



LabWindows est un environnement de développement en langage C pour la réalisation de systèmes de test, mesure et d'instrumentation. Ainsi, il est possible de profiter à la fois de la puissance et la précision du langage C mais aussi des bibliothèques de fonctions dédiées à l'instrumentation et au traitement du signal de National Instruments.

Avec [LabWindows/CVI](#), il est possible de développer :

- des logiciels pour Windows (exe) ou UNIX, Linux,
- des bibliothèques (DLL, Active X, .NET),
- des drivers d'instruments,
- sur cibles embarquées,
- sur cible temps réel.

Exemples de réalisations

Pour voir une idée des utilisations possibles de [LabWindows/CVI](#) pour la réalisation et le développement de projets, cliquez sur les liens suivants :

- [Système de calibrage de sondes de puits de pétrole](#) (description détaillée),
- [Validation d'autoradio](#) (description détaillée),
- [Liste des projets développés sous LabWindows/CVI](#) auxquels j'ai participé (descriptions rapides).

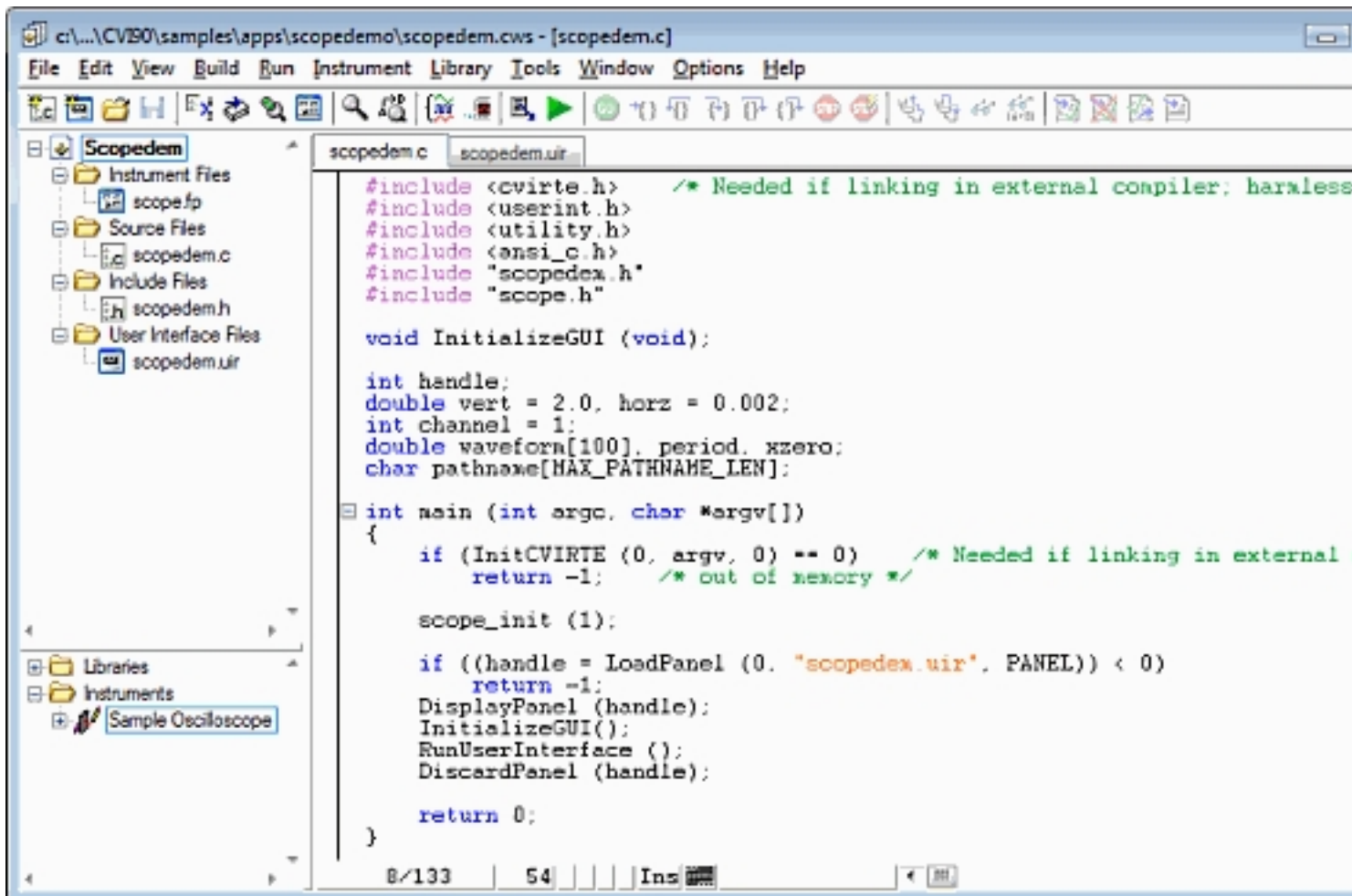
Les avantages de [LabWindows/CVI](#)

Utiliser [LabWindows/CVI](#) c'est la garantie d'un développement solide et d'une solution pérenne. Lisez plutôt les avantages suivants :

- contrôles graphiques spécialisés pour l'industrie et les mesures scientifiques,
- développement simplifié d'IHM,
- fonctions intégrées d'analyse et traitement,
- accès facilité aux instruments,
- programmation en C, langage puissance, précis et connu par tous (toutes les écoles d'ingénieurs et de techniciens l'enseignent),
- accès au SDK de Windows pour les développements très spécifiques.

