

Concepts de langage de programmation graphique

Objectif

Être capable de créer un Instrument virtuel, différencier la face avant du diagramme

Bases théoriques : Les bases de Labview, le VI

Outil : Labview

Synthèse et validation :

Être capable de recréer en autonomie les modèles proposés.

Concepts de langage de programmation graphique

Semestre: 2

Unité d'enseignement: UEM 1.2

**Matière: Concepts et langage de Programmation
graphique**

VHS: 37h30 (Cours: 1h30, TP: 1h00)

Crédits: 3

Coefficient: 2

Concepts de langage de programmation graphique

Objectifs de l'enseignement

Ce cours permettra à l'étudiant de se familiariser avec l'environnement de programmation graphique LabVIEW et avec les fonctionnalités LabVIEW de base,

Construire des applications d'acquisition de données et de contrôle d'instruments.

Concepts de langage de programmation graphique

Connaissances préalables recommandées

L'étudiant devra posséder les connaissances suivantes :

une connaissance d'un environnement informatique est préférable.

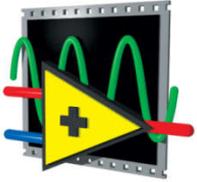
Pas de connaissance en programmation nécessaire

Introduction

Les programmes LabVIEW sont appelés instruments virtuels, ou VIs,

LabVIEW contient une grande gamme d'outils pour l'acquisition, l'analyse, l'affichage et l'enregistrement des données,

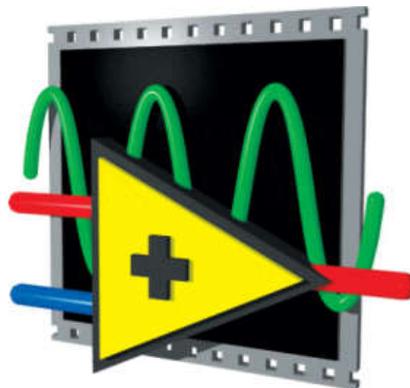
ainsi que des outils pour vous aider à mettre au point votre programme.



Introduction

LabVIEW

(Laboratory Virtual Instrument Engineering Workbench)
est un environnement de développement graphique qui permet de créer des applications modulaires (notion de sous VI) et extensibles, pour la conception d'applications de mesure, de contrôle et de test





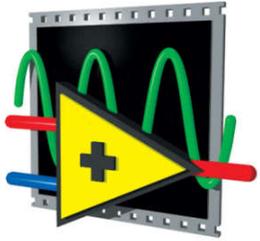
Concepts de langage de programmation graphique

National Instruments (NI)

Société à l'origine de la programmation graphique et, par conséquent, de LabVIEW

NI est classé parmi les 100 meilleurs entreprises américaines.





Concepts de langage de programmation graphique

Historique

1986 LabVIEW 1.0 pour Macintosh.

1992 LabVIEW pour Windows

2000 LabVIEW 6i Applications internet

labVIEW 2009 : éditeur d'icône amélioré, graphes 3D,

LabVIEW 2017 : LabVIEW 2017 ; LabVIEW NXG 1.0

LabVIEW 2018 : LabVIEW NXG 3.0

LabVIEW 2019, première version avec formalisme pour

Sets and Maps



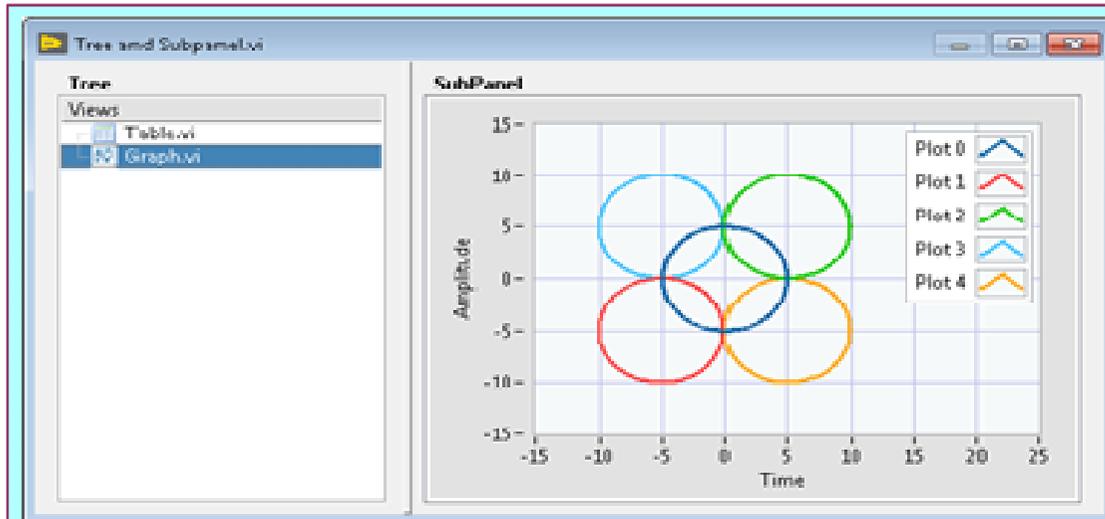
Vocabulaire LabVIEW

Les programmes LabVIEW appelés Instruments Virtuels ou Virtual Instruments (VI).

On parle d'instruments virtuels car leur apparence et leur fonctionnement sont semblables à ceux d'instruments réels, tels que les oscilloscopes et les multimètres.

