

Le fichier image est utilisé :

En fin d'utilisation de l'objet ARCHIVE, pour sauvegarder la base de données présente en mémoire.

Au lancement de l'objet ARCHIVE, pour reconstituer la base de données qui était présente en mémoire lors de la dernière utilisation.

Utiliser le fichier archive

Chaque acquisition est enregistrée dans ce fichier au format base de données standard.

Affichage

Date d'acquisition: Permet d'afficher la date d'acquisition d'un enregistrement.

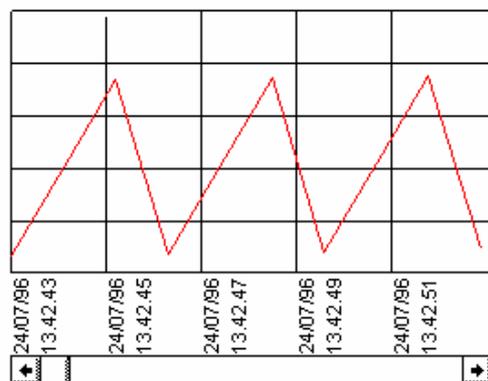
Heure d'acquisition: Permet d'afficher l'heure d'acquisition d'un enregistrement.

Heures, minutes, secondes, millisecondes: Permet de configurer l'affichage de l'heure d'acquisition.

L'affichage du temps se fait en aval de l'affichage des acquisitions pour l'objet TABLE (*figure 3.1*) ou sous la grille lorsqu'elle est définie pour l'objet COURBE (*figure 3.2*)

Date	Heure d'acquisition	Valeur
24/07/96	13.42.43.112	3141
24/07/96	13.42.44.211	25217
24/07/96	13.42.45.309	-18451
24/07/96	13.42.46.408	3489
24/07/96	13.42.47.506	25525
24/07/96	13.42.48.605	-17931
24/07/96	13.42.49.703	4085
24/07/96	13.42.50.802	26149
24/07/96	13.42.51.900	-17375
24/07/96	13.42.52.999	4709

(figure 3.1)



(figure 3.2)

Onglet « Tables »

Police de caractères

Permet de sélectionner une police de caractères pour l'affichage du nom des colonnes, du temps et la valeur des acquisitions.

Nom des colonnes

Permet de définir le nom des colonnes pour l'objet TABLE ainsi que le format d'affichage de ces colonnes (*figure 4.1*).

syntaxe : nom, format

format *

pas de format spécifié

h

d

ns

s

nv

v

Affichage

Signé, décimal, visible

Hexadécimal

Décimal

Non signé

Signé

Non visible

Visible

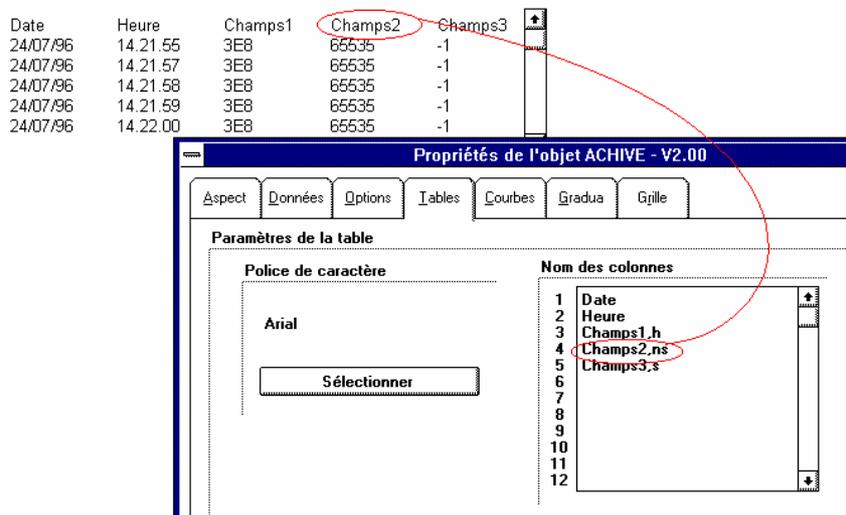
* On peut combiner les différentes options, par exemple:

Format

d,ns,v

Affichage

Décimal sans signe visible



(figure 4.1)

Onglet « Courbe »

Valeur minimale, maximale

Permet de sélectionner la valeur minimale et maximale pour l'affichage des courbes.

Seules les valeurs comprises entre la valeur minimale et la valeur maximale seront visibles sur l'écran.

Visualiser

Permet de définir le temps de visualisation. Celui-ci est communiqué à l'objet ARCHIVE suivant le format Jour(s) / Heure(s) / Minute(s) / Seconde(s) / Milliseconde(s) :

J pour les jours

H pour les heures

M pour les minutes

S pour les secondes

MS pour mes millisecondes

Ex : **Visualiser** 2H30M10S

Ex : **Visualiser** 100MS

Marquage des valeurs sur la courbe

Permet de faire une marque sur la courbe pour chaque acquisition (*figure 5.1*).

Affichage du temps

Permet d'afficher la date et l'heure d'acquisition d'une ou plusieurs variables sous la grille si elle est active. On peut définir la couleur et la police de caractères de l'affichage du temps.

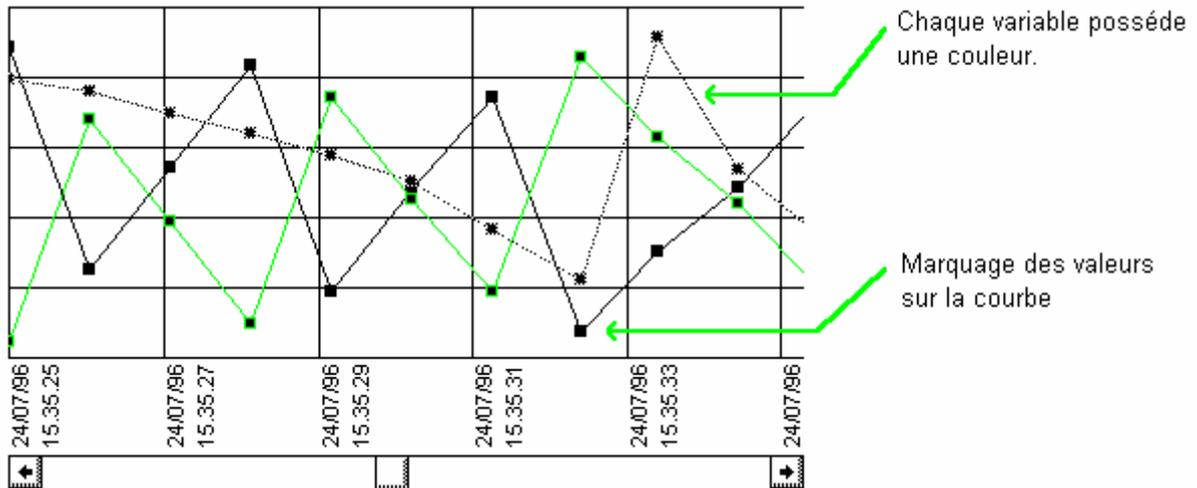
Couleurs des tracés

Permet de définir une couleur pour chaque courbe. La première courbe possède la couleur de la première ligne, la deuxième courbe possède la couleur de la deuxième ligne etc...

Les couleurs sont au format Rouge, Vert, Bleu.

Ex: 255,0,0 tracé rouge

Si une couleur n'est pas définie sur une ligne, la courbe correspondante à cette ligne ne sera pas tracée.



(figure 5.1)

Onglet « Graduations »

Utiliser les graduations

Valide ou invalide l'utilisation des graduations (figure 6.1).

Valeur de départ, valeur de fin

Valeurs affichées pour les graduations, ces valeurs peuvent être des nombres signés et/ou des nombres à virgule.

Pas des petites graduations, pas des grandes graduations

Pas des graduations (deux niveaux) par rapport aux valeurs de départ et de fin. Ces valeurs peuvent être des nombres à virgule.

Police de caractère

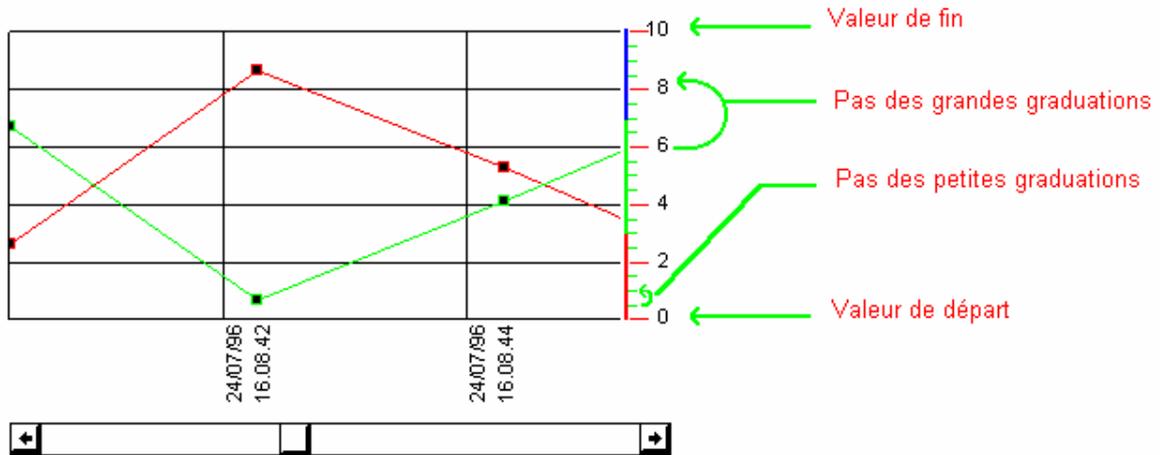
Définit les caractères utilisés pour les graduations.

Zone N°1, zone N°2 et zone N°3

Permet de définir des zones colorées. "Valeur de départ" et "Valeur de fin" définissent chaque zone. La couleur pour chaque zone est définie par trois composantes le rouge, le vert et le bleu comprises entre 0 et 255.

Couleurs

Détermine la couleur des caractères et des graduations. Les couleurs sont ici aussi exprimées par leur trois composantes : le rouge, le vert et le bleu.



(figure 6.1)

Onglet « Grille »

Afficher la grille

Valide ou invalide l'affichage de la grille.

Pas pour les ordonnées

Définit le pas vertical de la grille.

Pas pour les abscisses

Définit le pas horizontal de la grille. Le pas est au format Jour(s) / Heure(s) / Minute(s) / Seconde(s) / Milliseconde(s):

J pour les jours

H pour les heures

M pour les minutes

S pour les secondes

MS pour les millisecondes

Ex : 1J

Ex : 2H30M15S

Couleur

Permet de définir une couleur pour la grille.

La couleur est au format Rouge, Vert, Bleu

Ex : 255,0,0 Tracé rouge

Détail de l'objet « Objet »

Onglet « Aspect »

Type

Permet de définir un des types d'aspect de l'objet :

- « bitmap n aspects » : l'aspect de l'objet est donné par un fichier bitmap qui peut contenir plusieurs aspects voir le chapitre Onglet « Bitmap ».
- « bitmap n couleurs » : l'aspect de l'objet est donné par un fichier bitmap, la couleur est contrôlée par une variable de l'application d'automatisme qui remplace les pixels blancs du bitmap. Les autres pixels du bitmap doivent être noirs. La variable de l'application d'automatisme donne un numéro de couleur, les couleurs sont définies dans l'onglet « Couleurs ».
- « bitmap jauge » : l'objet est une jauge dont la forme est définie par un bitmap. Ce sont les pixels blancs du bitmap qui définissent la forme. Les autres pixels doivent être noirs. Le minimum, le maximum et l'orientation sont réglés dans l'onglet « Jauge ».
- « formes n couleurs » : un rectangle, un rectangle aux bords arrondis ou une ellipse. La couleur est gérée de la même façon que pour le type « bitmap n couleurs ».
- « formes jauges » : l'objet est une jauge rectangulaire. Le principe est le même que pour le type « bitmap jauge ».

Couleurs

Permet de choisir la couleur des caractères du texte affiché sur l'objet.

Fonte de caractères

Détermine la fonte utilisée pour afficher un texte sur l'objet.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Textes

Le texte d'aide et le texte bulle.

Le texte affiché sur l'objet : la position et l'orientation peuvent être modifiées.

Onglet « Liens »

Objet cliqué, objet non cliqué

Définit les actions à réaliser respectivement lorsque l'utilisateur clique sur l'objet et lorsqu'il arrête de cliquer l'objet.

Une action peut être le forçage de l'état d'une variable, par exemple :

```
o0=1, m200=4, _depart cycle_=3
```

Ou un mot clé prédéfini.

Exemple de configuration pour que l'entrée i10 reflète l'état cliqué de l'objet (i10 à 0 si l'objet n'est pas cliqué, i10 à 1 si l'objet est cliqué) :

Objet cliqué : `i10=1`

Objet non cliqué : `i10=0`

Lier en permanence avec ...

