

Initiation au logiciel LabVIEW

Cours nr 1



Architecture des Systèmes d'Information

- | Présentation générale de LabVIEW
- | Programmation graphique en langage « G »
- | Fonctionnement de LabVIEW
- | Édition des instruments virtuels (VIs)
 - *Palettes « Tools », « Controls » et « Functions »*
- | Programmation du diagramme
 - *Structures, Boîtes de calcul et Variables*

Cours nr 2

- Conception hiérarchique : VI et sous-VI
- Compilation et exécution des VIs
- Fonctions de mise au point
- Interfaçage avec du code provenant d'autres langages
- Autres fonctionnalités de LabVIEW

Présentation du logiciel LabVIEW



Architecture des Systèmes d'Information

- Fondé sur un langage de programmation graphique : le « G », pour créer un programme sous forme de diagramme
- Destiné au développement d'applications d'acquisition, d'analyse et de présentation de données
- Couplé à des cartes d'entrées/sorties, permet de gérer des flux d'informations numériques ou analogiques et de créer ou de simuler des appareils de laboratoire

Présentation du logiciel LabVIEW



Architecture des Systèmes d'Information

ACQUISITION	ANALYSE	PRESENTATION
<i>Contrôle d'instruments</i> <ul style="list-style-type: none"> - GPIB IEEE 488 - RS-232 - VXI 	<i>Traitement numérique</i> <ul style="list-style-type: none"> - Génération de signaux - Filtrage, fenêtrage - Analyse fréquentielle 	<i>Affichage des données</i> <ul style="list-style-type: none"> - Interfaces interactives - Graphiques, courbes?
<i>Acquisition de données</i> <ul style="list-style-type: none"> - E/S Analogiques - E/S Numériques 	<i>Traitement statistique</i> <ul style="list-style-type: none"> - Régression, lissage - Moyenne, écart type? 	<i>Stockage des données</i> <ul style="list-style-type: none"> - Archivage - Impression

Présentation du logiciel LabVIEW



Architecture des Systèmes d'Information

- Outil d'aide à la conception et à la réalisation des applications
 - Objets proches graphiquement des objets réels (*e.g. voyants, curseurs, interrupteurs, boutons, graphes*)
 - Outils familiers et simples d'utilisation pour la programmation (*e.g. structures de programme, fonctions arithmétiques, fonctions logiques, comparateurs*)
 - Fonctions plus complexes (*e.g. transformée de Fourier rapide, filtres numériques*)

Présentation du logiciel LabVIEW



Architecture des Systèmes d'Information

- Pour créer des programmes compilés
- Pour créer des exécutable autonomes
 - *qui offrent des vitesses d'exécution nécessaires pour des solutions d'acquisition de données, de test et de mesure personnalisées*

Programmation graphique en langage « G »

- Langage propre à LabVIEW
- Partie intégrante de BridgeVIEW *qui est l'environnement de développement d'applications de National Instruments*
- Accompagné de bibliothèques des fonctions et des outils conventionnels de développement
- REMARQUE : Un langage textuel suit des règles d'exécution séquentielle déterminée par la position des instructions dans le fichier source, tandis qu'un programme graphique s'exécute selon le principe de flux de donnée₈

Fonctionnement de LabVIEW



- Une application développée sous LabVIEW, est appelée *instrument virtuel* (**Virtual Instrument : VI**). Elle se compose :
 - d'une *face-avant* : interface utilisateur de la VI, *permettant de réceptionner les données acquises et d'afficher celles fournies en sortie par le programme*
 - d'un *diagramme* : code source de la VI, *permettant d'effectuer des traitements sur les entrées/sorties créées dans la face-avant*

Édition des instruments virtuels (VIs)

Architecture des Systèmes d'Information

- Choisir des objets dans les palettes : « **Tools** », « **Controls** » et « **Functions** »
- Placer les objets choisis sur la face-avant ou dans le diagramme
- Déplacer ou modifier les objets placés à l'aide des outils ou des menus

Palette « Tools »

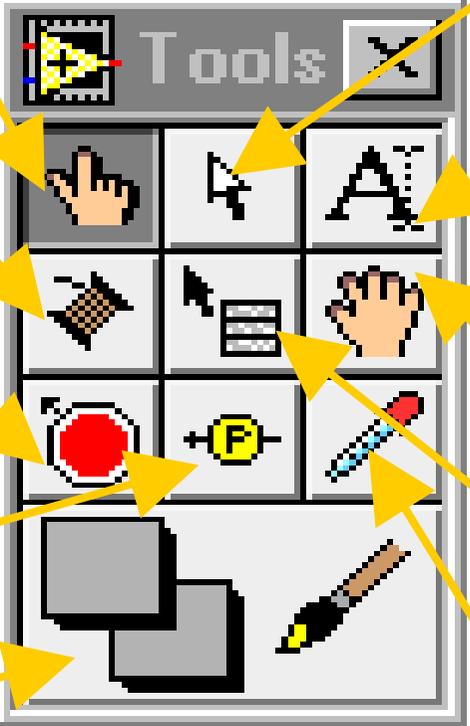
Positionner des éléments sur la face-avant et le diagramme