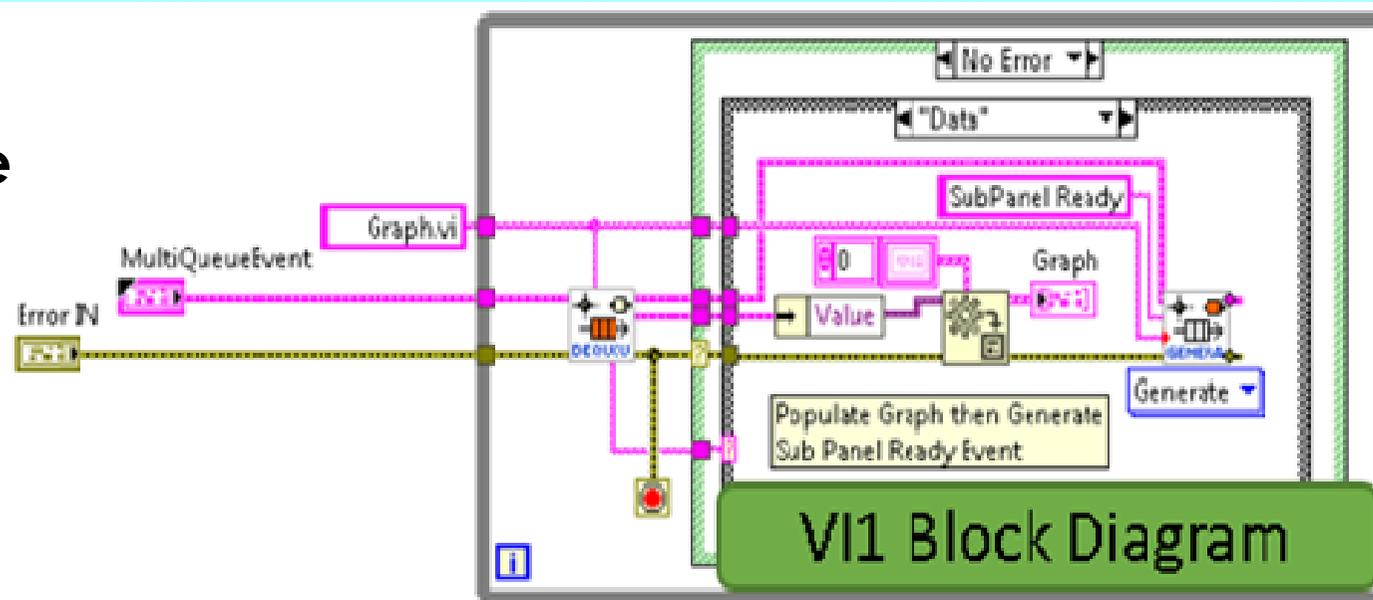


Vocabulaire LabVIEW



Face avant

Diagramme





Composantes d'un VI

Les VI se composent de trois éléments principaux :

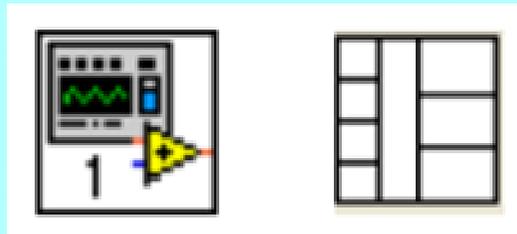
Face avant

- Interface utilisateur «Contrôles » = entrées
«Indicateurs » = sorties

Diagramme

- Fenêtre de programmation et d'affichage du code source
- Interaction entre face avant et diagramme : Ctrl+E

-Icône/Connecteur





Face avant d'un VI (Front panel)

Dans la **face-avant** nous trouverons tous les éléments interactifs du VI (commandes et indicateurs).

