

Embarcadero Conference 2024

Inovação faz parte do nosso DNA!

Evolução do código: **Adaptando sistemas legados** **para execução paralela**

{Moisés Baddini **and** Rafael Serafim



Zépa

Alterdata Software

Techlead com mais de 18 anos de experiência

Pós-graduado Gestão de projetos

Monitor projetos Serratec

Conteudista Escola Tech4me

if café and paçoca then code();



Rafael Serafim

Alterdata Software

Supervisor técnico de programação

Professor universitário na UNIFAA

MBA Gestão de projetos, pós em ciência de dados e arquitetura de software

Certified Delphi Developer





Alterdata
software

LEGADO ou **OBSOLETEO**



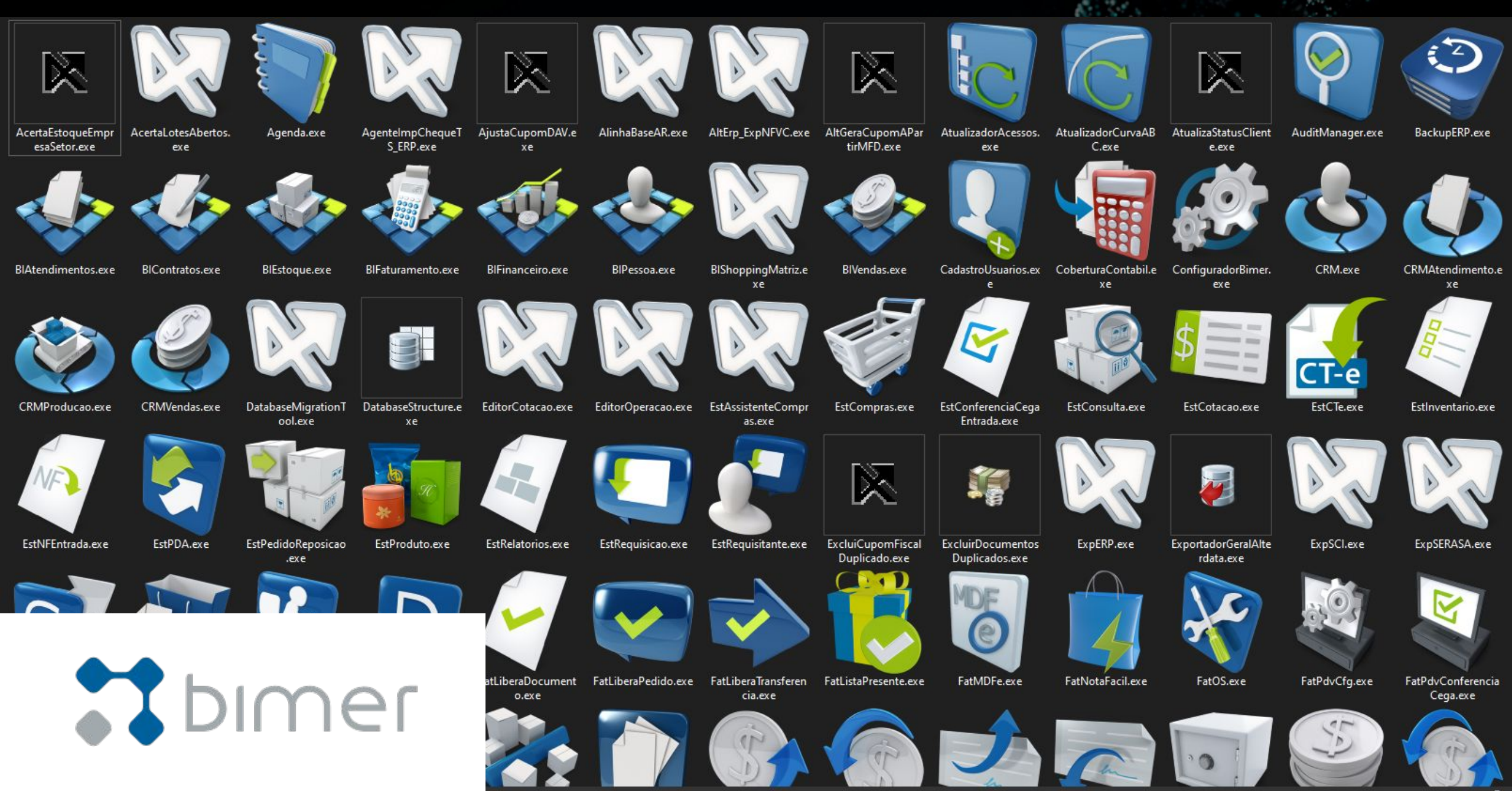


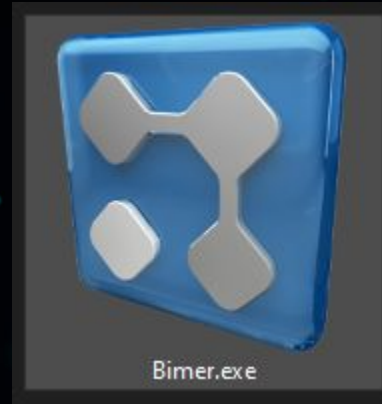
**AME SEU
LEGADO**



Cases e updates







Live desktop CRM ✕

Código

Nome

Nome curto

Dt. cadastro



E-mail



Pareceres pendentes



Atendimento



Seleção de pessoas



Enviar opinião



Ajuda



Live desktop Faturamento BI Atendimento Produto

Principal Relatórios Ferramentas Diversos

Novo Abrir Visualizar Excluir Localizar Imprimir

Cadastro

Cancelamento Excluir nota fiscal Devolver nota fiscal Manifesto de entrega Gerar notas a partir de pedido
Nota fiscalLinha do tempo Recarregar opções
Imprimir duplicatas Enviar opinião
Cancelar impressão de duplicata Ajuda

Outros

Empresa do estoque

Dt. emissão inicial

29/08/2024

Dt. emissão final

03/09/2024

Configurações

☒ Autoajustar as colunas
☒ Atualizar valores

Atualizar filtro

Filtrar documentos

☐ Impressos ☐ Não impressos ☐ Cancelados ☐ De devolução ☐ De transferência ☐ Impressos ainda não liberados

Nota fiscal

Legenda Restaurar Seleção de campos