Cette zone peut recevoir l'identificateur d'un objet frère. Si un tel objet existe alors la position de l'objet est calquée sur celui-ci. L'identificateur d'un objet est une valeur entière comprise entre 1 et 32767. Il est spécifié dans la zone d'édition « Identificateur » de la rubrique « Liens ».

Aspect / Couleur / Remplissage

Cette zone de la boîte de dialogue regroupe 8 zones d'édition qui permettent de définir différents types de comportement de l'objet par rapport aux variables de l'application d'automatisme.

Quel que soit le type de comportement il y aura toujours une position qui selon le type d'objet désignera :

- un aspect contenu dans un bitmap pour le type « bitmap n aspects »,
- un numéro de couleur pour les types « bitmap n couleurs » ou « forme n couleurs »,
- un remplissage pour les types « bitmap jauge » ou « forme jauge ».

La zone « Position » peut contenir un nom de variable numérique (C ou M). Les zones « + Position » et « - Position » peuvent contenir un nom de variables booléennes.

Deux types de fonctionnement sont possibles :

- si les zones « + Position » et « - Position » sont documentées alors les variables booléennes qu'elles contiennent pilotent la position : elles ajoutent ou enlèvent la valeur précisée dans la zone vitesse. Si la zone « Position » est documentée alors la position courante est écrite dans la variable dont elle contient le nom.

- si les zones « + Position » et « - Position » sont vides alors la valeur contenue dans la variable dont le nom est écrit dans la zone « Position » est lue comme position de l'objet.

La position peut varier entre les valeurs définies dans les zones « Mini » et « Maxi ».

Des capteurs peuvent être ajoutés (des noms de variables booléennes) qui seront vrais pour la position minimale et maximale (position égale au minimum ou au maximum).

Déplacement horizontal, déplacement vertical

Ces zones de la boîte de dialogue regroupent chacune 8 zones d'édition qui permettent respectivement de définir la position horizontale et verticale de l'objet. Le principe est identique à celui décrit ci-dessus.

Onglet « Formes »

Formes

Pour le type « Forme n couleurs » cette rubrique permet de sélectionner un rectangle, un rectangle avec des coins arrondis ou une ellipse.

Onglet « Bitmap »

Nom du fichier

Pour les types « Bitmap n aspects, bitmap n couleurs et bitmap jauge » cette zone d'édition doit contenir un nom d'accès complet vers un fichier « .BMP ». Ces fichiers peuvent être créés avec PAINTBRUSH ou tout autre éditeur graphique capable de créer des fichiers « .BMP ».

Les boutons poussoir « Parcourir » et « Editer » permettent respectivement de rechercher un fichier « .BMP » et d'éditer (lancement de PAINTBRUSH) le fichier « .BMP » dont le nom se trouve dans la zone d'édition.

Nombre d'aspects

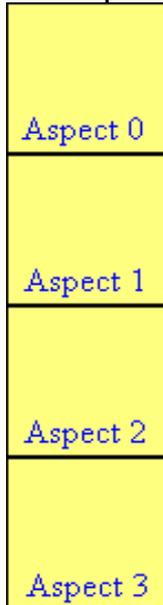
Cette zone d'édition doit contenir le nombre d'aspect (d'images) contenus dans un fichier « .BMP ». Cette option est utilisée pour le type « Bitmap n aspects ». Les différents aspects d'un objet doivent être dessinés les uns au dessous des autres. L'aspect le plus haut est le numéro 0.

Onglet « Wmf »

Nom du fichier

Pour les types « Métafichiers » cette zone d'édition doit contenir un nom d'accès complet vers un fichier « .WMF » ou « .EMF ».

Exemple de fichier « .BMP » à 4 aspects :



Le bitmap possède des zones transparentes ...

Cette option permet de créer un objet dont certaines parties seront transparentes (c'est le fond du pupitre père qui sera affiché). Les zones transparentes seront définies par des pixels possédant la même couleur, couleur qui sera déterminée par les trois composantes, rouge, verte et bleue. Pour régler ces composantes utilisez les trois barres de défilement. La couleur doit être réglée de façon très précise : exactement la même proportion de rouge, de vert et de bleu que la couleur des pixels des zones transparentes du bitmap.

Onglet « Couleurs »

Couleurs

Cette rubrique est utilisée pour les types « bitmap n couleurs » et « forme n couleurs ». Chaque ligne contient la définition pour une couleur. La syntaxe utilisée pour chaque ligne est : proportion de rouge (entre 0 et 255), proportion de vert (entre 0 et 255) et proportion de bleu (entre 0 et 255). La première ligne désigne la couleur numéro 0, la deuxième la numéro 1, etc ...

Cette rubrique est aussi utilisée pour les types « bitmap jauge » et « forme jauge ». La première ligne (couleur 0) et la deuxième (couleur 1) détermine les deux couleurs de la jauge (partie active et inactive).

Onglet « Jauge »

Jauge

Cette rubrique est utilisée pour les types « bitmap jauge » et « forme jauge ». Les zones « Valeur minimum » et « Valeur maximum » définissent les bornes pour la variable de pilotage de la jauge.

Orientation de la jauge

Détermine une des quatre directions possibles pour la jauge.

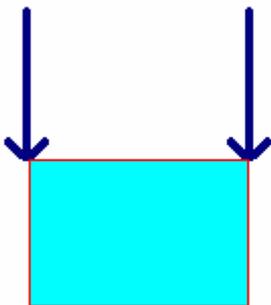
Onglet « Capteur »

L'objet OBJET peut être utilisé comme capteur. Le capteur est associé à une variable booléenne dont le résultat est vrai si le capteur est en contact avec une ou plusieurs couleurs prédéfinies (voir plus bas) ou faux autrement.

Position de la détection

Permet de déterminer le côté de l'objet qui doit détecter. La détection est réalisée sur les deux bords du côté choisi.

Exemple pour une détection au dessus :



Couleurs détectées

Un capteur est capable de détecter jusqu'à trois couleurs différentes. Si une de ces trois couleurs se trouve aux points de test alors la variable booléenne associée au capteur (voir chapitre Onglet « Liens ») est positionnée à 1, autrement elle est positionnée à 0.

Les trois zones d'édition peuvent contenir une définition de couleur sous la forme de trois valeurs comprises entre 0 et 255 et qui correspondent respectivement aux pourcentages de rouge, de vert et de bleu. Les pourcentages de ces trois couleurs doivent correspondre avec exactitude aux couleurs des objets à détecter pour que le capteur fonctionne.

Onglet « Options »

Touche

Définit une touche qui permet de simuler un clique sur l'objet.

Pour préciser le code de la touche, différentes syntaxes sont utilisables :

- un simple caractère : par exemple A, Z, 2,
- le caractère \$ suivi du code de la touche en hexadécimal,
- le nom d'une touche de fonction, par exemple F5.

Pour les combinaisons de touches il faut ajouter au début « CTRL+ » ou « SHIFT+ »

Par exemple : « CTRL+F4 » ou « SHIFT+Z ».

La touche TAB permet d'accéder à cet objet

Si cette case n'est pas cochée, alors la touche TAB ne permet pas d'activer l'objet.

Techniques avancées

Lier des objets de façon dynamique

Cette possibilité permet de lier momentanément un objet à un autre. Pour lier un objet à un autre les paramètres « + Position » et « - Position » qui gèrent la position horizontale sont utilisés de façon particulière. Ces deux paramètres doivent contenir le nom d'une variable numérique (M). La variable « + Position » doit contenir la valeur f000 (hexadécimal) et la variable « - Position » l'identificateur de l'objet avec lequel l'attachement doit être effectué. La variable « + Position » est remise à zéro lorsque l'attachement a été réalisé. Pour annuler l'attachement d'un objet il faut placer la valeur f001 (hexadécimal) dans la variable « + Position ». Voir le chapitre : Exemple de simulation d'une partie opérative 1

Echange de paramètres entre deux objets

Un objet peut accéder aux paramètres d'un objet frère en utilisant le mot clé « SISTERPARAM ».

La syntaxe est :

SISTERPARAM(identificateur de l'objet frère , paramètre)

« paramètre » peut prendre les valeurs suivantes :

STATE	état de l'objet : valeur de Aspect / Couleur / Remplissage
-STATE	idem mais valeur négative
POSX	position sur l'axe horizontal
-POSX	idem mais valeur négative

POSY	position sur l'axe des y
-POSY	idem mais valeur négative
POSX+STATE	position sur l'axe horizontal plus état
POSX-STATE	position sur l'axe horizontal moins état
POSY+STATE	position sur l'axe vertical plus état
POSY-STATE	position sur l'axe vertical moins état

Détail de l'objet « Son »

Onglet « Aspect »

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Onglet « Sons »

Nom des fichiers sons

Le nom d'accès complet vers les fichiers « .WAV ».

Variables associées

La variable booléenne associée à chaque son.

Détail de l'objet « Boîte de dialogue »

Onglet « Aspect »

Type de boîte

Permet de choisir les différents contrôles présents dans la boîte de dialogue : un seul bouton OK, deux boutons OK et CANCEL, ou enfin deux boutons OUI et NON.

Icônes

Permet de choisir l'icône qui apparaîtra dans la boîte de dialogue. Il y a quatre icônes différentes, mais bien sur, on peut décider d'en afficher aucune. Il convient de noter aussi qu'à chaque icône est associé un son système particulier. Pour plus de renseignements à ce sujet, se reporter à la rubrique sur l'option BEEP.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Beep

Permet de spécifier si l'apparition de la boîte de dialogue doit être accompagnée d'un avertissement sonore.

Titre

Permet de spécifier le titre de la boîte de dialogue.

Type de message

Il y a ici deux possibilités. Un message prédéfini est un message présent dans les variables utilisateurs de l'application d'automatisme. L'autre possibilité est de spécifier une liste de messages, et dans ce cas, le message affiché sera fonction de l'état de la variable surveillée.

Onglet « Liens »

Nom de la variable

Spécifie le nom de la variable à surveiller. On peut entrer indifféremment des variables booléennes ou numériques.

Par exemple :
m200, i0

Si la variable en question est booléenne, alors le message n°1 dans la liste s'affichera dès que l'état de cette variable passera à 1.

Pour une variable numérique, si l'option de configuration « Liste de message » est cochée, alors la boîte de dialogue s'affichera dès que la valeur sera comprise entre 1 et le nombre de messages mémorisés dans la liste.

Par exemple, si la liste contient 8 messages, alors il ne se passera rien quand la variable prendra des valeurs négatives ou supérieures à 8. Par contre, dès que sa valeur passe dans l'intervalle 1..8, alors le message approprié est affiché...

Si l'option « Message prédéfini » est activée, alors la boîte de dialogue s'affichera avec un message de longueur le contenu de la variable, et

situé dans les variables de l'application d'automatisme suivant cette variable.

Par exemple, si $m200=4$, cela signifie qu'un message de longueur 4 caractères est situé dans les 4 variables suivants $m200$, c'est à dire $m201$, $m202$, $m203$, $m204$.

Code de retour d'une boîte de dialogue

Dans le cas d'une variable booléenne, quelle que soit l'action effectuée par l'utilisateur, son contenu passera à 0. Pour une variable numérique, il existe différents codes de retour :

Appui sur un bouton OK : la variable prend la valeur 8000 (hexa)
Appui sur un bouton CANCEL: la variable prend la valeur 8001 (hexa)
Appui sur un bouton OUI : la variable prend la valeur 8002 (hexa)
Appui sur un bouton NON : la variable prend la valeur 8003 (hexa)

Remarque : L'activation d'une boîte de dialogue est basée sur le front montant, c'est à dire passage de 0 à 1 pour une variable booléenne, et passage d'une valeur extérieure à l'intervalle de la liste de messages, à une valeur incluse dans celui-ci, pour une variable numérique.

Identificateur

Permet de référencer l'objet par rapport aux autres objets.

Onglet « Messages »

Liste de messages

Entrer dans cette zone les différents messages prédéfinis.

Détail de l'objet « Programme »

Répartition du temps d'exécution

Les objets d'IRIS sont exécutés tour à tour. Cette répartition du temps d'exécution est gérée de façon transparente par le gestionnaire d'objet. Deux niveaux de priorité sont possibles pour les objets « PROG » : si la case à cocher « Exécution prioritaire » de l'onglet « Programme » est cochée, alors la totalité du programme s'exécute lorsque l'objet à la main. Dans le cas contraire, seule une ligne s'exécute avant que l'objet

ne rende la main. Des exceptions existent à cette règle : les fonctions d'accès aux variables d'automatisme (« READVAR » et « WRITEVAR ») peuvent provoquer un passage de la main, la fonction « YIELD » force un passage de la main. En mode d'exécution prioritaire, cette fonction doit obligatoirement être utilisée à l'intérieur d'une boucle pour ne pas bloquer l'exécution des autres objets.