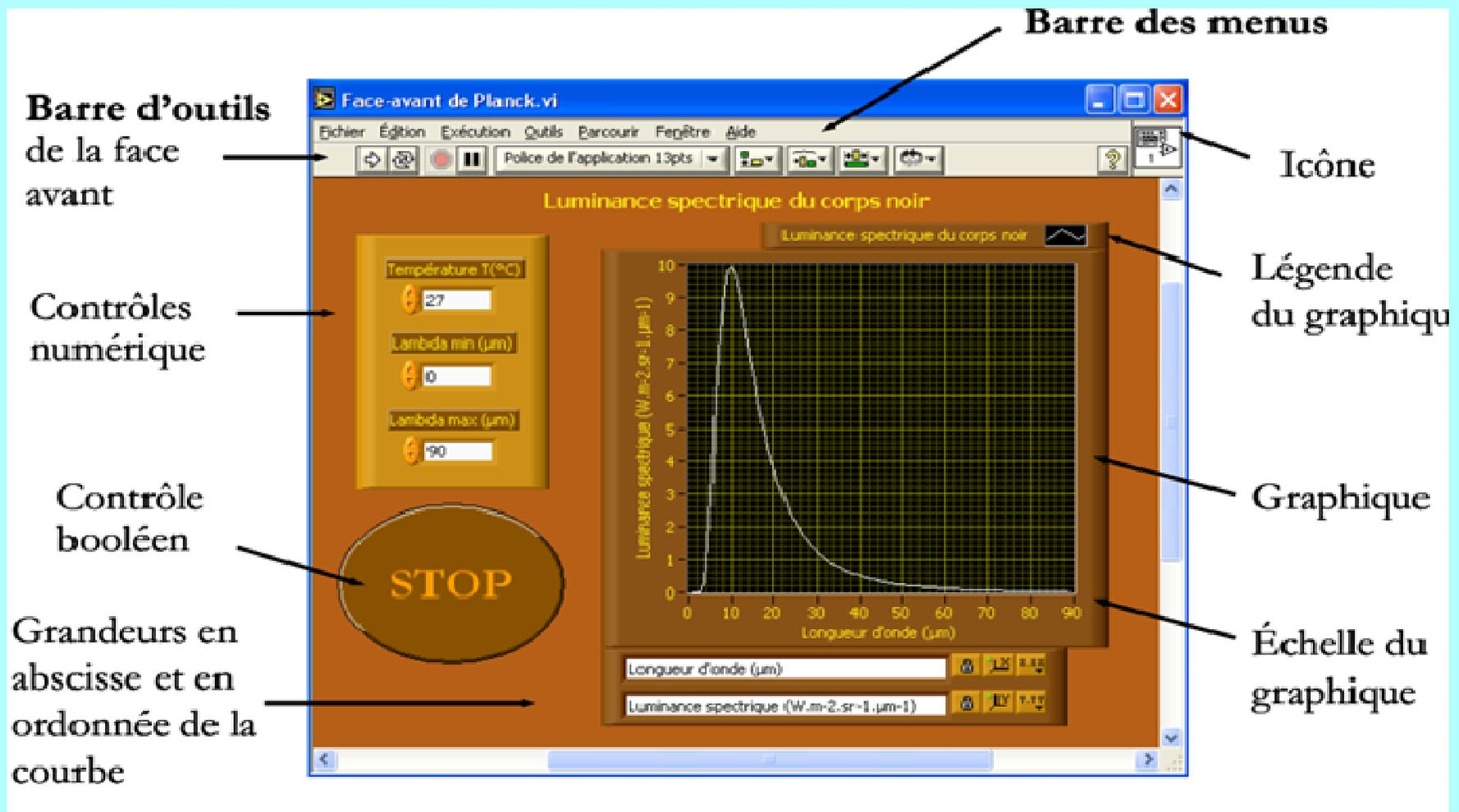




Diagramme d'un VI

Dans le diagramme nous trouverons tous les éléments propres au code développé (structures, fonctions, constantes,...).

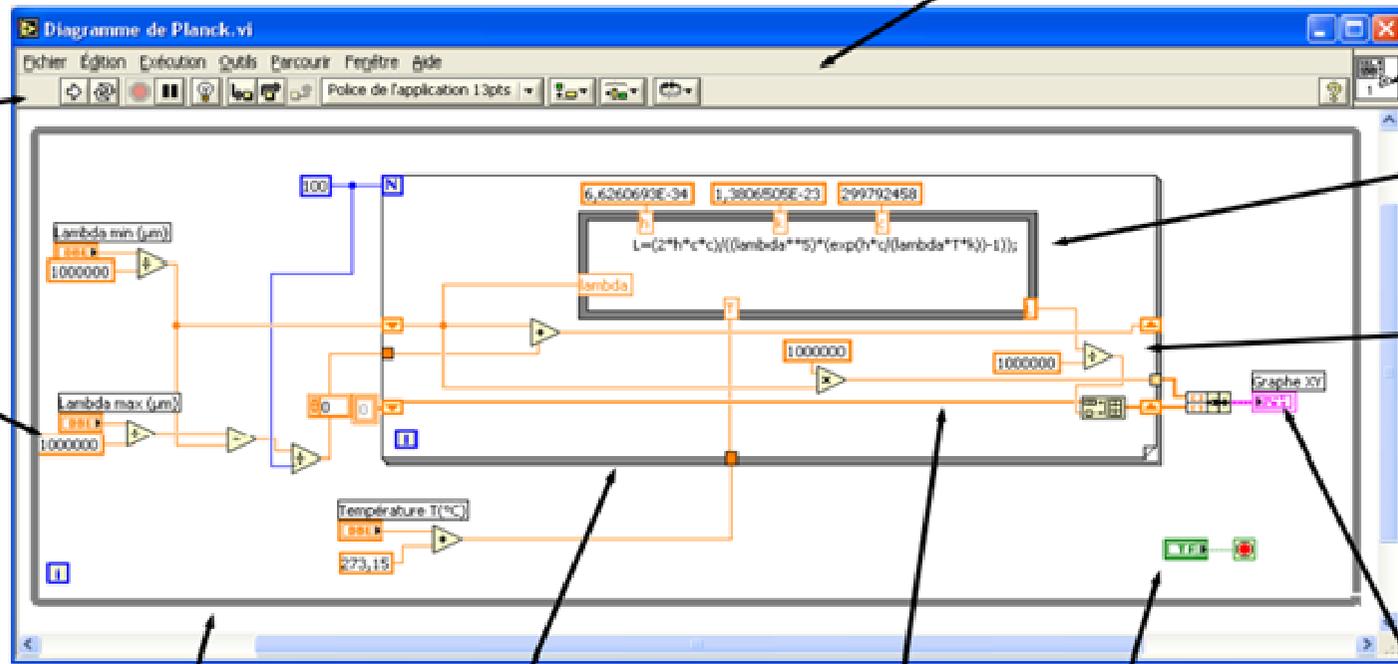
Barre d'outils du diagramme

Icône

Boîte de calcul

Fonction division

Constante numérique



Structure d'une boucle While (tant que)

Structure d'une boucle For

Fil de données

Terminal de contrôle booléen

Terminal graphique



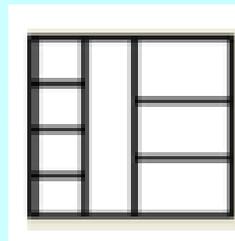
Icône/connecteur d'un VI

Chaque VI affiche une icône, dans le coin supérieur droit des fenêtres de la face-avant et du diagramme. Une icône est une représentation graphique d'un VI qui permet de l'identifier au sein d'un autre VI.



Icône par défaut

Un connecteur est un ensemble de terminaux correspondant aux commandes et aux indicateurs du VI qui sont accessibles.