Arraste o título de uma coluna aqui para agrupar por essa coluna

<	Nr. NF	▲	Empresa de venda	Espéci	Série	Tipo	Pessoa	Dt. emissão	Usuário	Pedido	Diversos	Cancela	Impress	Devolvido	Cupom	VI. NF	Cód. da
---	--------	---	------------------	--------	-------	------	--------	-------------	---------	--------	----------	---------	---------	-----------	-------	--------	---------

<Não existem dados para visualizar>

0

Nota fiscal Ordem de entrega Nota fiscal de serviço

NUM

PATRICK.DSN.ERP

Alterdata trunk

update

alterdata

3 Dicas para transformar Legado em thread



1. Desacoplar



1. Desacoplar

A VCL não garante thread-safe

```
TThread.CreateAnonymousThread(  
  procedure  
  begin  
    var MeuFormLegado := TMeuFormLegado.Create(nil);  
    try  
      MeuFormLegado.BtnOK.Click();  
    finally  
      MeuFormLegado.Free;  
    end;  
  end  
) .Start;
```



1. Desacoplar

A VCL não garante thread-safe

```
TThread.CreateAnonymousThread(  
  procedure  
  begin  
    var RotinaDesacopladaDoForm := TRotinaDesacopladaDoForm.Create();  
    try  
      RotinaDesacopladaDoForm.Executar();  
    finally  
      RotinaDesacopladaDoForm.Free;  
    end;  
  end  
) .Start;
```



- 1. Desacoplar**
- 2. Sincronizar**



2. Sincronizar

Não interagir com a **tela a partir de uma thread** diretamente

```
TThread.CreateAnonymousThread(  
  procedure  
  begin  
    var RotinaDesacopladaDoForm := TRotinaDesacopladaDoForm.Create();  
    try  
      RotinaDesacopladaDoForm.Executar();  
      RotinaDesacopladaDoForm.PreencherTelaComResultado();  
    finally  
      RotinaDesacopladaDoForm.Free;  
    end;  
  end  
) .Start;
```