Affichage

La surface de l'objet peut être utilisée pour afficher des informations. La fonction « PRINT » permet d'afficher des informations.

Syntaxe

Le caractère « ; » (point virgule) est utilisé comme séparateur. Des commentaires peuvent être écrits entre les chaînes « (* » et « *) ». Il n'y a pas de différence entre les majuscules et les minuscules pour les mots clés et les noms de fonctions, en revanche, pour les noms de variables la différenciation est faite.

Déclaration des variables

Les variables utilisées dans un programme doivent être déclarées avant le programme entre les mots clés « BEGINVAR; » et « ENDVAR; ».

Les types de variables utilisables sont :

INT	entier 16 bits signé
UINT	entier 16 bits non signé
LONG	entier 32 bits signé
ULONG	entier 32 bits non signé
STRING	chaîne de caractères
FLOAT	flottant

La syntaxe générale d'une déclaration est :

<type> <nom de variable>;

La syntaxe générale pour déclarer un tableau de variables est :

<type> <nom de variable> [<longueur>];

Par exemple :

```
BEGINVAR;
INT compteur;    (* un entier 16 bits signé *)
STRING chaine;  (* une chaîne *)
(* un tableau de 100 entiers 32 bits non signés *)
ULONG table[100];
ENDVAR;
```

Ecriture du programme

Le programme doit être écrit entre les mots clés « BEGIN; » et « END; ».

Exemple :

```
BEGIN;  
print "Bonjour !";  
END;
```

Constantes

- entier 16 bits : un nombre décimal compris entre -32768, et 32727 ou "\$" suivi d'un nombre hexadécimal compris entre 0 et FFFF. Exemple : 12, -4, \$abcd
 - entier 32 bits : un nombre décimal compris entre -2147483648 et 214743648 suivi de "L" ou "\$" suivi d'un nombre hexadécimal compris entre 0 et FFFFFFFF suivi de "L". Exemple: 10000L, -200000L, \$12345678L
 - chaîne de caractères : un caractère guillemet suivi des caractères de la chaîne suivi d'un caractère guillemet. Des caractères de contrôles peuvent être insérés dans une chaîne. « \n » remplace un caractère LF (code ASCII 10), « \r » un caractère CR (code ASCII 13). Exemple : "Abcdef", "" (chaîne nulle), "Suite\r\n"
- flottant : un nombre décimal suivi du caractère "R", le caractère "." sert de délimiteur entre la partie entière et la partie décimale. Exemple : 3.14r, -100.4r

Affectation

La chaîne « := » marque une affectation.

Exemple :

```
compteur:=4;  
var:="ABCDEF";
```

Calculs

Les opérateurs de calculs sont évalués de la gauche vers la droite. Des parenthèses peuvent être utilisées pour spécifier une priorité de calcul.

Liste des opérateurs de calcul :

- + addition (concaténation pour les chaînes de caractères)
- - soustraction
- * multiplication
- / division
- << décalage à gauche
- >> décalage à droite
- ^ élévation à une puissance
- AND "et" binaire
- OR "ou" binaire
- XOR "ou exclusif" binaire

Exemples :

```
resultat:=var1*(var2+var3);  
resultat:=resultat<<2;
```

Tests

Syntaxe :

```
IF <condition> THEN ... ENDIF;
```

ou

```
IF <condition> THEN ... ELSE ... ENDIF;
```

Exemple :

```
IF (count<100) AND (count>10)  
    THEN  
    count:=count+1;  
    ELSE  
    count:=0;  
    ENDIF;
```

Boucles

Syntaxe :

```
WHILE <condition> DO ... ENDWHILE;
```

Exemple :

```
count:=0;
WHILE count<1000
    DO
        table[count]:=table[count+1];
        count:=count+1;
    ENDWHILE;
```

Adresse d'une variable ou d'un tableau de variables

La syntaxe &nom de variable ou &nom d'un tableau de variables retourne l'adresse d'une variable ou d'un tableau de variables. Cette syntaxe est nécessaire pour certaines fonctions.

Liste des fonctions

Pour les exemples proposés dans ce qui suit on admettra que :
vint est une variable de type INT, vlong une variable de type LONG, vuint
une variable de type UINT, vulong une variable de type ULONG, vfloat
une variable de type FLOAT et vstring une variable de type STRING.

PRINT

Fonction d'affichage. Les données à afficher sont écrites à la suite et séparées par des virgules. Exemple :

```
print "Le résultat est :",vint/12,"\n";
```

NOT

Complément. Cette fonction peut être utilisée avec le test if pour compléter un résultat.

Exemple :

```
if not(1<2) then ...
```

ABS

Valeur absolue.

Exemple :

```
print abs(0-4); (* affiche 4 *)
```

VAL

Retourne la valeur d'une chaîne de caractères exprimée sous la forme d'un nombre décimal.

Exemple :

```
vlong=val("-123456"); (* vlong contiendra -123456 *)
```

HVAL

Retourne la valeur d'une chaîne de caractères exprimée sous la forme d'un nombre hexadécimal.

Exemple :

```
vuint=hval("abcd"); (* vuint contiendra abcd hexa *)
```

ASC

Retourne le code ASCII du premier caractère d'une chaîne.

Exemple :

```
vuint :=asc("ABCD"); (* vuint contiendra 65 : code ascii de 'A' *)
```

CHR

Retourne une chaîne composée de 1 caractère dont le code ASCII est passé en paramètre.

Exemple :

```
vstring:=chr(65); (*vstring contiendra la chaîne "A" *)
```

STRING

Retourne une chaîne composée de n caractères. Le premier argument est le nombre de caractères, le deuxième le caractère.

Exemple :

```
vstring:=string(100," ");  
(* vstring contiendra une chaîne composée de 100 espaces *)
```

STR

Convertit une valeur numérique entière en une chaîne de caractères représentant cette valeur en décimal.

Exemple :

```
vstring:=str(100); (* vstring contiendra la chaîne "100" *)
```

HEX

Convertit une valeur numérique entière en une chaîne de caractères représentant cette valeur en hexadécimal.

Exemple :

```
vstring:=hex(100); (* vstring contiendra la chaîne "64" *)
```

LEFT

Retourne la partie gauche d'une chaîne de caractères. Le premier argument est la chaîne de caractères, le deuxième le nombre de caractères à extraire.

Exemple :

```
vstring:=left("abcdef",2); (* vstring contiendra "ab" *)
```

RIGHT

Retourne la partie droite d'une chaîne de caractères. Le premier argument est la chaîne de caractères, le deuxième le nombre de caractères à extraire.

Exemple :

```
vstring:=right("abcdef",2); (* vstring contiendra "ef" *)
```

MID

Retourne une partie d'une chaîne de caractères. Le premier argument est la chaîne de caractères, le deuxième la position ou doit commencer l'extraction, le troisième le nombre de caractères à extraire.

Exemple :

```
vstring:=mid("abcdef",1,2); (* vstring contiendra "bc" *)
```

LEN

Retourne la longueur d'une chaîne de caractères.

Exemple :

```
vuint:=len("123"); (* vuint contiendra 3 *)
```

COS

Retourne le cosinus d'une valeur réelle exprimée en radian.

Exemple :

```
vfloat:=cos(3.14r); (* vfloat contiendra le cosinus de 3.14 *)
```

SIN

Retourne le sinus d'une valeur réelle exprimée en radian.

Exemple :

```
vfloat:=sin(3.14r); (* vfloat contiendra le sinus de 3.14 *)
```

TAN

Retourne la tangente d'une valeur réelle exprimée en radian.

Exemple :

```
vfloat:=tan(3.14r); (* vfloat contiendra la tangente de 3.14 *)
```

ATN

Retourne l'arc tangente d'une valeur réelle.

Exemple :

```
vfloat:=atn(0.5r); (* vfloat contiendra l'arc tangente de 0.5 *)
```

EXP

Retourne l'exponentielle d'une valeur réelle.

Exemple :

```
vfloat:=exp(1r); (* vfloat contiendra l'exponentielle de 1 *)
```

LOG

Retourne le logarithme d'une valeur réelle.

Exemple :

```
vfloat:=log(1r); (* vfloat contiendra le logarithme de 1 *)
```

LOG10

Retourne le logarithme base 10 d'une valeur réelle.

Exemple :

```
vfloat:=log10(1r);  
(* vfloat contiendra le logarithme base 10 de 1 *)
```

SQRT

Retourne la racine carrée d'une valeur réelle.

Exemple :

```
vfloat:=sqrt(2); (* vfloat contiendra la racine carrée de 2 *)
```

DATE

Retourne une chaîne de caractères représentant la date.

Exemple :

```
print "La date est :",date(),"\n";
```

TIME

Retourne une chaîne de caractères représentant l'heure.

Exemple :

```
print "L'heure est :",time(),"\n";
```

RND

Retourne un nombre aléatoire.

Exemple :

```
print rnd();
```

OPEN

Ouvre un fichier. Le premier argument est le nom du fichier, le deuxième le mode d'accès qui peut être : « r+b » ouverture en lecture / écriture, « w+b » ouverture en écriture (si le fichier existe il est détruit). La fonction retourne un long qui identifiera le fichier. Si l'ouverture échoue, la valeur retournée est 0.

Exemple :

```
vulong:=open("nouveau","w+b");
```

CLOSE

Ferme un fichier. L'argument est l'identificateur de fichier retourné par la fonction OPEN.

Exemple :

```
close(vulong);
```

WRITE

Écrit des données dans un fichier. Le premier argument est l'identificateur de fichier retourné par la fonction OPEN. Le deuxième argument est l'adresse d'une variable, le troisième le nombre d'octets à écrire. La fonction retourne le nombre d'octets effectivement écrits.

Exemple :

```
vuint:=write(vulong, &buff, 5);
```

READ

Lit des données dans un fichier. Le premier argument est l'identificateur de fichier retourné par la fonction OPEN. Le deuxième argument est l'adresse d'une variable, le troisième le nombre d'octets à lire. La fonction retourne le nombre d'octets effectivement lus.

Exemple :

```
vuint:=read(vulong, &buff, 5);
```

SEEK

Déplace un pointeur de fichier. Le premier argument est l'identificateur de fichier retourné par la fonction OPEN, le deuxième la position.

Exemple :

```
seek(vulong, 01);
```

GOTO

Effectue un saut au label passé en argument. L'argument est une chaîne de caractères.

Exemple :

```
goto "fin"
```

```
...
```

```
fin;
```

CALL

Effectue un saut à un sous-programme. L'argument est une chaîne de caractères contenant le label du sous-programme.

Exemple :

```
BEGIN;
```

```
(* programme principal *)
```

```
call "sp"
```

```
END;
```

```
BEGIN;
```

```
(* sous programme *)
```

```
sp:
```

```
print "Dans le sous-programme\n";
```

```
return;  
END;
```

RETURN

Marque la fin d'un sous-programme.

READVAR

Lit une ou plusieurs variables de l'application d'automatisme. Le premier argument est le nom de la variable d'automatisme (nom de variable ou de symbole). Le deuxième argument est l'adresse d'une variable ou d'un tableau de variables 32 bits (longs ou flottants). Le troisième argument est le nombre de variables à lire. Si la fonction est exécutée sans erreur, la valeur retournée est 0.

Exemple :

```
readvar("i0",&buff,16); (* lecture de 16 entrées à partir de i0 *)
```

WRITEVAR

Ecrit une ou plusieurs variables de l'application d'automatisme. Le premier argument est le nom de la variable d'automatisme (nom de variable ou de symbole). Le deuxième argument est l'adresse d'une variable ou d'un tableau de variables 32 bits (longs ou flottants). Le troisième argument est le nombre de variables à écrire. Si la fonction est exécutée sans erreur, la valeur retournée est 0.

Exemple :

```
writevar("o0",&buff,16);  
(* écriture de 16 sorties à partir de o0 *)
```

CMD

Exécute une commande. L'argument est une chaîne qui spécifie la commande à exécuter. Cette fonction permet d'utiliser les commandes prédéfinies d'IRIS. Pour plus de détails consultez le chapitre Ordres spéciaux. La valeur retournée est 0 si la commande a été exécutée sans erreur.

Exemple :

```
cmd("run");
```

YIELD

Passe la main. Cette fonction permet de ne pas monopoliser l'exécution lorsque l'objet est exécuté en mode prioritaire.

Exemple :

```
WHILE 1
    DO
        ...
        yield();
ENDWHILE;
```

DLL

Appelle une DLL. Le premier argument est le nom du fichier DLL. Le deuxième est le nom de la fonction. Le troisième est un pointeur sur une variable 32 bits qui recevra le code de retour de la fonction. Les autres arguments sont passés à la fonction.

