



Formation Threads et Muti Tache

L'Institut Pascal - formation@formation-delphi.com - Tél 01.42.83.69.36

Objectifs	comprendre les concepts multi-tâche liés aux <i>Threads</i> Windows, et construire des applications utilisant des threads communiquant entre eux, et partageant des données de façon synchronisée entre eux et avec la Forme
Durée	1 jours
Versions Delphi	Delphi 6 à Delphi 2010
Pré Requis	expérience du développement Delphi d' au moins trois mois
Public	développeur et analyste, dans le cadre du plan de formation d'entreprise , ou d'un DIF

Programme Détaillé - Formation Threads et Muti Tache

1 Principe des Threads

- les processus Windows
- multi-tâche Windows coopératif et préemptif
- définition et propriétés des threads
- un premier exemple
- les primitives des threads Windows (API)

2 Lancement et Arrêt

- la classe *tThread*
- création en mode immédiat ou suspendu
- Suspend* et *Resume*
- arrêt du thread
- test et gestion de la fin du *tThread*

3 Communication entre les Threads

- communication par *SendMessage* et *PostMessage*
- PostThreadMessage*
- les API *WaitForSingleObject*
- accès à la *tForm Delphi* et *Synchronize*
- affichage d'un journal dans un *tMemo*

4 Accès aux données

- données TLS, **ThreadVar** et attributs du *tThread*
- tThreadLocalCounter*
- propriété *Lock* de la VCL
- tThreadList*, *LockList* et *UnlockList*
- accès aux globales

5 mécanismes de synchronisation

- section critique
- mutex
- sémaphore
- événement *Windows*
- comparaison entre les primitives de synchronisation
- consommateur / producteur
- les bases de données et les threads
- exemple de journal disque

6 Les primitives avancées

- primitive Windows *InterlockedIncrement*
- iReadWriteSync*
- tMultiReadExclusiveWriteSynchronizer*
- construction de primitive spécialisée

7 Utilisation des Threads

- les sockets Tcp/Ip Delphi