Environnement de développement intégré (IDE)

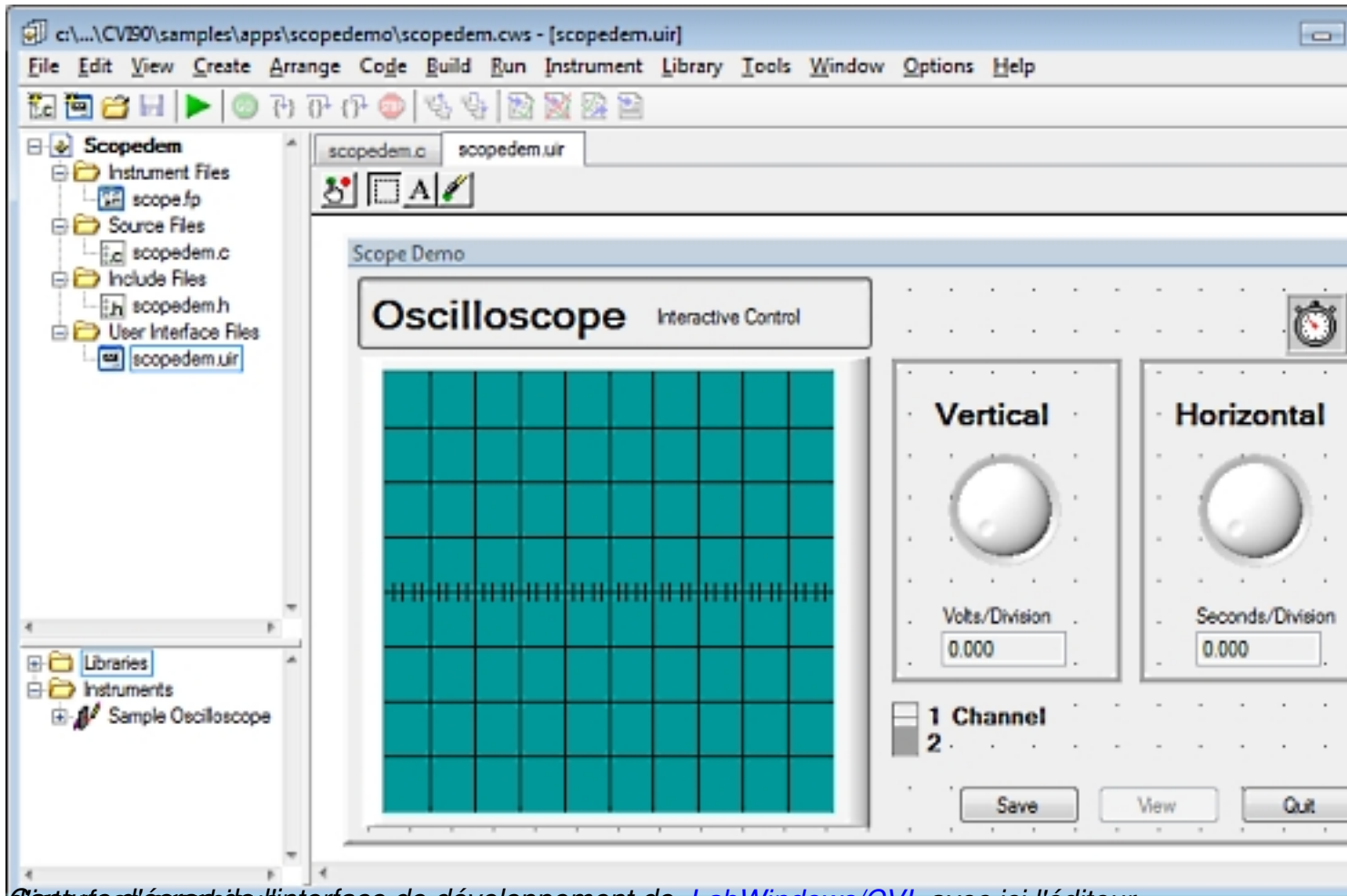
L'interface de développement de [LabWindows/CVI](#) est assez similaire aux autres IDE, comme Visual Studio par exemple.