Exemple :

```
dll "user", "messagebeep", &vulong, -1;
```

Messages d'erreurs

« séparateur ';' manquant »	il manque un point virgule
« erreur de syntaxe »	une erreur de syntaxe a été détectée
« variable définie plusieurs fois »	une variable est définie plusieurs fois
« pas assez de mémoire »	l'exécution du programme a saturé la mémoire disponible
« variable non définie »	une variable utilisée dans le programme n'est pas définie
« constante trop grande »	une constante est trop grande
« programme trop complexe »	une expression est trop complexe, il faut la décomposer
« type de variable ou de constante incompatible »	une variable ou une constante n'est pas du type attendu
« ')' manquant »	il manque une parenthèse fermante

« ENDIF manquant »	il manque le mot-clé ENDIF
« 'ENDWHILE' manquant »	il manque le mot-clé ENDWHILE
« label introuvable »	un label de saut ou de sous-programme est introuvable
« ']' manquant »	il manque un crochet fermant
« numéro d'élément hors borne »	un élément de tableau hors limite a été utilisé
« trop de 'CALL' imbriqués »	trop de sous-programme imbriqués ont été utilisés
« 'RETURN' trouvé sans 'CALL' »	RETURN trouvé en dehors d'un sous-programme
« taille de variable trop petite »	la taille d'une variable est insuffisante
« fichier DLL introuvable »	le fichier DLL est introuvable
« fonction introuvable dans DLL »	la fonction est introuvable dans le fichier DLL
« division par zéro »	une division par 0 s'est produite
« erreur mathématique »	une fonction mathématique a provoqué une erreur

Onglet « Aspect »

Couleurs

Permet de choisir la couleur du fond et des caractères de l'objet.

Taille de l'objet

Détermine les dimensions de l'objet en nombre de points. La modification de ces valeurs permet de régler très précisément la taille de l'objet.

Texte

Permet de spécifier un texte bulle qui apparaît lorsque le curseur est placé sur l'objet.

Onglet « Programme »

Programme

Cette zone d'édition contient le programme.

Exécuter

Si cette case est cochée, alors le programme est exécuté.

Exécution prioritaire

Si cette case est cochée, alors le programme s'exécute plus rapidement.

Exécution au démarrage

Si cette case est cochée, alors le programme s'exécute à l'ouverture de l'objet. Cette option permet de sauvegarder un objet avec l'option « Exécuter » non cochée tout en demandant une exécution de l'objet au chargement.

Aller sur l'erreur

Si une erreur a été détectée pendant l'exécution du programme, alors ce bouton poussoir permet de placer le curseur à l'endroit qui a provoqué l'erreur.

Exemples IRIS 2D

Le nom des fichiers d'exemple fait référence au sous répertoire « Exemples » du répertoire où a été installé AUTOMGEN.

Exemple d'objet composé

Cet exemple va vous permettre de comprendre comment créer un objet « Clavier décimal » composé de touches : « 0 » à « 9 » plus une touche [ENTER] pour valider.

Créez un objet « Pupitre », puis à partir du menu du pupitre créez un objet « Bouton Voyants ». Nous allons paramétrer cet objet puis nous le dupliquerons pour obtenir les autres touches. Ensuite nous retoucherons les propriétés des touches dupliquées pour les personnaliser : texte affiché sur la touche et action. Nous aurons ainsi un clavier dont l'aspect des touches sera homogène.

Le lien avec l'application sera réalisé par l'intermédiaire d'un mot.

Lorsqu'une touche sera enfoncée elle écrira son code (0 à 9 ou 13 pour la touche de validation) dans ce mot.

Pour spécifier ce mot nous pourrions donner son numéro sous la rubrique action des propriétés de chacun des objets. L'inconvénient est que lorsque nous réutiliserons l'objet « Clavier décimal » et si nous voulons utiliser un autre mot il faudra modifier les propriétés des 11 objets « Bouton voyant ».

Pour contourner ce problème nous allons utiliser la possibilité donnée aux objets enfants d'accéder à un paramètre défini dans les propriétés d'un pupitre parent. L'onglet « Liens » de la fenêtre des propriétés du pupitre permet de définir le paramètre. Ecrivez dans la zone d'édition sur une seule ligne : « CLAVIER=M200 ». Cette ligne signifie que le paramètre clavier est égal à M200.

Les touches du clavier feront référence au paramètre « CLAVIER » et non directement au mot M200. Ainsi pour changer le mot utilisé il suffira de modifier la définition du paramètre dans les propriétés du pupitre.

Revenons à l'aspect de notre clavier ...

Pour que l'aspect du clavier soit satisfaisant nous allons définir une grille pour aligner les touches. Dans la fenêtre des propriétés du pupitre et l'onglet « Options » écrivez la valeur « 10 » sous les deux rubriques « Grilles ». Ainsi la fonction déplacer du menu du pupitre utilisera une grille de 10 pixels. Nous allons également fixer les dimensions de la première touche. Nous pourrions directement modifier les dimensions de la touche en la saisissant par un de ses bords, cependant pour plus de précision nous allons modifier directement les dimensions sous la rubrique « Taille de l'objet en pixel » de l'onglet « Aspect » de la fenêtre des propriétés de l'objet « Bouton Voyant ».

Entrez par exemple « 30 » pour la largeur et la hauteur.

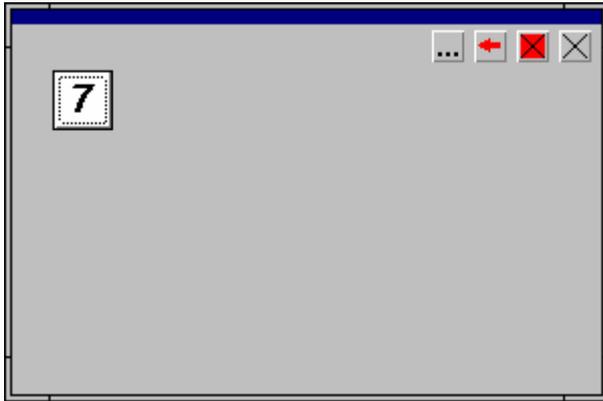
A ce stade, vous pouvez également personnaliser le style de la touche : la couleur, la fonte utilisée pour le marquage, etc ...

Nous placerons cette première touche en haut et à gauche du clavier (c'est un choix arbitraire). Le clavier que nous allons créer ressemblera au pavé numérique d'un clavier d'ordinateur. Nous allons donc marquer cette touche avec le texte « 7 » sous la rubrique « Texte » de l'onglet « Aspect ».

Nous allons également paramétrer l'aspect fonctionnel de la touche : sous la rubrique « Action lorsque le bouton est enfoncé » de l'onglet « Liens » nous allons écrire: « PARENTPARAM(CLAVIER)=7 ». Ce qui signifie que lorsque le bouton poussoir sera enfoncé le mot désigné par le paramètre « CLAVIER » du pupitre parent recevra la valeur 7. Effacez ce qui se trouve sous la rubrique « Action lorsque le bouton est relâché ».

Nous pouvons également affecter à l'objet « Bouton Voyant » une touche du clavier de l'ordinateur. Ainsi il sera possible d'utiliser le clavier avec la souris ou le clavier de l'ordinateur. Pour affecter une touche à l'objet « Bouton Voyant », utilisez la rubrique « Touche » de l'onglet « Options ». Entrez par exemple « 7 » pour associer la touche « 7 » du clavier de l'ordinateur à l'objet.

Placez ensuite la touche « 7 » en haut et à gauche du clavier, comme ceci :



Pour déplacer cette touche sélectionnez d'abord l'objet (touche [SHIFT] enfoncée puis cliquez avec le bouton gauche de la souris sur l'objet), puis utilisez la fonction « Déplacer » du menu du pupitre. Cette fonction est la seule qui utilise la grille contrairement au déplacement en saisissant la barre des objets enfants.

Pour créer les autres touches, dupliquez la touche existante :

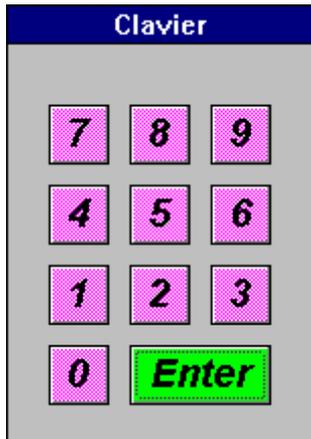
- sélectionnez la première touche,
- choisissez « Copier » dans le menu du pupitre, puis « Coller »,
- déplacez la touche ainsi collée,
- paramétrez la nouvelle touche : (texte, liens et touche du clavier de l'ordinateur).

Lorsque vous avez terminé la rangée du haut (touches « 7 », « 8 » et « 9 ») vous pouvez sélectionner ensemble ces trois touches et les dupliquer.

Vous pouvez créer une touche de validation (plus large pour remplir la surface du clavier).

Pour terminer, redimensionnez le pupitre, passez les objets en mode « Exploitation ».

Le résultat final doit ressembler à ceci :



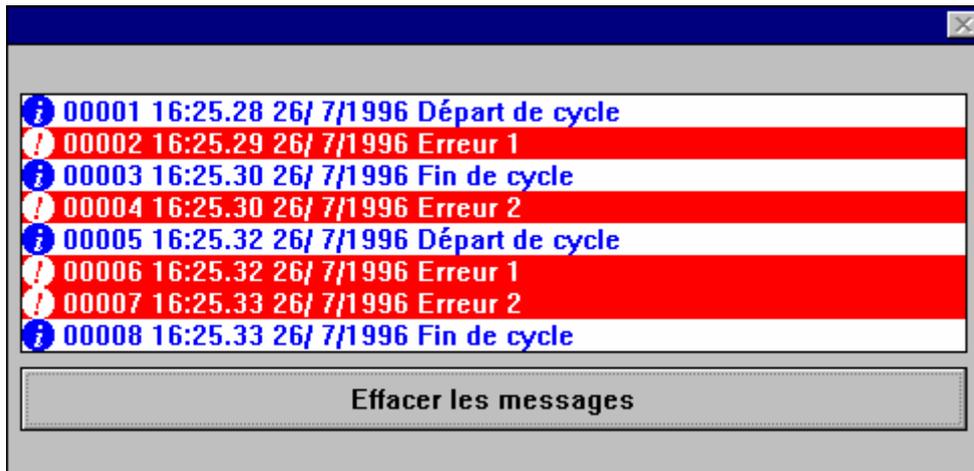
📁 « Exemples\IRIS2D\clavier.agn »

Exemple d'utilisation de l'objet « Ecran, clavier, liste à messages » comme liste à messages

Cahier des charges :

- l'objet doit afficher quatre messages différents suivant l'état de quatre entrées (i0 à i3),
- pour l'entrée 0 : un message d'information « Départ de cycle »,
- pour l'entrée 1 : un message d'information « Fin de cycle »,
- pour l'entrée 2 : un message d'erreur « Erreur 1 »,
- pour l'entrée 3 : un message d'erreur « Erreur 2 ».
- les messages doivent être affichés à l'apparition du front montant des entrées,
- un historique de 50 messages sera conservé dans l'objet et sauvegardé sur disque,
- les messages seront dupliqués sur une imprimante connectée sur « LPT1 : »,
- un bouton poussoir doit permettre d'effacer les messages.

Solution :



☞ « Exemples\IRIS2D\écran clavier.agn »

Variante :

L'appui sur le bouton poussoir « Effacer les messages » provoque l'ouverture d'une boîte de dialogue « Voulez vous effacer les messages » avec un choix OUI ou NON.

Solution :

☞ « Exemples\IRIS2D\Ecran clavier 2.agn »

Exemple d'utilisation de l'objet ECRANCLA comme terminal

Cahier des charges :

Afficher un message « Entrez une valeur », attendre une valeur décimale tapée au clavier (deux caractères) puis afficher cette valeur multipliée par deux derrière le texte « Résultat : ».

Solution :



☞ « Exemples\IRIS2D\terminal 1.agn »

Variante :

Les messages affichés sont stockés dans l'objet et non plus dans l'application d'automatisme.

Solution :

📁 « Exemples\IRIS2D\terminal 2.agn »

Exemple d'application composée de plusieurs pages

Cet exemple va vous permettre de comprendre comment créer une application composée de plusieurs éléments : dans ce cas particulier un menu qui permet d'accéder à deux pages différentes.

📁 « Exemples\IRIS2D\menu.agn »

Exemple d'utilisation de l'objet OBJET

Simulation d'un vérin.