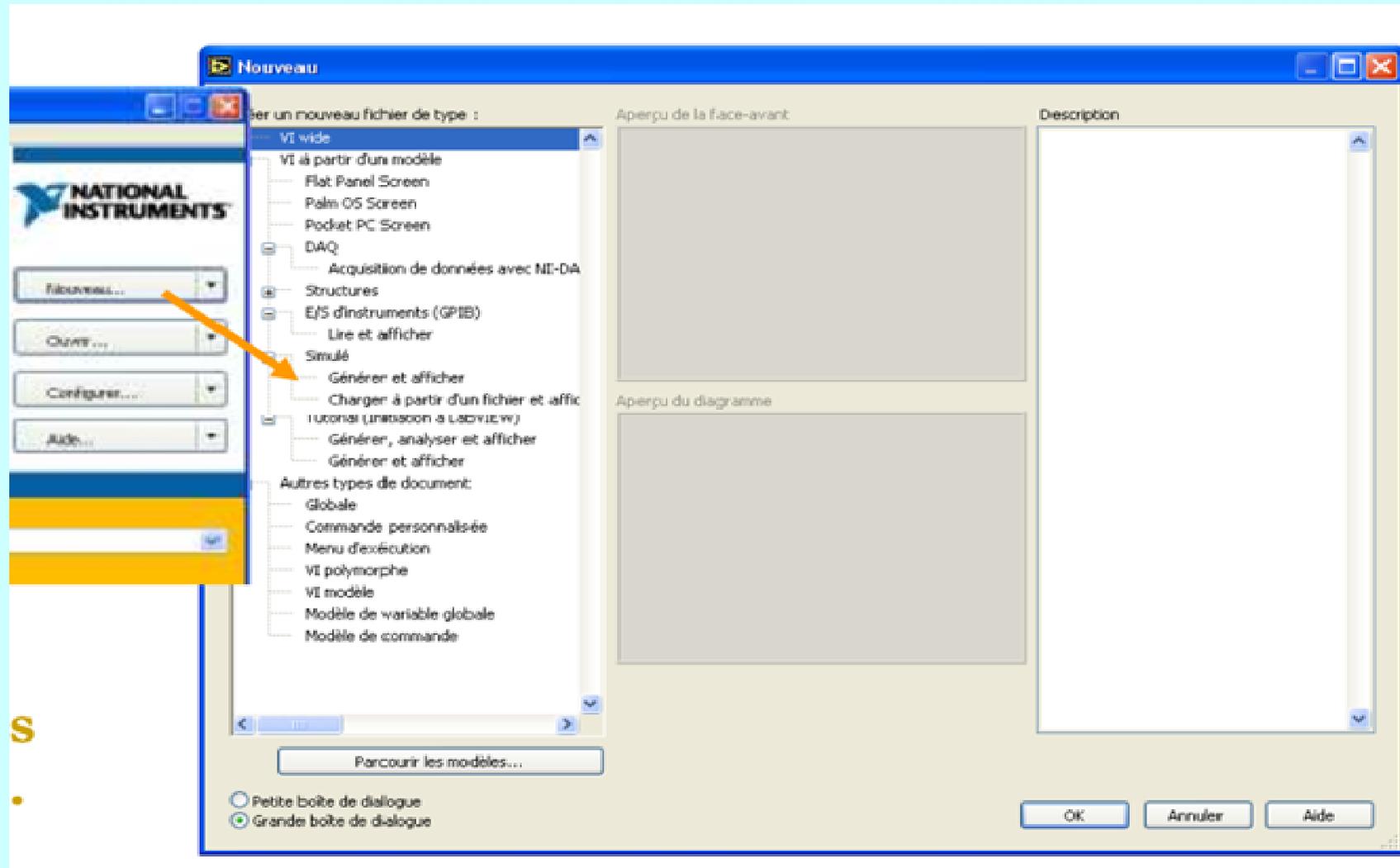




Environnement LabVIEW

Ouvrir un VI : Modèles de VI

Des modèles de VI déjà pré-codés sont disponibles.





Environnement LabVIEW

Ouvrir un VI :

Ouvrir un VI

Modèles de VI pour assistants personnel (PDA)

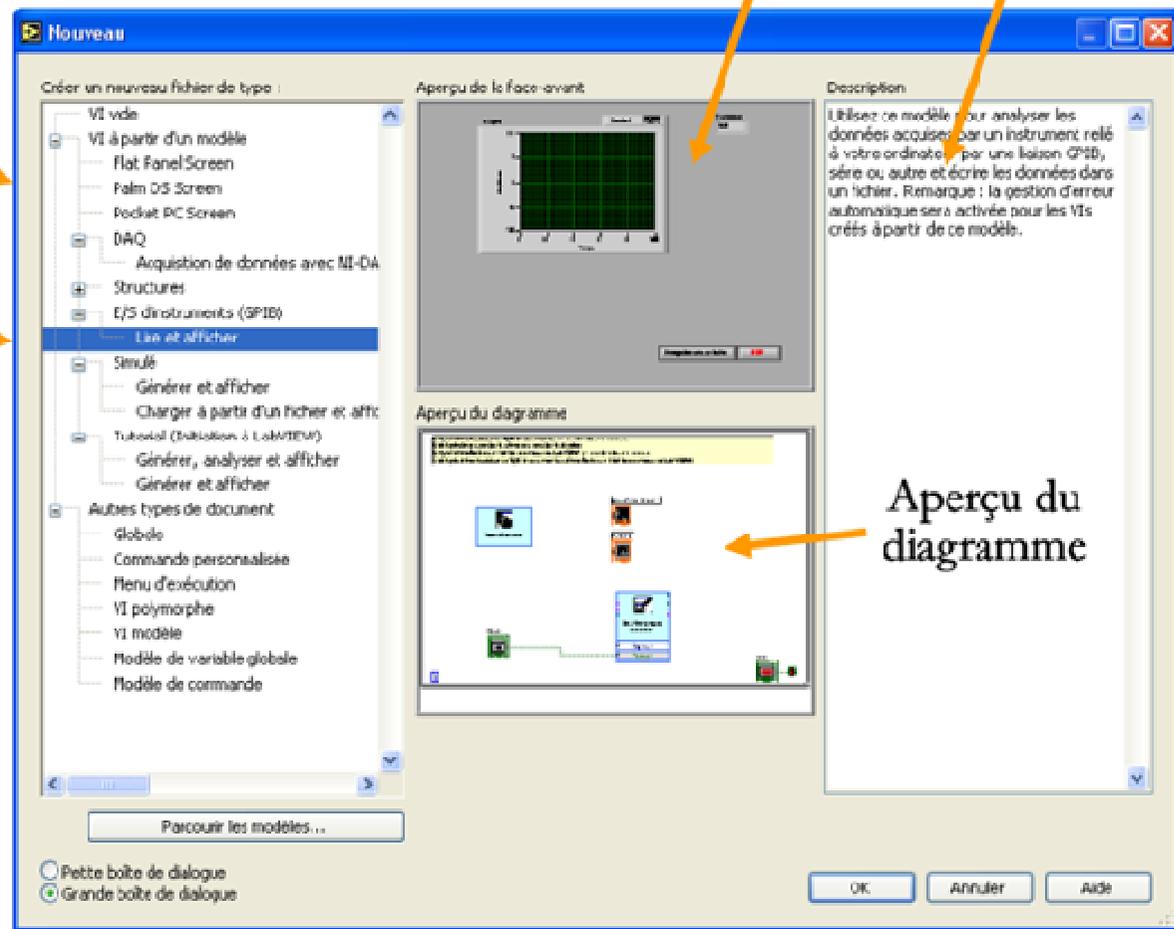
Modèles de VI pour communications par GPIB