Connecter des éléments dans le diagramme

Placer des points d'arrêt dans les VIs

Créer des sondes sur les fils

Définir les couleurs de la face-avant et du diagramme



Sélectionner un élément pour le déplacer, le copier...

Modifier le texte et créer des commentaires

Déplacer le contenu de la fenêtre

Afficher le menu local d'un objet

Copier les couleurs pour les coller à l'aide du Pinceau

Palette « Controls »

Saisie/affichage des quantités numériques

Affichage/sélection d'une liste d'options

Saisie/affichage du chemin des fichiers

Personnaliser la face-avant



Saisie/affichage des valeurs booléennes

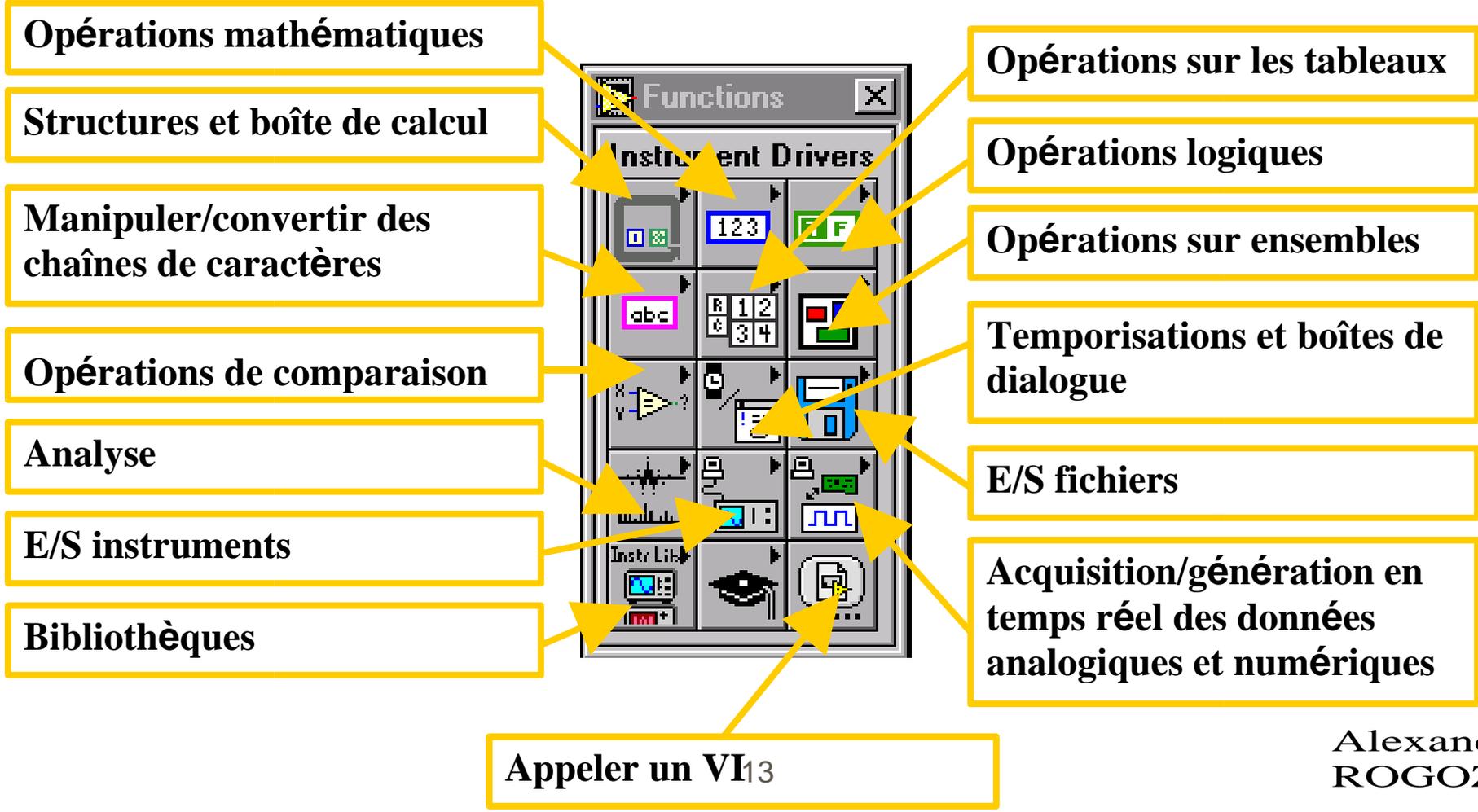
Saisie/affichage des chaînes de caractères