Cahier des charges :

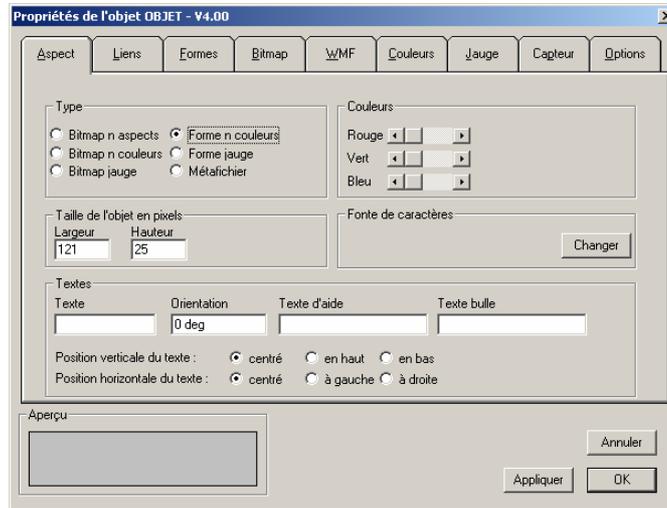
- vérin piloté par deux sorties o0 (sortir le vérin) et o1 (rentrer le vérin),
- deux entrées de fin de course i0 (vérin rentré) et i1 (vérin sorti).

Trois objets seront utilisés :

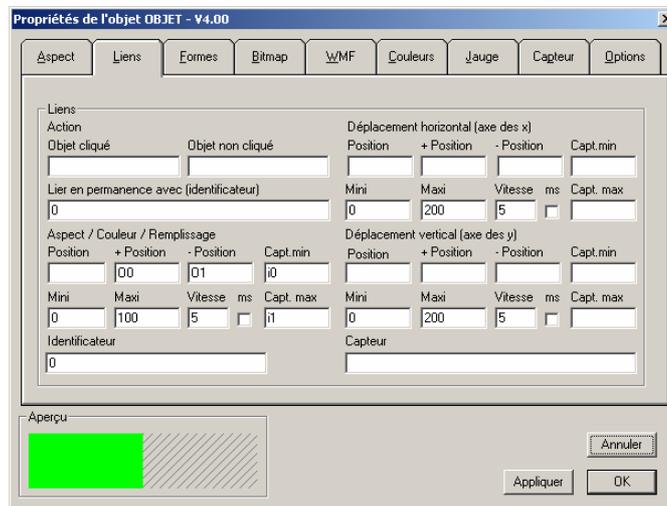
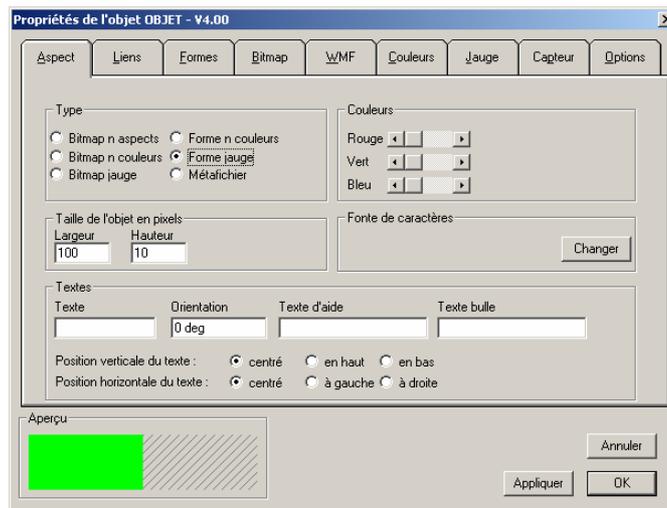
- un objet « Pupitre » servant de support,
- un objet « Objet » servant de corps de vérin,
- un objet « Objet » servant de tige de vérin.

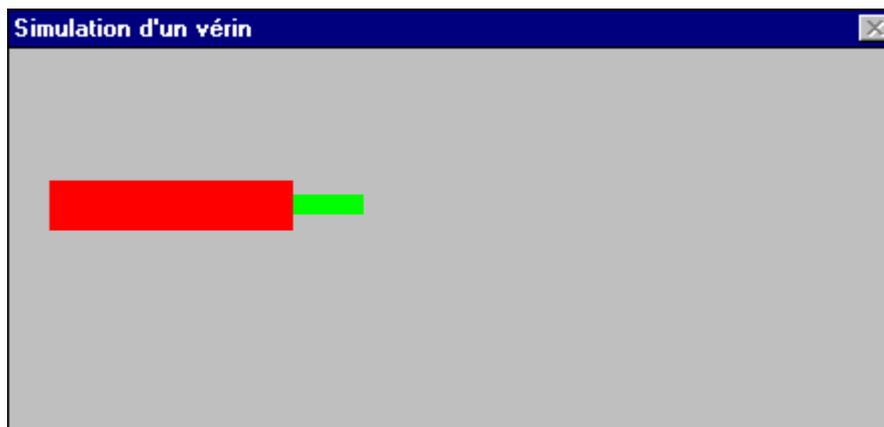
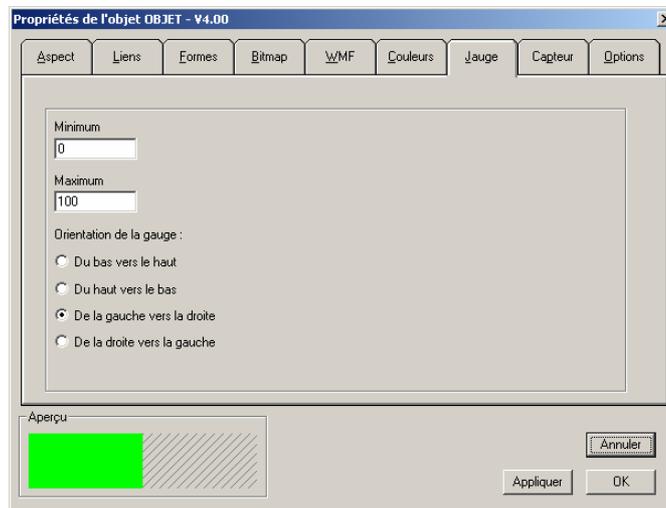
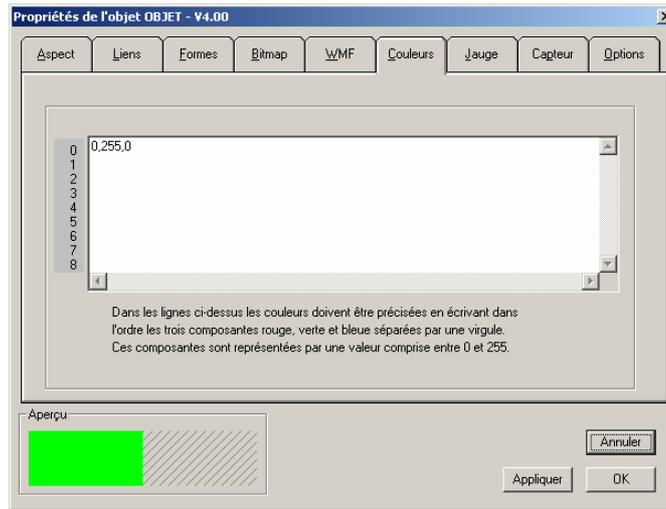
Solution :

Le corps du vérin est un objet OBJET qui reste statique, seul son aspect est configuré :



La tige du vérin est un objet OBJET configuré de la façon suivante :





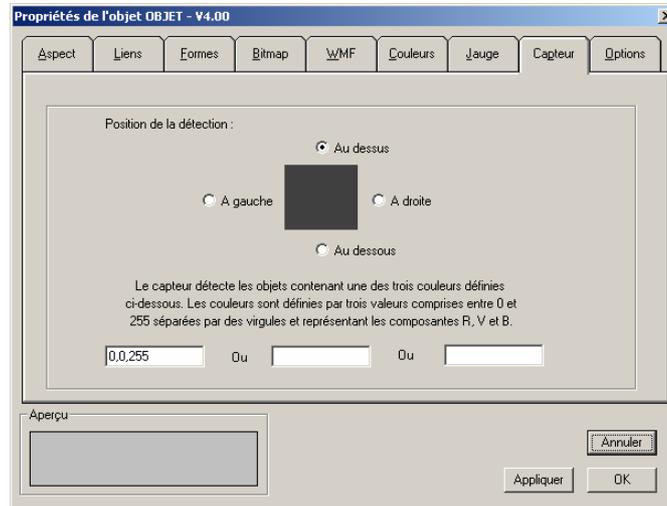
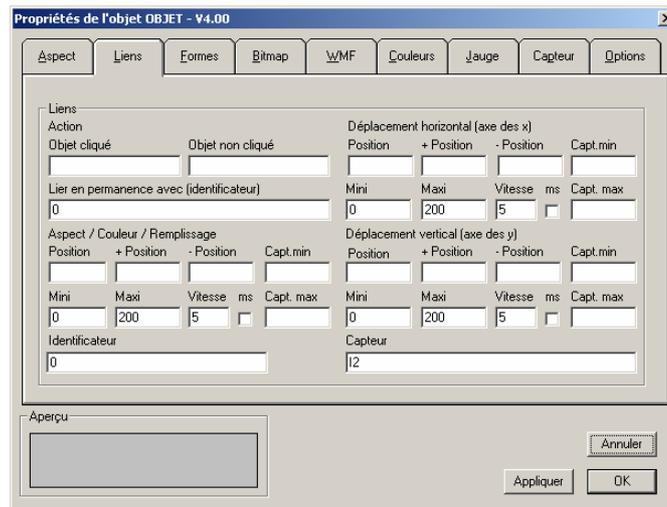
« Exemples\Simulation PO\2D\tutorial1.agn »

Variante :

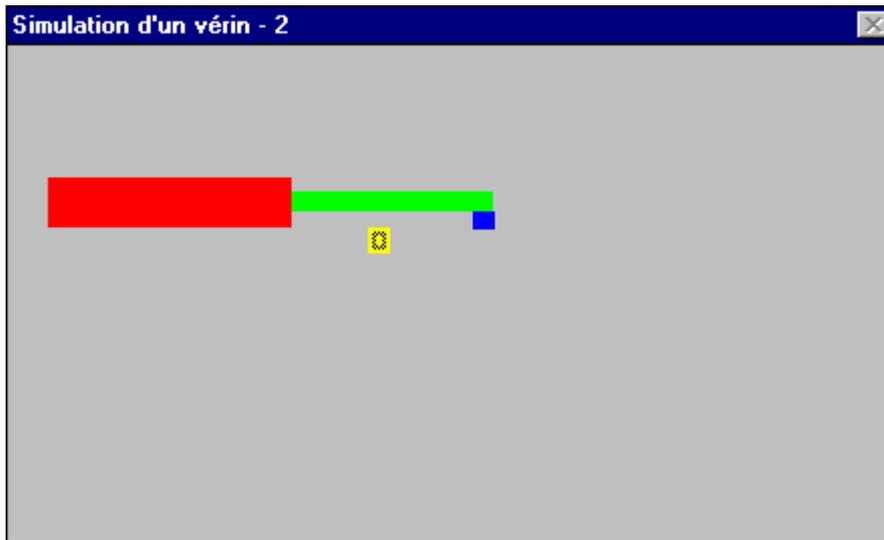
Une position intermédiaire doit être ajoutée sur le vérin. Pour cela nous allons utiliser deux objets supplémentaires : une pièce fixée sur la tige du vérin qui actionnera un capteur et un capteur.

Pour lier la pièce actionnant le capteur à la tige du vérin il faut associer à la tige du vérin un identificateur : sous la rubrique « Identificateur » de l'onglet « Liens » écrire « 100 ». Pour lier la pièce à la tige, écrire sous la rubrique « Déplacement horizontal, Position » de l'onglet « Liens » : « `SISTERPARAM(100,STATE)` ». Ceci a pour effet de lier la pièce avec l'état de la tige du vérin.

L'objet utilisé comme capteur est paramétré comme suit :



Le résultat est le suivant :

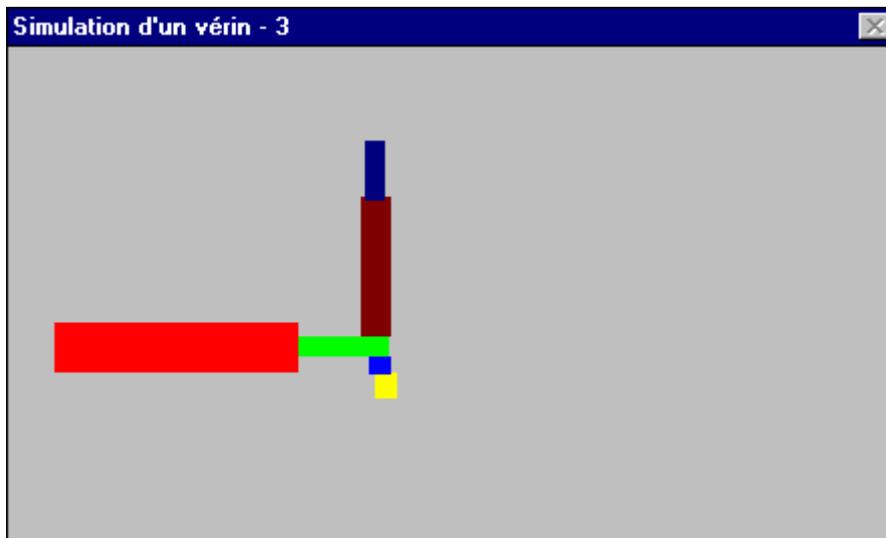


« Exemples\Simulation PO\2D\tutorial2.agn »

Deuxième variante :

Un vérin vertical fixé sur la tige du vérin horizontal est ajouté. Ce vérin est actionné par une seule sortie ($O2=1$ pour sortir le vérin, $O2=0$ pour le rentrer). Deux fins de course sont associés $i3$ et $i4$.