2. Sincronizar

Não interagir com a **tela a partir de uma thread** diretamente

```
TThread.CreateAnonymousThread(  
  procedure  
  begin  
    var RotinaDesacopladaDoForm := TRotinaDesacopladaDoForm.Create();  
    try  
      RotinaDesacopladaDoForm.Executar();  
  
      TThread.Synchronize(nil,  
        procedure  
        begin  
          RotinaDesacopladaDoForm.PreencherTelaComResultado();  
        end  
      );  
    finally  
      RotinaDesacopladaDoForm.Free;  
    end;  
  end  
) .Start;
```



- 1. Desacoplar**
- 2. Sincronizar**
- 3. Gerenciar concorrência**



3. Gerenciar concorrência

Quando 2 códigos tentam acessar o mesmo recurso ao mesmo tempo.

```
Initialization  
FCacheGlobal := TList<TObjetoQualquer>.Create();
```

```
procedure TRotinaDesacopladaDoForm.Executar;  
begin  
  var lObjeto := ObterObjetoDoCacheGlobal();  
  ExecutarRotina(lObjeto);  
end;
```



3. Gerenciar concorrência

Quando 2 códigos tentam acessar o mesmo recurso ao mesmo tempo.

```
initialization
  FCacheGlobal := TList<TObjetoQualquer>.Create();
  FLock := TCriticalSection.Create;
```

```
procedure TRotinaDesacopladaDoForm.Executar;
begin
  var lObjeto := nil;

  FLock.Acquire();
  try
    lObjeto := ObterObjetoDoCacheGlobal();
  finally
    FLock.Release();
  end;

  ExecutarRotina(lObjeto);
end;
```



3. Gerenciar concorrência

Quando 2 códigos tentam acessar o mesmo recurso ao mesmo tempo.

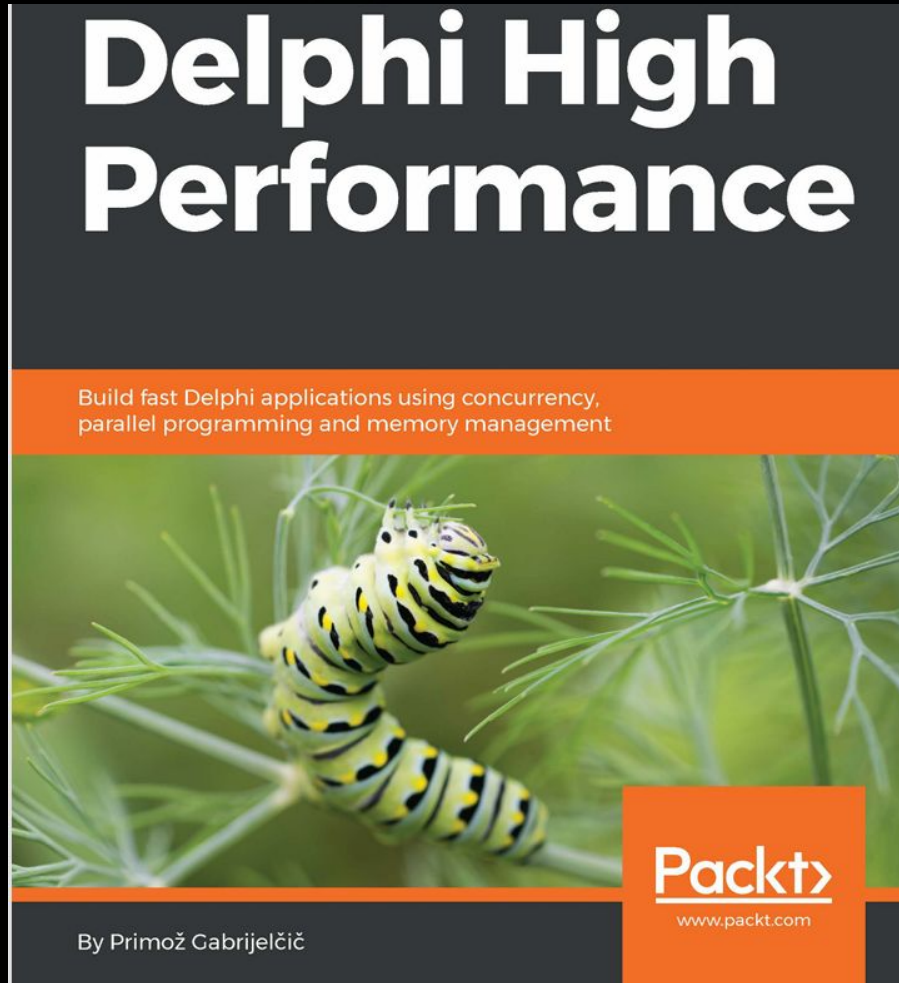
```
Initialization  
FCacheGlobal := TThreadList<TObjetoQualquer>.Create();
```



```
procedure TRotinaDesacopladaDoForm.Executar;  
begin  
  var lObjeto := nil;  
  var lThreadList := FCacheGlobal.LockList;  
  try  
    lObjeto := ObterObjetoDoCacheGlobal();  
  finally  
    FCacheGlobal.UnlockList;  
  end;  
  
  ExecutarRotina(lObjeto);  
end;
```