Tracé 2D des données numériques

Matrices et ensembles des données

Sélectionner un 'control' défini par l'utilisateur

Palette « Functions »



Alexandrina
ROGOZAN

Programmation du diagramme



Architecture des Systèmes d'Information

- **Terminaux** = ports, par lesquels les données passent entre le diagramme et la face-avant, ainsi qu'entre les nœuds et le diagramme.
 - *Remarque* : Pour afficher les terminaux d'une fonction ou d'un VI, ouvrez un menu local sur l'icône et sélectionnez **visualiser” Terminaux**.

- **Nœuds** = éléments d'exécution de programme, analogues aux instructions, opérateurs, fonctions et sous-programmes des langages de programmation conventionnels

- **Fils de liaison** = chemins de données entre les terminaux d'entrée et de sortie.

EX 1



Architecture des Systèmes d'Information

- Construire un VI « Thermomètre.vi » simulant l'acquisition de mesure de température
 - Imaginez que vous disposez d'un capteur qui convertit la température en tension.
 - Utilisez un VI existant **Demo Voltage Read** pour mesurer cette tension, puis multipliez le résultat par un nombre N pour convertir la tension en température exprimée en degrés (Fahrenheit).
 - Affichez le résultat de la mesure sur la face-avant au moyen d'un indicateur thermomètre₁₅

Alexandrina
ROGOZAN

VI et sous-VI



Architecture des Systèmes d'Information

- Un VI peut être utilisé comme sous-VI dans le diagramme d'un VI de niveau supérieur.
- Il n'existe aucune limite au nombre de VIs pouvant être utilisés dans un programme écrit en « G ».
- Un sous-VI peut également être appelé au sein d'un autre sous-VI.

Création de sous-VI à partir de VI



Architecture des Systèmes d'Information

- Pour appeler un VI à partir du diagramme d'un autre VI, il faut d'abord créer son icône et son connecteur.
 - L'*icône* d'un VI en est sa représentation graphique.
 - Le *connecteur* d'un VI affecte les commandes et les indicateurs aux terminaux d'entrée et de sortie.

EX 2

- Transformer le VI créé précédemment en sous-VI
 - Appelez l'Éditeur d'icônes en ouvrant un menu local sur le cadre « icône » de la face-avant
 - Créez une icône pour le sous-VI
 - Créez le connecteur et affectez celui-ci au thermomètre

- Visualiser sous forme graphique les liens de dépendance du sous-VI créé

Compilation et exécution des VIs



Architecture des Systèmes d'Information

I Mode d'exécution continu du VI

- I Il faut activez la face-avant en cliquant n'importe où sur celle-ci. Lancez le VI en cliquant sur le bouton « Exécution » de la barre d'outils de la face-avant.
 - *Remarque : on se doit de relancer le VI à chaque fois*
- I Si l'on souhaite une exécution permanente, on doit cliquer sur le bouton « Exécution permanente ».
 - *Remarque : il suffit de cliquer une 2ème fois sur le bouton « Exécution permanente » pour le désactiver. Le VI termine alors l'exécution et se ferme.*

Compilation et exécution des VIs



Architecture des Systèmes d'Information

I Mode d'exécution pas à pas du VI

- I Pour exécuter sans détailler une boucle ou un sous-VI, il faut cliquer sur le bouton « Exécution semi-détaillée ».
- I Pour exécuter de façon détaillée une boucle ou un sous-VI cliquez sur le bouton « Exécution détaillée ».
 - *Remarque : Pour sortir d'une boucle ou d'un sous-VI, il faut cliquer sur le bouton « Sortie ».*

Compilation et exécution des VIs



I Mode d'exécution pas à pas du VI

I Définir le niveau d'exécution d'un VI

- *Demander à ce que l'exécution s'interrompe à un instant donné en cliquant sur le bouton « Sortie » tout en maintenant le bouton de la souris enfoncé*

I Suivre le flux des données dans le diagramme

- *Animer le diagramme pendant l'exécution d'un VI en cliquant sur le bouton « Ampoule »*

Fonctions de mise au point



Architecture des Systèmes d'Information

- Poser un point d'arrêt sur une structure ou un fil
 - Choisir l'outil « Point d'arrêt » de la palette « Tools », placer le curseur « Point d'arrêt » sur l'élément et cliquer dessus. Un cadre rouge encercle alors le nœud de l'élément visé.
 - Cliquer sur le curseur « Point d'arrêt » de l'élément pour supprimer le point d'arrêt.