Le résultat est le suivant :



« Exemples\Simulation PO\2D\tutorial3.agn »

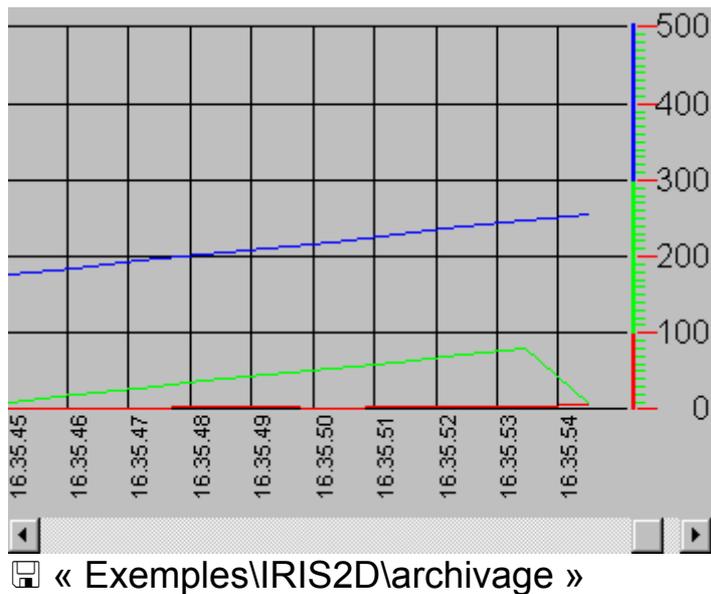
Deux objets OBJET sont ajoutés : un pour le corps du vérin et un pour la tige.

Exemples d'utilisation de l'objet ARCHIVE

Cahier des charges :

- archiver l'état de 3 mots de l'application d'automatisme (m31 à m33) toutes les secondes,
- l'état des 4 mots sera affiché sur une courbe laissant apparaître 10 secondes d'acquisition,
- 1000 valeurs seront mémorisées dans l'objet,
- les acquisitions seront archivées dans un fichier « data.txt » au format texte.

Solution :



Exemple d'utilisation de l'objet PROG

Cahier des charges :

- l'appui sur un bouton poussoir doit provoquer l'inversion de l'état des sorties O0 à O99.

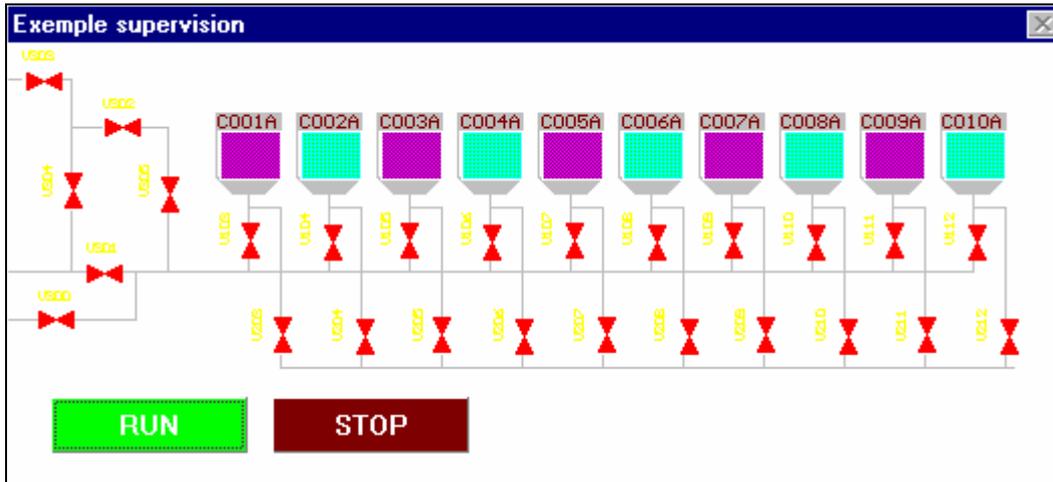
Solution :

« Exemples\IRIS2D\programme.agn »

Exemples d'application de supervision 1

L'exemple suivant illustre la création d'une application de supervision. L'application de supervision affichera l'état de vannes et de niveaux de cuves. L'action de l'utilisateur sur les vannes aura pour effet d'inverser l'état de la vanne (ouverte ou fermée). L'état RUN/STOP de l'application d'automatisme sera également affiché et deux boutons poussoirs permettront de passer en RUN ou en STOP.

Le résultat est le suivant :



« Exemples\IRIS2D\supervision 1 »

Des objets OBJET seront utilisés pour représenter les vannes. Un fichier bitmap est créé pour représenter les vannes : état ouvert (couleur verte) et état fermée (couleur rouge) :



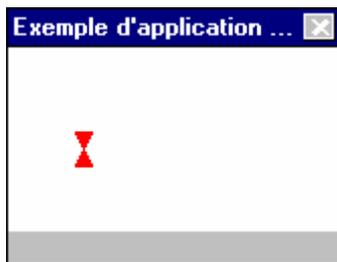
Exemple d'application de supervision 2

Cet exemple illustre une utilisation plus évoluée d'un objet OBJET. L'application affiche l'état d'une vanne qui peut être :

- vanne ouverte (ouverture commandée et capteur vanne ouverte vrai) : couleur verte,
- vanne fermée (fermeture commandée et capteur vanne fermée vrai) : couleur rouge,
- vanne en cours d'ouverture (ouverture commandée et capteur vanne ouverte faux) : couleur bleue,
- vanne en cours de fermeture (fermeture commandée et capteur vanne fermée faux) : couleur violette.

L'utilisateur peut inverser l'état de la vanne en cliquant dessus.

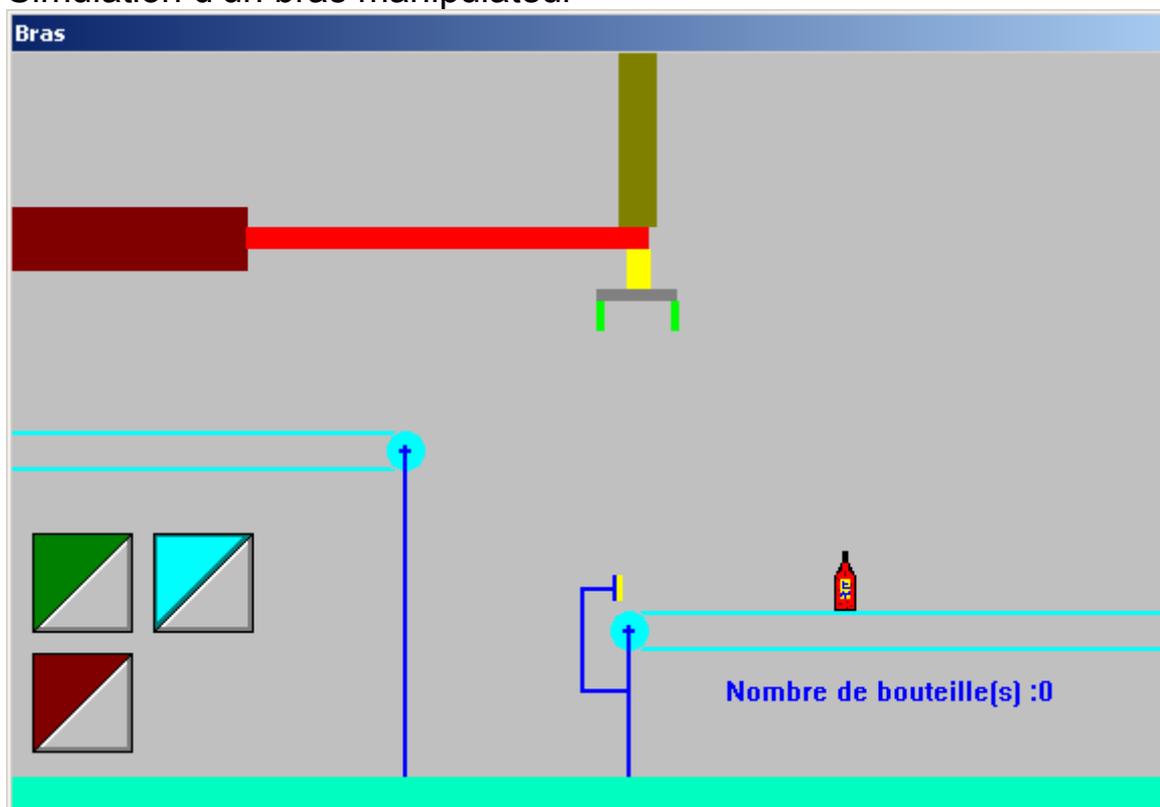
L'application d'automatisme gère l'état de la vanne.



« Exemples\IRIS2D\supervision 2.agn »

Exemple de simulation d'une partie opérative 1

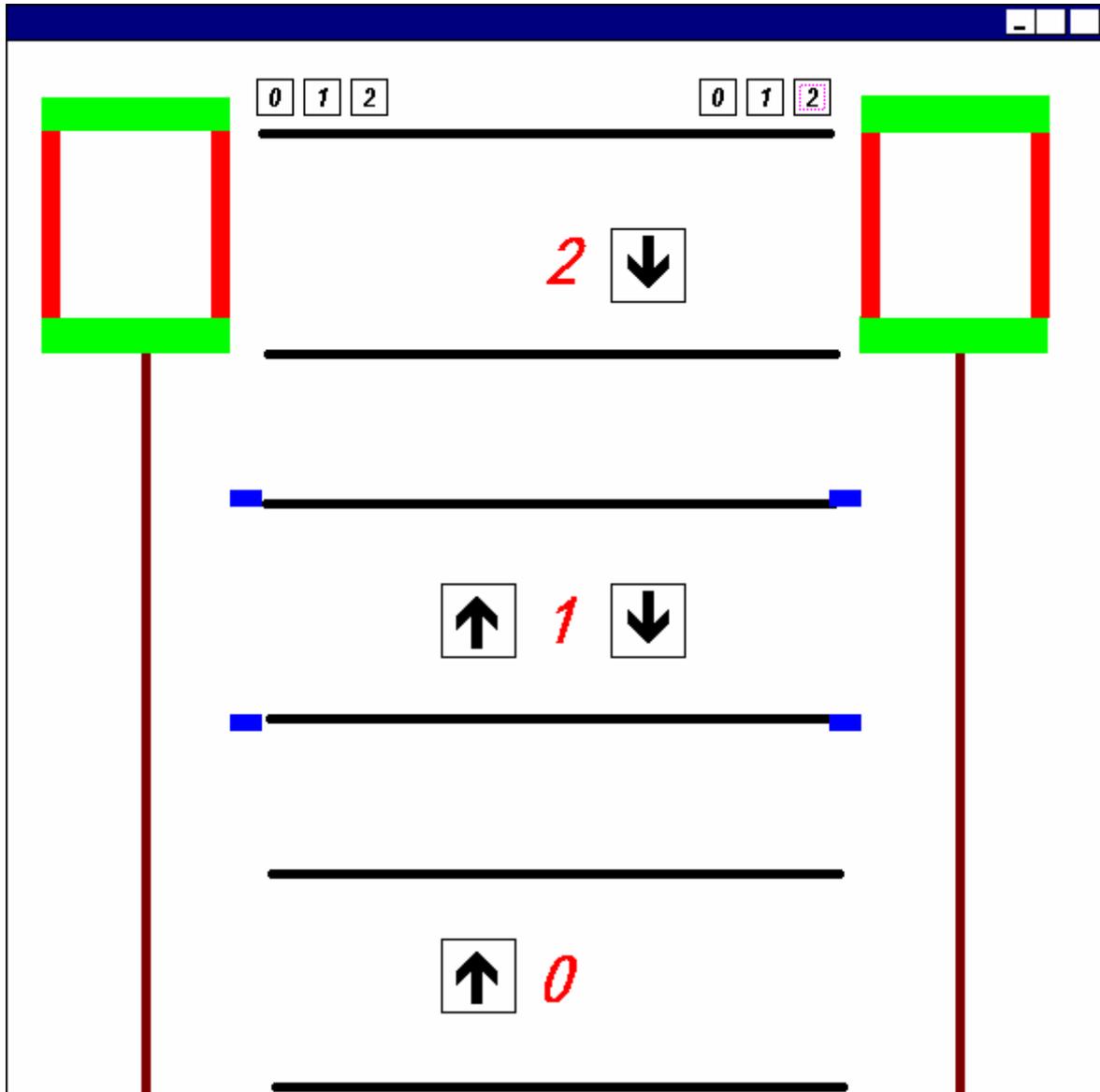
Simulation d'un bras manipulateur



« Exemples\Simulation PO\2D\bras manipulateur.agn »

Exemple de simulation d'une partie opérative 2

Simulation d'un ascenseur



« Exemples\Simulation PO\2D\ascenseur.agn »



Les objets IRIS 2D permettent de créer des applications de supervision et de simulation de parties opératives 2D.