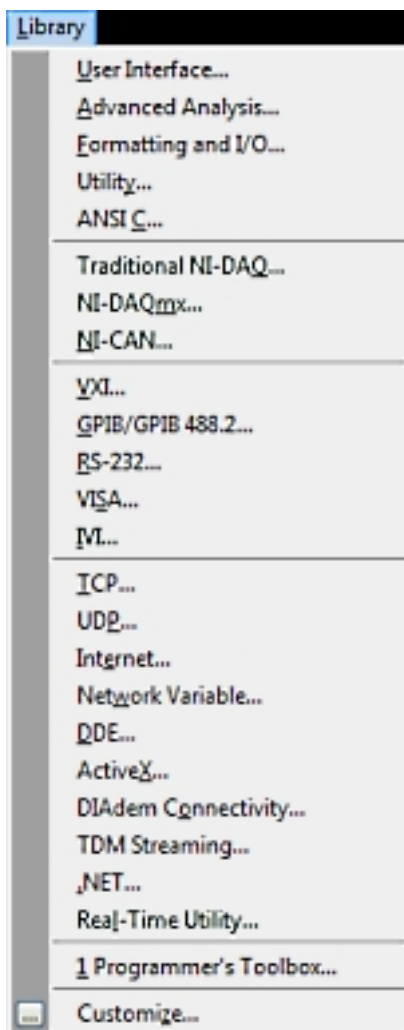
Capture d'écran de l'interface de développement de [LabWindows/CVI](#), ici avec l'éditeur de code C

Il comporte les fonctions qu'on aime à retrouver :

- un panneau du workspace (en haut à gauche) contenant une arborescence du ou des projets concernés par le développement et leurs fichiers sources,
- un panneau des librairies et drivers d'instruments (en bas à gauche),
- un panneau principal multi-usage :
 - éditeur des fichiers source en C (comme ci-dessus) avec coloration syntaxique, complétion des instructions...,
 - éditeur d'interface graphique (comme ci-dessous),
 - éditeur de "function panel" (aide à la programmation et documentation pour l'utilisation de drivers et autres composants logiciels),
 - navigateur de code source,
 - watch et view sur les variables avec visualisation avancée : on peut par exemple afficher les tableaux numérique sous forme de graph,
 - liste des modules chargés (DLL pour Windows) lors de l'exécution du code,
 - liste des tâches,
 - liste de la pile de fonctions,
 - éditeur de la mémoire.



Capture d'écran de l'interface de développement de [LabWindows/CVI](#) avec ici l'éditeur

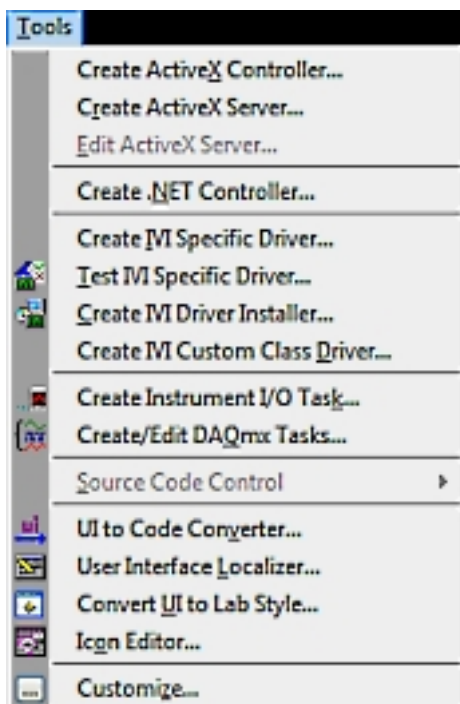


Bibliothèques de fonctionnalités

En parcourant le menu des librairies (voir ci-contre), on remarque la présence de ce qu'il est nécessaires au développement d'applications modernes mais surtout de nombreuses fonctions pour :

- la communication avec les cartes d'acquisitions (traditional NI- [DAQ](#) et NI-DAQmx),
- le support de protocoles et normes les plus utilisés dans l'instrumentation ([VXI](#) , [GPIB](#) et 488.2, [RS-232](#) , NI- [CAN](#) , VISA, IVI),
- l'échange de données sur le réseau (TCP, UDP, Internet, Network Variable),
- la communication avec d'autres composants logiciels (DDE, ActiveX, DIAdem Connectivity, .NET),

- la sauvegarde de données massives (TDM Streaming),
- la programmation temps réel (Real-Time Utility),
- le traitement des données de mesure (Advanced Analysis),
- les aides et assistants pour les fonctions du langage C (ANSI C),
- le traitement simplifié des chaînes de caractères et des fichiers (Formatting and I/O),
- la gestion des interfaces graphiques (User Interface)



Assistants

[LabWindows/CVI](#) comporte également plusieurs assistants (voir ci-contre) qui facilitent grandement le développement d'applications industrielles et scientifiques :

- création de tâches de communication avec un instrument,
- création de tâches d'acquisition de mesure,
- création de drivers IVI.

mais aussi des assistants pour la communication avec d'autres composants logiciels :

- création de contrôleur ActiveX,
- création de serveur ActiveX,
- création de contrôleur .NET.

Alternatives à [LabWindows/CVI](#)

Pour bénéficier des mêmes fonctionnalités que [LabWindows/CVI](#) pour d'autres langages, référez vous aux articles suivants :

- langage C++, C# ou Visual Basic sous Visual Studio, lisez l'article "[Measurement Studio](#)" ,
- langage Graphique, lisez l'article "[LabVIEW](#)" ,
- pour les bancs de test, on utilise souvent [LabWindows/CVI](#) avec un séquenceur de test comme [TestStand](#) .

[Système de calibrage de sondes de puits de pétrole](#)