Permet d'avoir une trame simple et fonctionnelle rapidement

Modèles de VI

Aperçu de la face avant

Description du VI

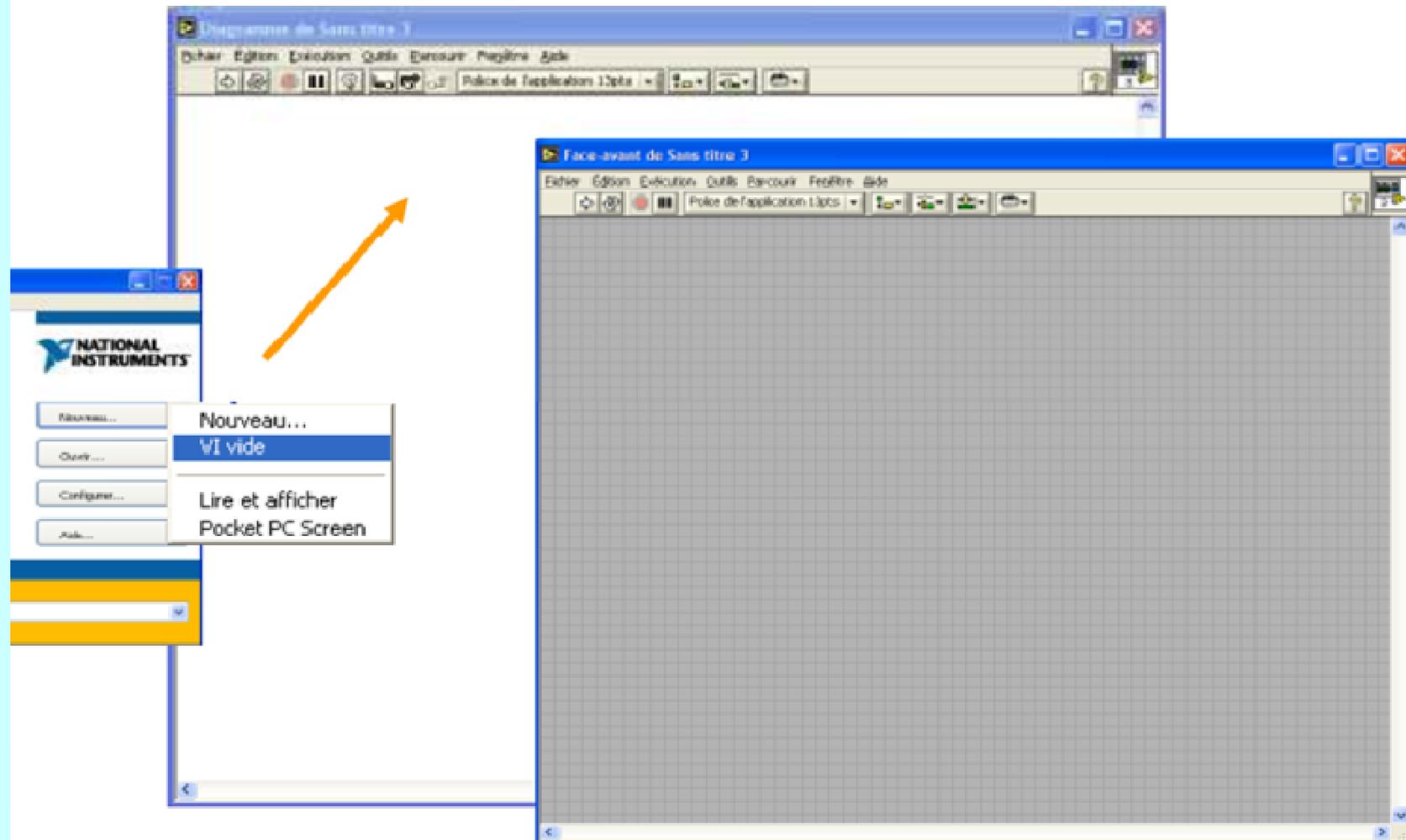




Environnement LabVIEW

Ouvrir un VI :