Références IRIS 3D

IRIS 3D vous permet de créer des applications de simulation de parties opératives 3D.

IRIS 3D permet d'animer des objets 3D créés dans des modeleurs standards : 3D STUDIO, SOLIDWORKS, SOLIDCONCEPTER, etc ...

Le format natif des fichiers traité par IRIS 3D est celui des fichiers « .X » défini par DIRECTX 8 de Microsoft.

Un convertisseur du format « .3DS » vers « .X » est intégré à l'environnement.

L'utilitaire CROSSROADS fourni sur le CD-ROM d'installation d'AUTOMGEN⁷ ou téléchargeable sur www.irati.com permet de convertir un nombre important de format de fichier 3D vers le format « .3DS ».

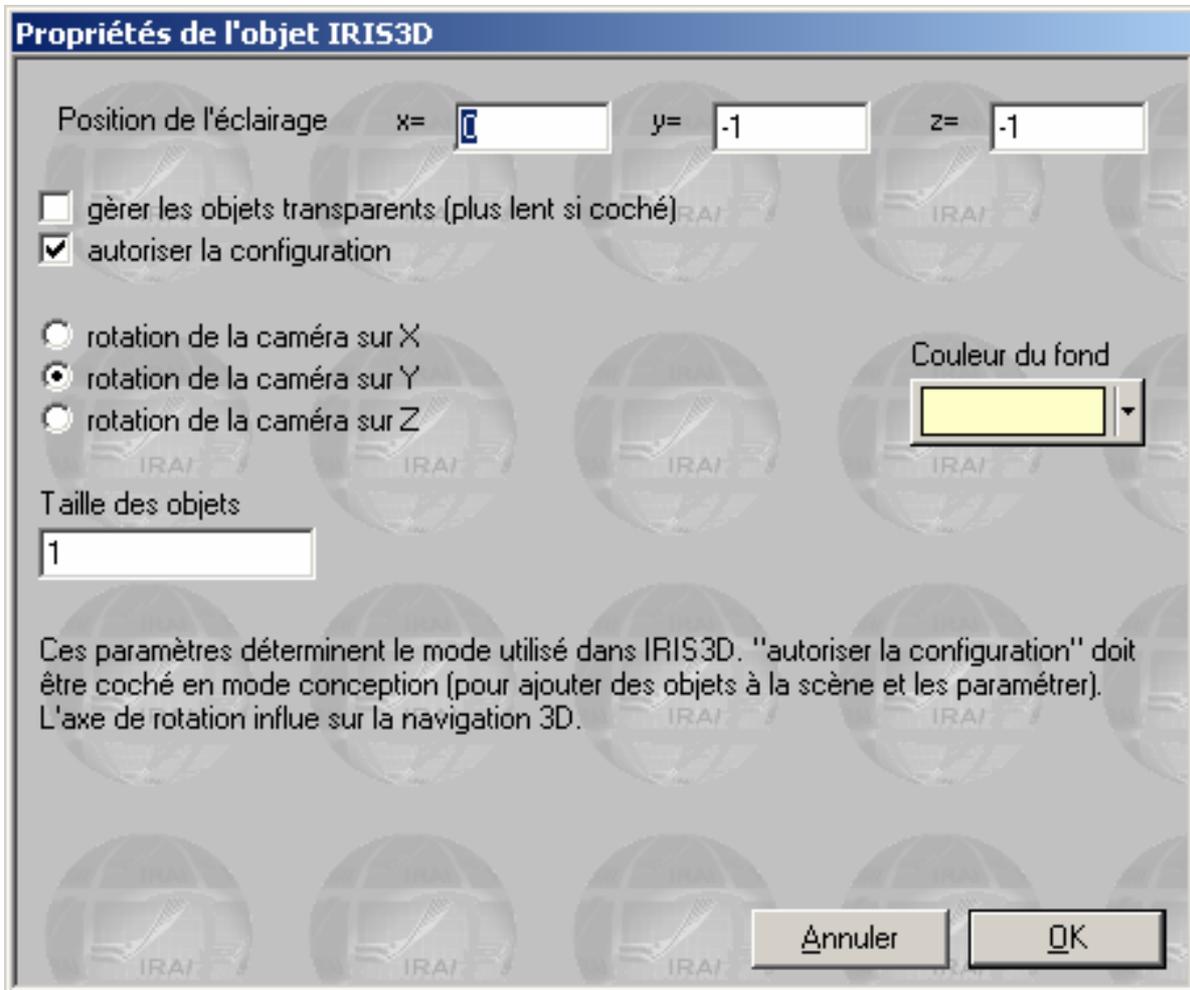
IRIS 3D se présente sous la forme d'une fenêtre encapsulée dans un pupitre IRIS 2D. Sur ce pupitre viendront s'animer les objets 3D.

Chaque fichier 3D représentera un objet dans IRIS 3D. Les éléments d'une partie opérative devant avoir un mouvement propre, devront être représentés par des fichiers séparés. Par exemple, pour un vérin composé d'un corps et d'une tige, il faut créer un fichiers pour le corps du vérin et un pour la tige du vérin.

Pour créer l'animation des objets dans le monde 3D, un ou plusieurs comportements peuvent être appliqués à chacun des objets. Un comportement se compose d'une modification de l'objet (déplacement, changement de couleur, etc ...) et d'un lien avec les variables de l'application d'automatisme pour conditionner cette modification. Par exemple : sortir la tige du vérin si la sortie 1 de l'application d'automatisme est vraie.

Créer un pupitre IRIS 3D

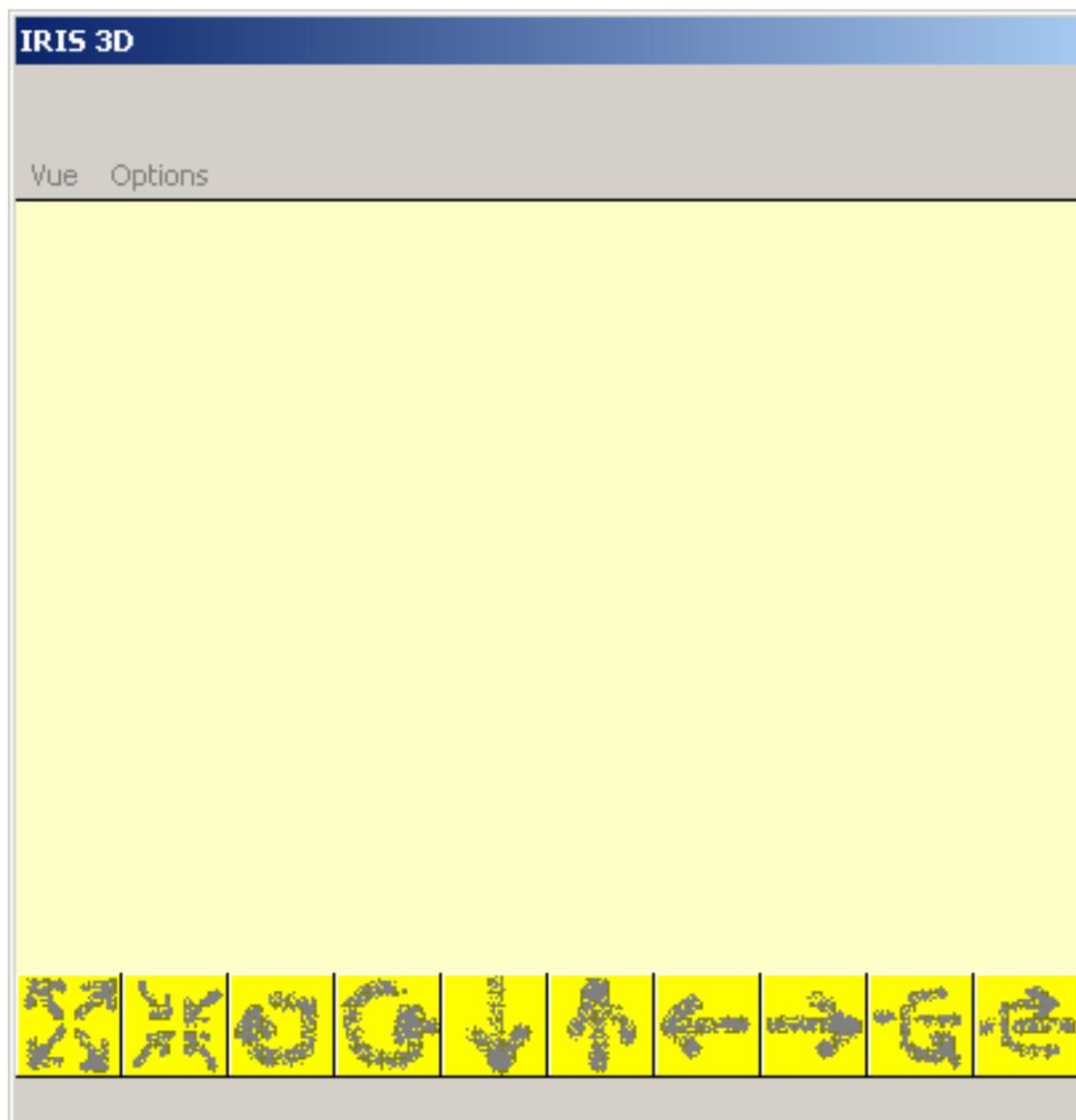
Cliquez avec le bouton droit de la souris sur l'élément « Iris » dans le navigateur et choisissez l'option « Ajouter un pupitre IRIS 3D ».



Création d'un pupitre IRIS 3D

Ajouter des fichiers 3D au projet

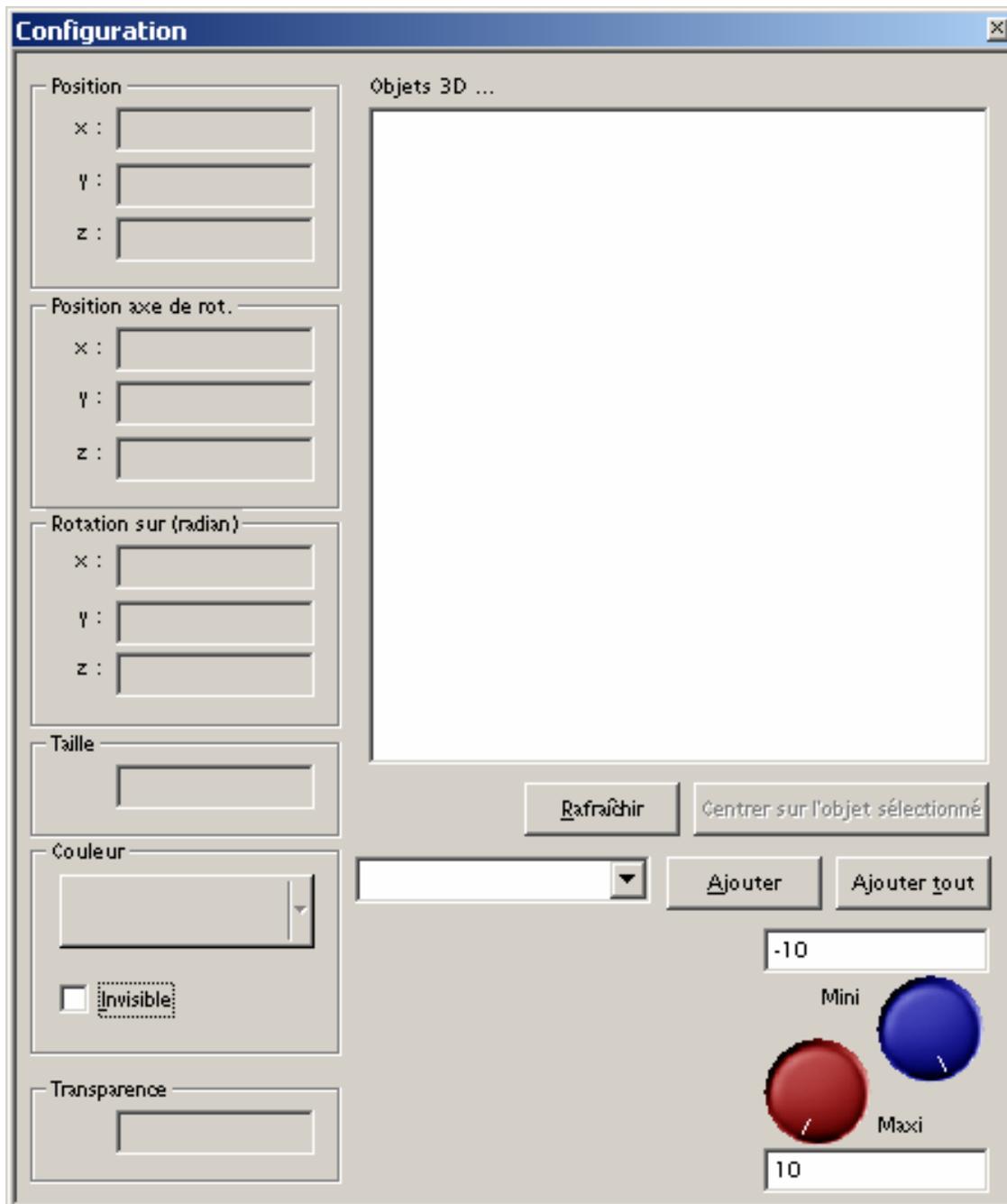
Cliquez avec le bouton droit de la souris sur l'élément « Ressources » dans le navigateur et choisissez « Importer un ou des fichiers 3D » dans le menu. Sélectionnez un ou plusieurs fichiers « .3DS ». (si vos fichiers ne sont pas au format « .3DS », utilisez l'utilitaire « CROSSROAD » pour les convertir).