3. Gerenciar concorrência



Inc/Dec	
Single threaded	Single threaded: 0 [0 ms] Single threaded: 0 [0 ms] Single threaded: 0 [0 ms]
Multithreaded	Critical section: 0 [117 ms] Critical section: 0 [8 ms] Critical section: 0 [7 ms]
MT with locking	Mutex: 0 [1059 ms] Mutex: 0 [1110 ms] Mutex: 0 [1057 ms]
MT with mutex	TMonitor: 0 [17 ms] TMonitor: 0 [5 ms] TMonitor: 0 [10 ms]
MT with TMonitor	Spinlock: 0 [17 ms] Spinlock: 0 [19 ms] Spinlock: 0 [4 ms]
MT with spinlock	Interlocked: 0 [2 ms] Interlocked: 0 [2 ms] Interlocked: 0 [4 ms]
MT with interlocked	





Técnicas para paralelizar Legado



1. Escopo para threadvar

2. Function Hub



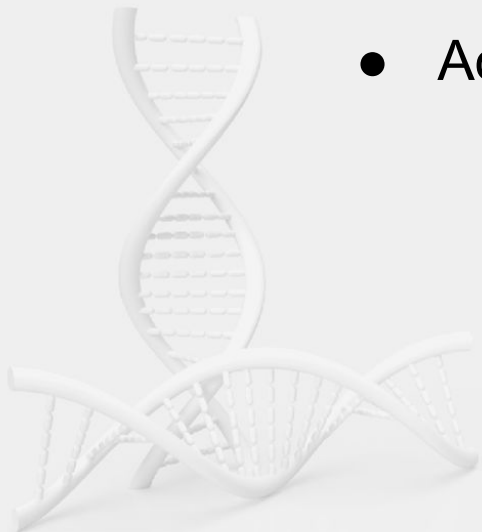
1. Escopo para threadvar



O que é threadvar

Declarar variáveis específicas para a thread, apesar de serem 'globais'

- Cada thread tem sua própria cópia da variável
- Essas variáveis não são compartilhadas
- Ao iniciar uma nova thread, a variável é nova