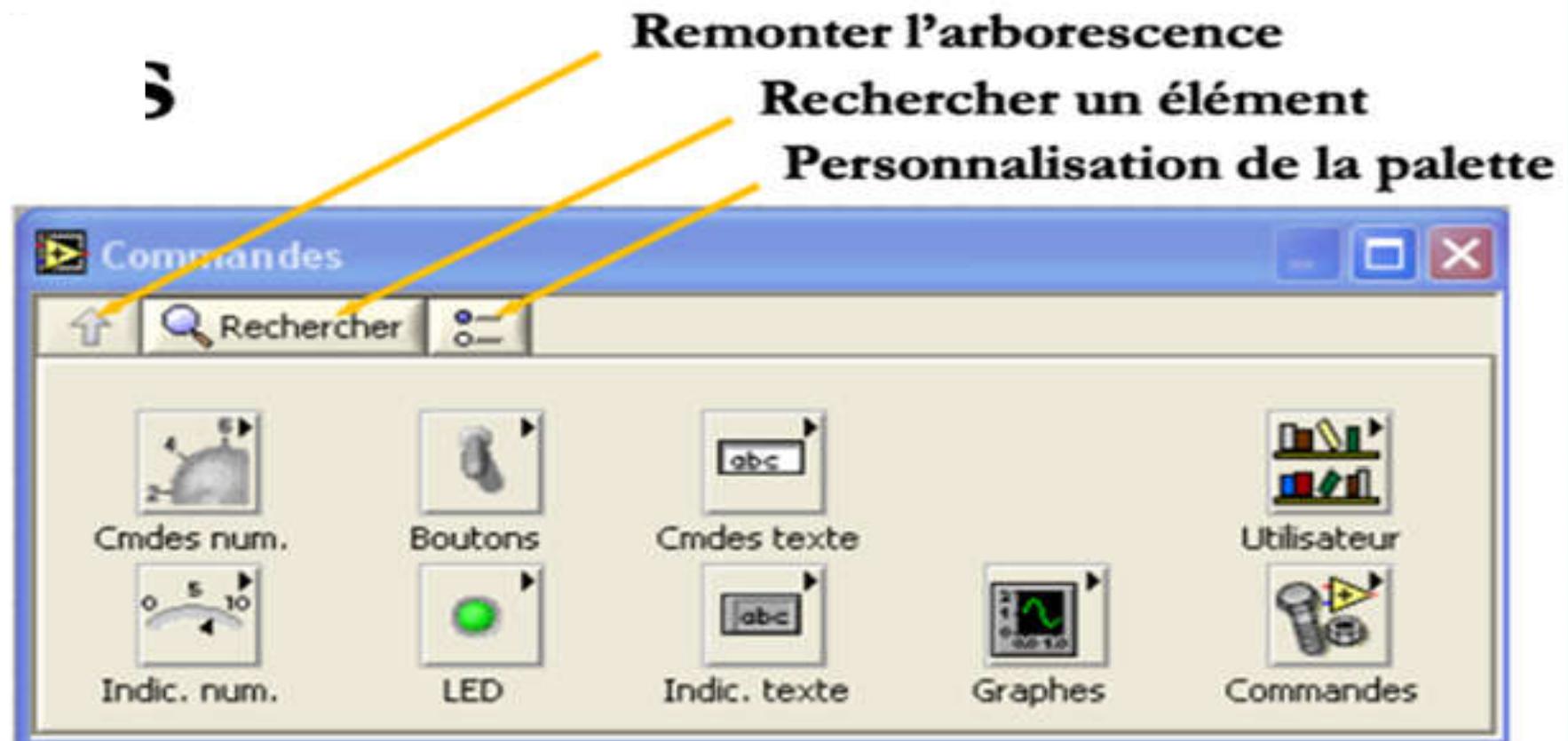
VI vide





Palettes de commandes

Palette de **commandes** (disponible à partir de la fenêtre **face avant** par un clic droit avec la souris ou dans la barre des menus : “Fenêtre”).

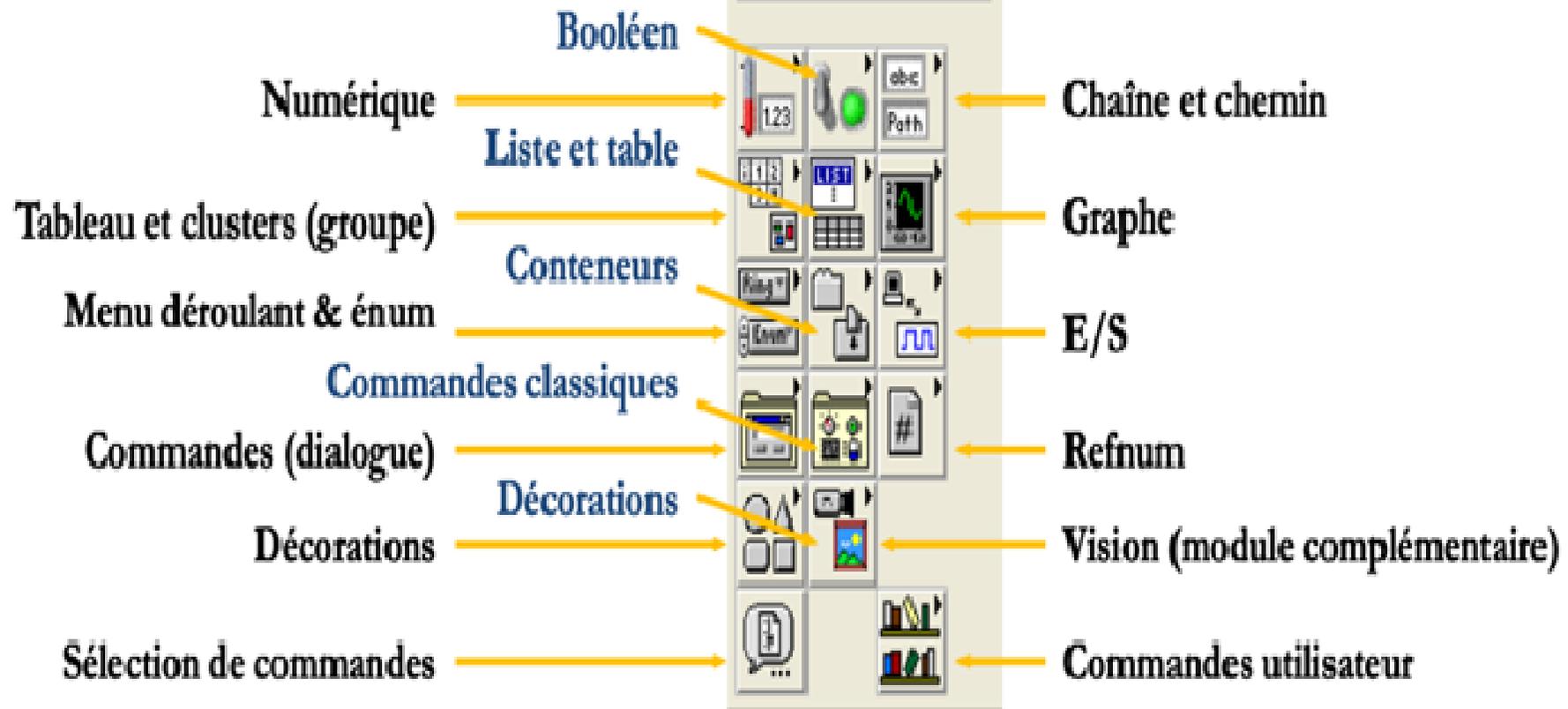




Palettes de commandes

Dans cette palette nous trouverons tous les éléments nécessaires à la création de la **face-avant** (commandes et indicateurs).