Le pupitre IRIS 3D

Configurer les objets

Choisissez « Ouvrir la fenêtre de configuration » dans le menu « Options » de la fenêtre IRIS 3D.



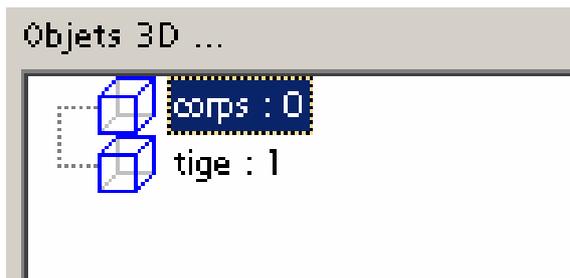
La fenêtre de configuration d'IRIS 3D

Ajouter des objets dans le monde 3D

En cliquant sur l'élément  vous accédez à la liste des objets 3D présents dans les ressources. Par exemple :



En sélectionnant un objet dans cette liste et en cliquant sur « Ajouter » vous ajouter l'objet sélectionné au monde 3D. En cliquant sur « Ajouter tout » vous ajouter l'ensemble des objets de la liste au monde 3D. Les objets ainsi ajoutés apparaissent dans la liste de la fenêtre de configuration.



Enlever un fichier 3D des ressources

Cliquez avec le bouton droit de la souris sur le fichier 3D dans le navigateur et choisissez « Effacer ». L'objet doit également être effacé du monde 3D.

Enlever un objet du monde 3D

Cliquez avec le bouton droit de la souris sur l'objet dans la fenêtre de configuration d'IRIS 3D et choisissez « Effacer » dans le menu.

Appliquer un comportement à un objet

Cliquez avec le bouton droit de la souris sur l'objet dans la fenêtre de configuration d'IRIS 3D et choisissez « Ajouter ... » dans le menu.

Nom des variables AUTOMGEN

Les noms de variables AUTOMGEN utilisables dans les comportements sont limités aux syntaxes suivantes :

Accès aux variables booléennes

On : sortie « n », par exemple O0, O10,
/On : complément de la sortie « n », par exemple /O1, /O15,
In : entrée « n », par exemple I0, I4,
/In : complément de l'entrée « n », par exemple /I4, /I56,
Bn : bit « n », par exemple B100, B200,
/Bn : complément du bit « n », par exemple /B800, /B100,

L'accès aux bits B est limité à une table de bits linéaire, une directive #B doit être utilisée pour réserver des bits (voir le manuel de référence langage).

Accès aux variables numériques

Mn : mot « n », par exemple : M200, M300
Fn : flottant « n », par exemple : F200, F400

Ajouter une translation

Translation

tige : 1

Position

Pas de pilotage
 Pilotage bistable
 Pilotage monostable
 Pilotage numérique

Mini 0.000000
 Maxi -10.000000
 Temps pour la course en ms 1000.000000

Axe

X
 Y
 Z

Détection

Capteur mini
 Capteur maxi
 Autre capteur
 Autre capteur
 Autre capteur
 Autre capteur

0.000000 0.000000
 0.000000 0.000000
 0.000000 0.000000
 0.000000 0.000000

Annuler

OK

Propriétés d'une translation

Nom

La première zone permet d'entrer un nom générique pour la translation. Ce nom apparaîtra dans la liste de la fenêtre de configuration d'IRIS 3D, il sert uniquement de commentaire et peut être laissé vierge.

Axe

Détermine dans quelle dimension va s'appliquer la translation.

Type

- pas de pilotage : aucune translation, ceci permet de rendre inopérante une translation sans avoir besoin de l'effacer (pour faire des essais par exemple).
- pilotage bistable : deux variables booléennes : la translation est pilotée par deux variables booléennes : la première pilote la translation dans un sens (du mini vers le maxi), la deuxième dans l'autre sens (du maxi vers le mini).

Etat de la première variable	Etat de la deuxième variable	Objet
0	0	Immobile
1	0	Translation du mini vers le maxi
0	1	Translation du maxi vers le mini
1	1	Immobile

- pilotage monostable : une variable booléenne pilote la translation, si la variable est vraie

Etat de la variable	Objet
1	Translation du mini vers le maxi
0	Translation du maxi vers le mini

- pilotage numérique : la position de l'objet sur l'axe désigné est égale au contenu de la variable numérique spécifiée.

Amplitude et origine

Les zones « Mini » et « Maxi » détermine l'amplitude et l'origine de la translation.

Vitesse

Le temps pour la course détermine la vitesse pour aller du point mini au point maxi (elle est identique à la vitesse de retour).

Détection

Permet de définir des capteurs pour la translation. Les capteurs mini et maxi gèrent les fins de courses, les 4 autres capteurs possibles peuvent être utilisés pour créer des positions intermédiaires.

Ajouter une rotation

Les paramètres sont en tout point semblable à la translation voir chapitre Ajouter une translation. Les angles sont exprimés en radians.

Le centre de rotation de l'objet doit être réglé pour chaque objet dans la fenêtre de configuration d'IRIS 3D.

Ajouter un changement de couleur

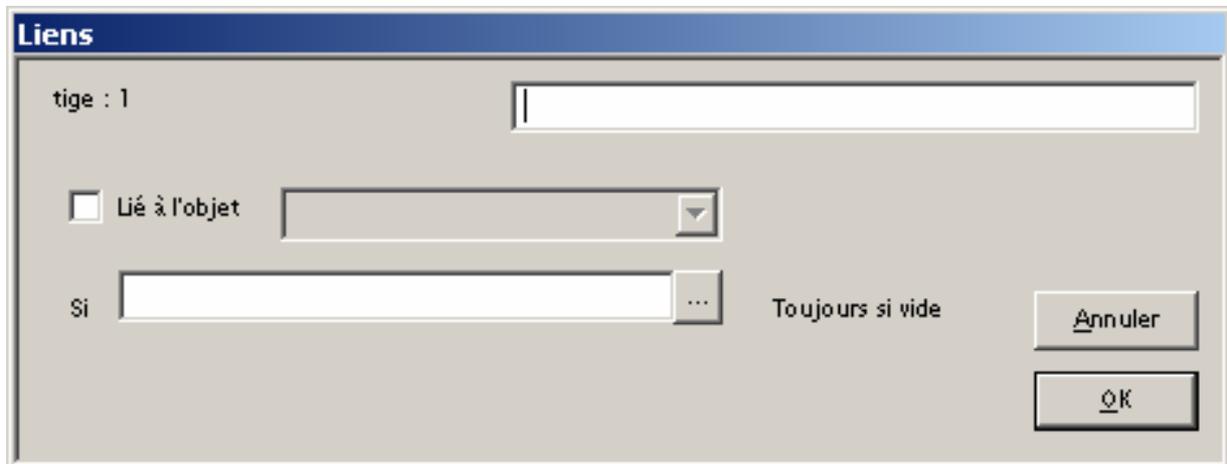


Changement de couleur

Le pilotage d'une couleur par une variable doit faire référence à une variable booléenne.

Ajouter un lien

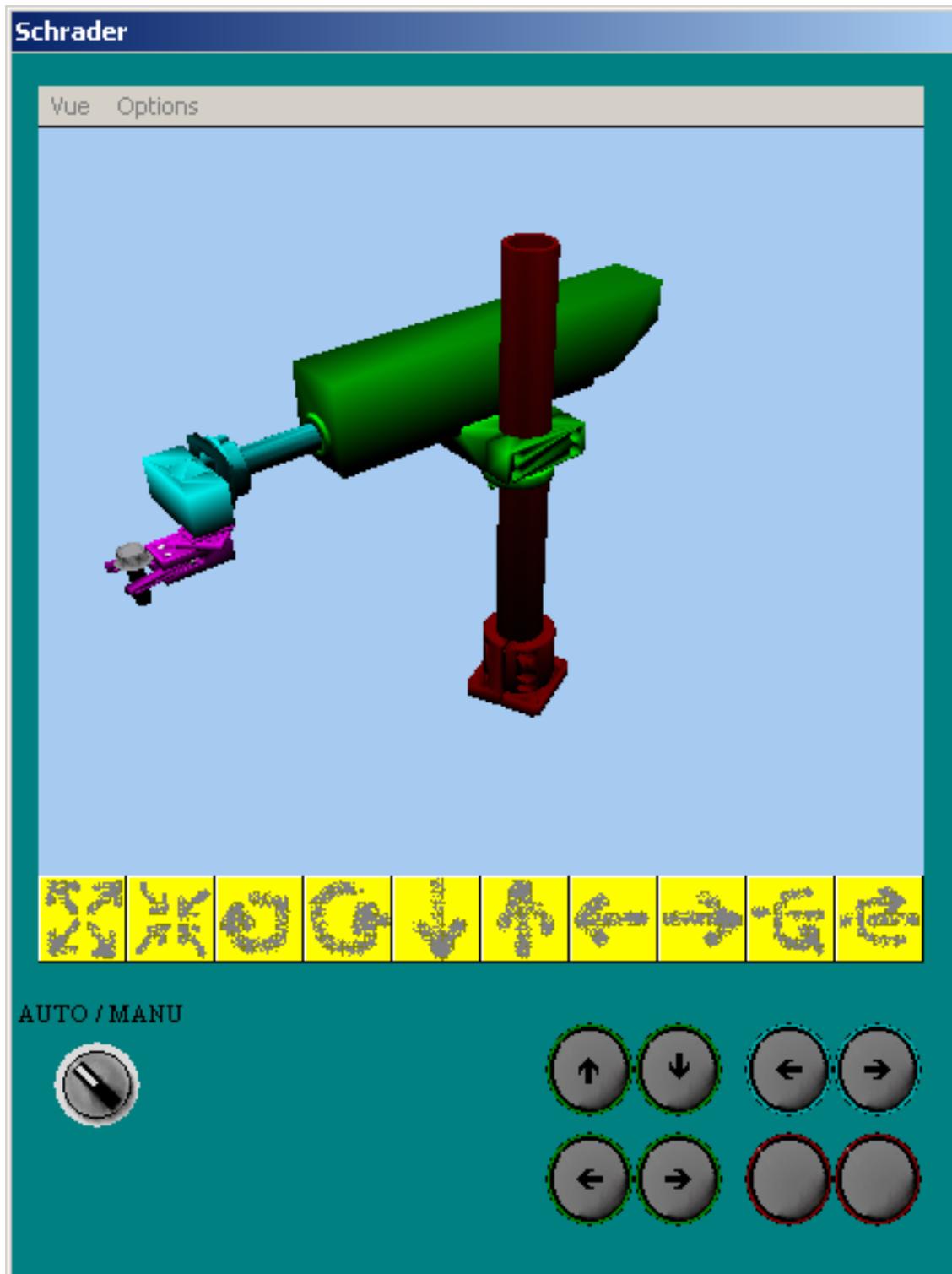
Un lien force l'objet auquel s'applique ce comportement à suivre les déplacements d'un autre objet.



Liens entre objet

La condition du lien peut être une variable booléenne. Le lien est inconditionnel (objet toujours lié) si la condition est laissée vierge.

Exemple IRIS 3D



« Exemples\Simulation PO\3D\Schrader.agn »



IRIS 3D permet de concevoir des applications de simulation de parties opératives en 3D. Les objets doivent être créés dans un modéleur standard et importés dans les ressources du projet AUTOMGEN. Des comportements sont ensuite appliqués aux objets pour créer les animations 3D.