SIMPLE **PARALLEL** PROGRAMMING WITH **PATTERNS** AND **OMNITHREADLIBRARY**

PRIMOŽ GABRIJELČIČ

WWW.THEDELPHIGEEK.COM, @THEDELPHIGEEK, SKYPE: GABR42

HTTP://PRIMOZ.GABRIJELCIC.ORG

Intro music: *Nothing is Permanent* by *Deathmøle, http://deathmole.bandcamp.com*

Parallel programming

*The **art** of doing multiple things at the same time*

WHY?

- CPUs are not getting any faster
- They just contain more and more parallel cores

WHEN?

- Slow background process
- Background communication
- Executing synchronous API
- Multicore data processing
- Multiple clients

THREAD VS. TASK

- *Task* is part of code that has to be executed
- *Thread* is the execution environment

- .NET – Task Parallel Library
- Delphi – Parallel Programming Library (XE7)
- Delphi – OmniThreadLibrary (2007)

PATTERN APPROACH

- Working with building blocks
- .NET – foreach, Dataflow
- C# – async
- Delphi PPL (XE7) – Future, For, Join
- Delphi OTL (2009) – Async, Future, Join, For, Pipeline …

ASYNC/AWAIT DEMO

JUST SAY NO!

Never, never, never access UI from a background thread!

NEVER!

ASYNC/AWAIT

```
Async(
    some_code    // executed in a background thread
).Await(
    other_code    // executed in a main thread
)
```

PARALLEL PATTERNS

- Async/Await
- Async
- Future
- ForEach / For
- Join
- Parallel task
- Pipeline
- Background worker
- Fork/Join
- Map

PROJECT STATUS

- http://www.omnithreadlibrary.com
- Google+ community *OmniThreadLibrary*
- StackOverflow `[omnithreadlibrary]`
- http://www.thedelphigeek.com
- http://primoz.gabrijelcic.org
- `Twitter: @thedelphigeek`
  `Skype: gabr42`