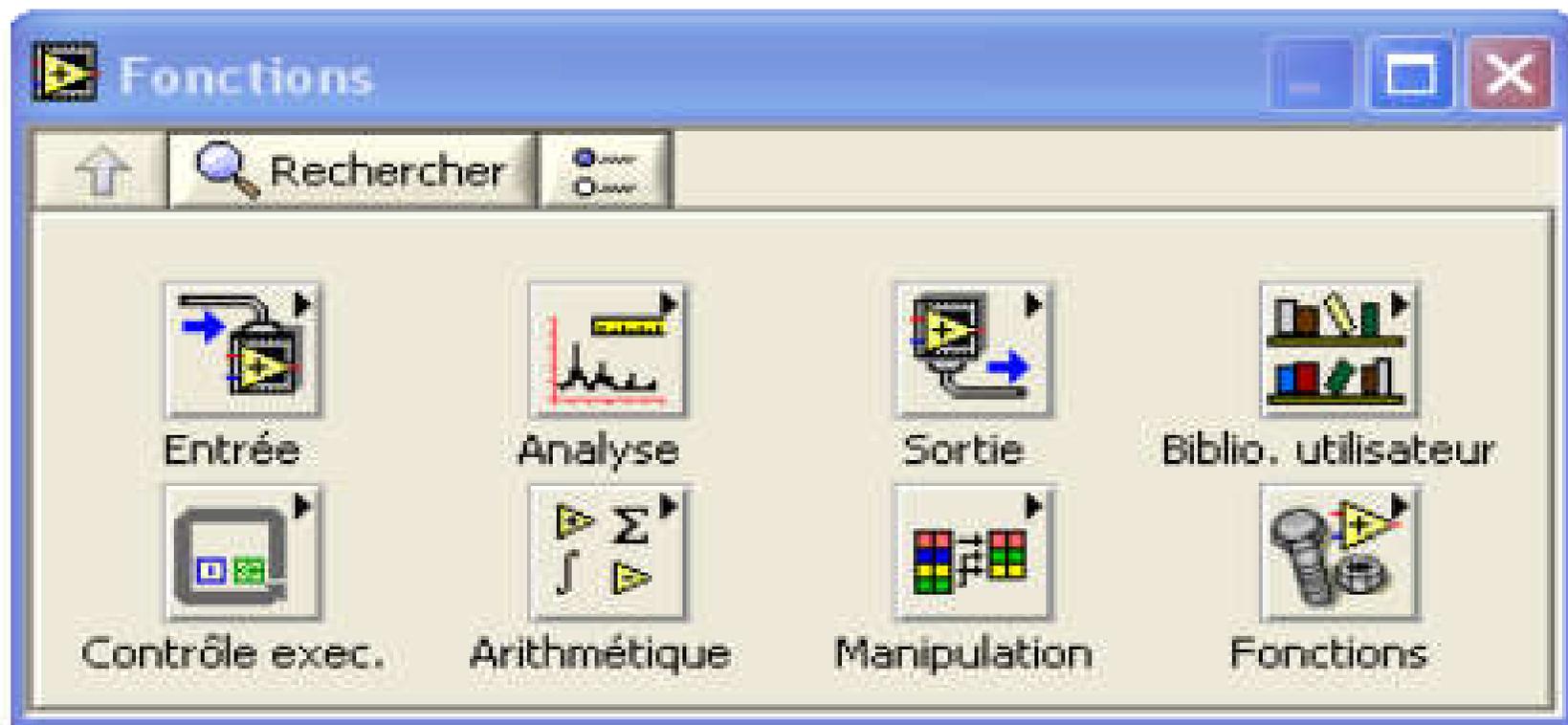
Permet de garder la palette visible sur l'écran





Palettes de fonctions.