- Visualiser les données dans les fils de connexion
 - Choisir l'outil « Sonde » dans la palette « Tools », placer une sonde sur un fil. Une fenêtre libellée « Probe 1 » apparaît à la fois dans la face-avant et dans le diagramme.
 - Pendant l'exécution du VI en mode Pas à Pas, la fenêtre de la sonde affiche la valeur des données lorsqu'elles passent par le fil sélectionné.

Ex 3

- Exécuter en mode continu le VI « Thermomètre.vi »
- Positionner l'Outil « Sonde » sur un fil du VI de manière à visualiser la valeur des données qui y circulent
- Examiner le flux des données en utilisant le Mode « Animation »

Structures



Architecture des Systèmes d'Information

- Séquence (*Sequence Structure*)
- Alternative (*Case Structure*)
- Répétition (*While Loop* et *For Loop*)

Ex 4

- Utiliser une boucle « While » et un graphe déroulant pour acquérir et présenter les mesures acquises à l'aide du sous-VI « thermomètre.vi ».

- Modifier le VI créé pour que l'acquisition se fasse à des intervalles de temps réguliers.
 - On en règle la durée au moyen d'un bouton rotatif qui doit donc contrôler le temps de cycle de la boucle « While »

Ex 5



Architecture des Systèmes d'Information

- Créer un VI qui mesure une température toutes les 0,25 secondes pendant 10 secondes.
- En cours d'acquisition, le VI visualise les mesures en temps réel sur un graphe déroulant.
- Lorsque le processus d'acquisition est terminé, le VI trace un graphe mettant en surbrillance les températures moyenne, maximale et minimale.

Création d'une boîte de calcul

- Placer la boîte de calcul sur le diagramme en la sélectionnant dans **Fonctions**»**Structures**.

- Entrer la ou les formules dans la boîte en utilisant l'outil « Texte ».
 - *Remarques* : Chaque déclaration de formule doit se terminer par un point virgule.
Les variables dans les formules font la différence entre majuscules et minuscules.

- Créer des terminaux d'entrée/sortie de la boîte de calcul en ouvrant le menu local sur la bordure du nœud et choisissez « Ajouter une entrée/Ajouter une sortie ».

VARIABLES LOCALES ET GLOBALES



Architecture des Systèmes d'Information

- Une variable globale est un objet intégré au G.
 - *Lorsque on crée une variable globale, un type spécial de VI est automatiquement créé.*
 - *On ajoute à ce VI des commandes de face-avant qui définissent les types de données qu'il contient.*

- Une variable locale permet d'écrire/lire une commande ou un indicateur de la face-avant d'un VI.
 - *On peut utiliser une commande de face-avant comme une entrée/sortie grâce à une référence de variable locale.*

**Alexandrina
ROGOZAN**

Ex 6



Architecture des Systèmes d'Information

- Remplacer la Fonction « Multiplication » utilisée dans le VI « thermomètre.vi » par une boîte de calcul effectuant l'opération.

Ex 7



Architecture des Systèmes d'Information

- Créer un VI pour ajouter des données de température à un fichier ASCII.
- Ce VI utilise une boucle « For » pour générer les valeurs de température et les stocker dans un fichier.
- Au cours de chaque itération, convertir les données en chaîne de caractères, ajouter une virgule comme séparateur, puis ajouter la chaîne de caractères dans un fichier.

Interfaçage avec du code provenant d'autres langages

- Transmettre des structures de données complexes en code C au travers un CIN (Code Interface Node).
 - *Obtention de meilleures performances, les structures de données étant transmises au CIN au même format que leur stockage dans le « G »*

LabVIEW offre aussi la possibilité :



- d'utiliser des contrôles ActiveX et de faire appel à des fonctions DLL,
- de faire de l'acquisition/restitution à l'aide de cartes propriétaires,
- d'exploiter des outils de contrôle du protocole GPIB,
- de communiquer avec les services réseaux proposés (UDP, TCP, ...),
- d'utiliser un outil de visualisation des performances des VIs, afin d'optimiser les temps d'exécution,
- d'utiliser des outils d'analyse en traitement du signal (FFT, filtrages, convolutions,...).