embarcadero

Embarcadero Conference

Um único esforço, uma única base de código, múltiplas plataformas, múltiplos dispositivos

Kelver Merlotti

TParallel - Framework Nativo para Computação Paralela

Kelver Merlotti

- Gerente de Treinamentos e Serviços da E do Brasil
- Coordenador Editorial da Active Delphi
- +14 anos de diversão com Delphi! :)

embarcadero

O que tínhamos?

TThread

O que tínhamos...

```
type
  TMyThread = class(TThread)
  private
    { Private declarations }
  protected
    procedure Execute; override;
  end;

implementation

{ TMyThread }

procedure TMyThread.Execute;
begin
   ...
   Synchronize(UpdateCaption);
end;

procedure TMyThread.UpdateCaption;
begin
   Form1.Caption := 'Updated in a thread';
end;
```

embarcadero

E depois?

TThread.CreateAnonymousThread

Um pouco mais fácil...

```
...
var
  LocalVar: TSomeType;
begin
  TThread.CreateAnonymousThread(
    procedure
    begin
      // ... some code which uses LocalVar

      TThread.Synchronize(
        TThread.CurrentThread,
        procedure
        begin
          Form1.Caption := 'Updated in a thread ' + LocalVar.ToString;
        end;
      )
    end).Start;
end;
```

embarcadero

E agora...

TParallel ...threads made easy! :)

TParallel

- Novo framework para computação paralela
- Tira proveito de CPU's muti-core
- Multi-plataforma e multi-device
- Escalonável e auto dimensionável
- Pool de threads customizável, se necessário
- Tudo em uma única unit: **System.Threading**

TParallel.For

- Executa uma estrutura de repetição com quantidade predefinida, dividindo as iterações entre processos paralelos (threads)

embarcadero

TParallel.For na prática!

L SEP

Tasks

- **TTask** é uma classe que permite criar e gerenciar instâncias de **ITask**, que podem ser compreendidas como tarefas.

- Antes de seguir adiante com o código, pode-se optar por esperar todas as tarefas em execução terminarem, usando o método **WaitForAll**, ou esperar por qualquer uma delas através do método **WaitForAny**)

- Também pode ser usado para um processamento simples necessariamente em background


embarcadero
TTask
SEP
na prática!

TTask.IFuture

- Implementa um ITask para criar uma função que retorna um tipo específico, usando Generics

- Permite que o código seja executado sem interrupção, até que o valor de IFuture seja requisitado


embarcadero
TTask.IFuture
na prática!
L
SEP

Thread Pool

- Automaticamente gerenciado, aumenta ou diminui seu tamanho baseado na demanda da CPU durante a execução da aplicação
- Apesar de desnecessário, pode ser customizado
- Por padrão, considera a criação de 25 threads por núcleo de processamento (MaxThreadsPerCPU), sendo o máximo de threads definido pela expressão:
  - `TThread.ProcessorCount * MaxThreadsPerCPU`

Customizando o Thread Pool

- Variáveis globais para armazenar um TThreadPool tem mais performance do que locais
- Exemplo de uso:

```
var
  CustomPool: TThreadPool;
...

Procedure ButtonXClick(Sender: Tobject);
begin
  if CustomPool = nil then
  begin
    CustomPool := TThreadPool.Create;
    CustomPool.SetMaxWorkerThreads(10);
  end;

  ...
  TParallel.For(2, 1, Max, procedure (I: Integer)
    begin
      if IsPrime (I) then
        TInterlocked.Increment (Tot);
    end, CustomPool);
  ...
end;
```

Resumo

- **TParallel** é um conjunto de classes que facilitam a programação paralela, com aproveitamento e controle automático do uso de CPUs, podendo ainda customizar este **Pool de Threads**;
- TParallel.**For** executa um "for" quebrando-o em Threads;
- **Tasks** permitem aguardar um conjunto de processamentos ser executado, ou ainda executar um ou mais processos em segundo plano;
- **Future** permitem executar um código em paralelo, interrompendo o fluxo principal caso a thread não tenha sido finalizada ainda.

Dúvidas?

Links úteis:

http://community.embarcadero.com/index.php/blogs/entry/parallel-programming-using-the-new-rad-studio-xe7-runtime-library

docwiki.embarcadero.com/Libraries/XE7/en/System.Threading

www.embarcaderobr.com.br/servicos

www.embarcaderobr.com.br/treinamentos

https://www.facebook.com/EmbarcaderoBR

https://www.youtube.com/user/EmbarcaderoTechNet

https://www.youtube.com/user/embarcaderodobrasil

www.activedelphi.com.br

Obrigado!

kmerlotti@embarcadero.com.br
kelver@activedelphi.com.br
fb.com/kmerlotti
@kmerlotti