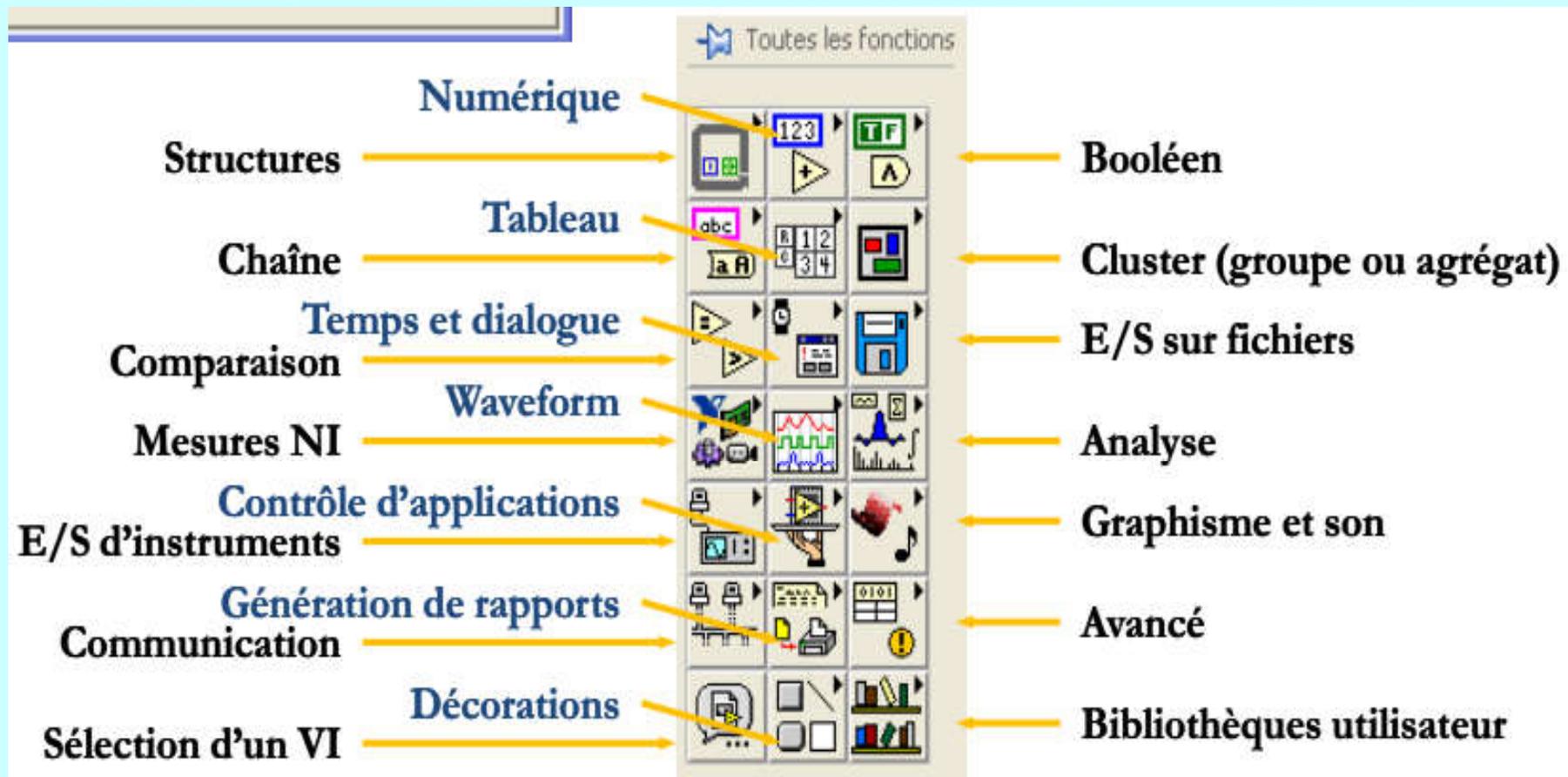
Palettes de fonctions (disponible à partir de la fenêtre diagramme par un clic droit avec la souris ou dans la barre des menus : “Fenêtre”).





Environnement LabVIEW

Dans cette palette nous trouverons tous les éléments nécessaires à la création du code graphique dans la fenêtre diagramme (fonctions de base, VI Express,...).





Palette d'outils

Utilisée pour agir sur les objets de la face avant et du diagramme (disponible dans la barre des menus : “Fenêtre”).



Outil sélection automatique
(actif lorsque la LED verte est allumée)



Outil d'action sur la face avant



Outil déplacement et taille



Outil texte



Outil connexion par fils



Outil raccourci menu





Palette d'outils

Utilisée pour agir sur les objets de la face avant et du diagramme (disponible dans la barre des menus : “Fenêtre”).



Outil déplacement



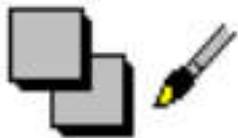
Outil d'arrêt



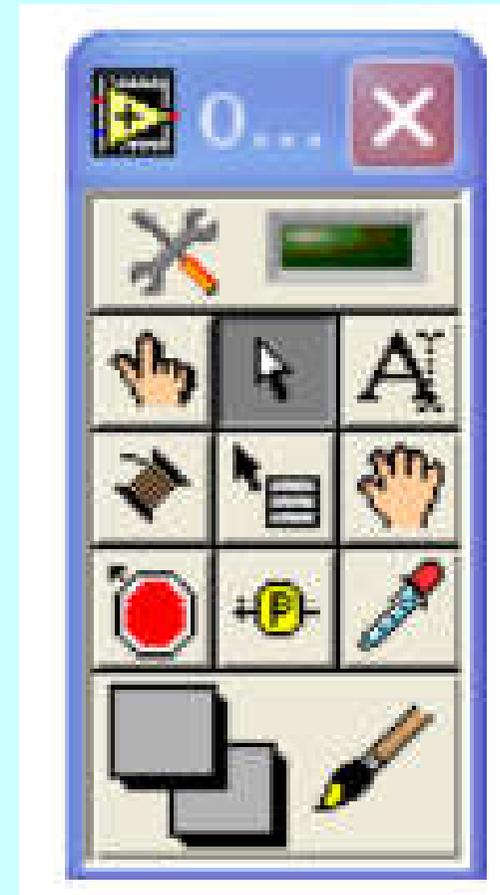
Outil sonde



Outil copie couleur



Outil coloriage





11. Barre d'outils de la face-avant (1)

Aide contextuelle



Bouton **Exécution du programme (VI)**



Bouton **Exécution continue**



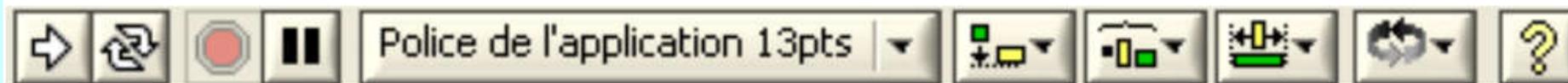
Bouton d'**Arrêt d'exécution**



Bouton **Pause/Reprendre**



11. Barre d'outils de la face-avant (2)



Police de l'application 13pts

Configuration du format du texte
(taille, style, couleur,...)



Aligner les objets



Égalisation de l'espacement entre
les objets



Plan de l'objet (premier ou arrière
plan,...)



Redimensionner les objets de la face
avant



12. Barre d'outils du diagramme.



Animer l'exécution : pour animer le diagramme et voir les données évoluer en fonction de l'exécution du code.



Exécuter de façon détailler : exécute le programme de la façon la plus détaillée possible action par action.



Exécuter sans détailler : exécute le programme nœud par nœud sans rentrer dans le détail de leur exécution interne.



Terminer l'exécution : poursuivre l'exécution du programme jusqu'à son terme.



Palette d'outils





Palette d'outils