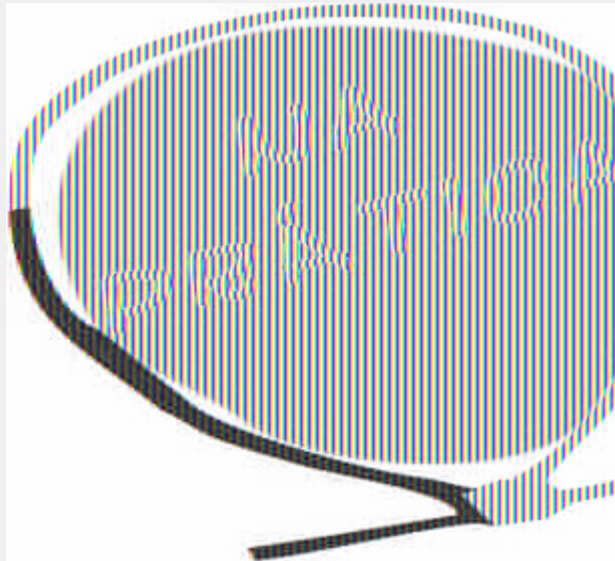


Como criar o escopo?

1. Utilizar uma rotina personalizada de iteração em threads
2. Criar uma seção de inicialização dentro de cada thread
3. Executar o processamento
4. Finalizar aglutinando os resultados de cada thread



Zepa.Threads.Iterador



2. Function Hub



Function Hub

Um intermediário que permite armazenar em lista os parâmetros necessários para a chamada de uma função.

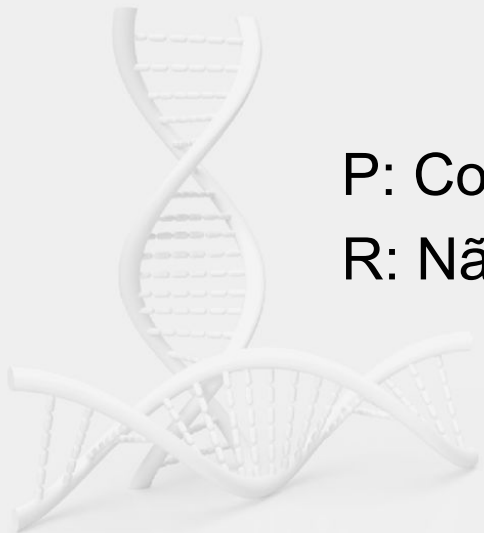
O que o hub precisa ter?

- Lista para armazenar os argumentos da função legada
- Método para adicionar os args na lista
- Chamar a execução em paralelo, que por sua vez,
- Executa a rotina legada



Como substituir?

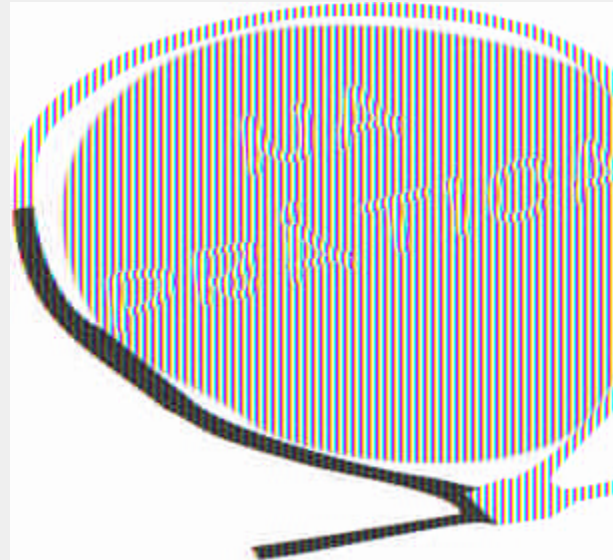
1. Retirar a unit legada e trocar pela Hub
2. Substituir as chamadas da função legada por hub.add
3. Chamar o método de execução em paralelo ao final das adições



P: Como funciona internamente o código legado que será paralelizado?

R: Não sabemos! Porém, temos sua assinatura...

ImportarTabelaGeral



Embarcadero Conference 2024

Inovação faz parte do nosso DNA!



Quer me ver na
#ECON25?
Acesse o QRCode
e avalie minha palestra!



Moisés Baddini e Rafael Serafim



[@zepasoftware](#)
[@rf.serafim](#)



[in/moisesbaddini](#)
[serafim-software](#)



<https://github.com/zepasoftware>
serafim.dsn.erp@alterdata.com.br

