

.NET Conf

Focus on F#

July 29, 2021

focus.dotnetconf.net

Data Analytics & Machine Learning in F#

Luis Quintanilla (@ljquintanilla)



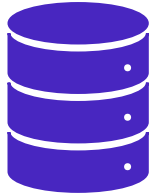
Agenda

- Data analytics & machine learning workflow
- Data analytics & machine learning landscape
- Why F# & .NET?
- Data science & machine learning with F#

Data analytics & ML workflow



Define



Get Data



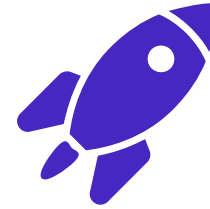
Prepare Data



Modeling



Evaluate

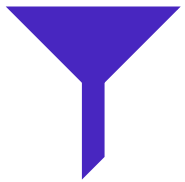


Deploy



Predict

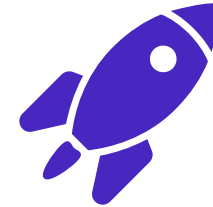
Data analytics & ML landscape



Prepare Data



Modeling



Deploy



DASK



Flask



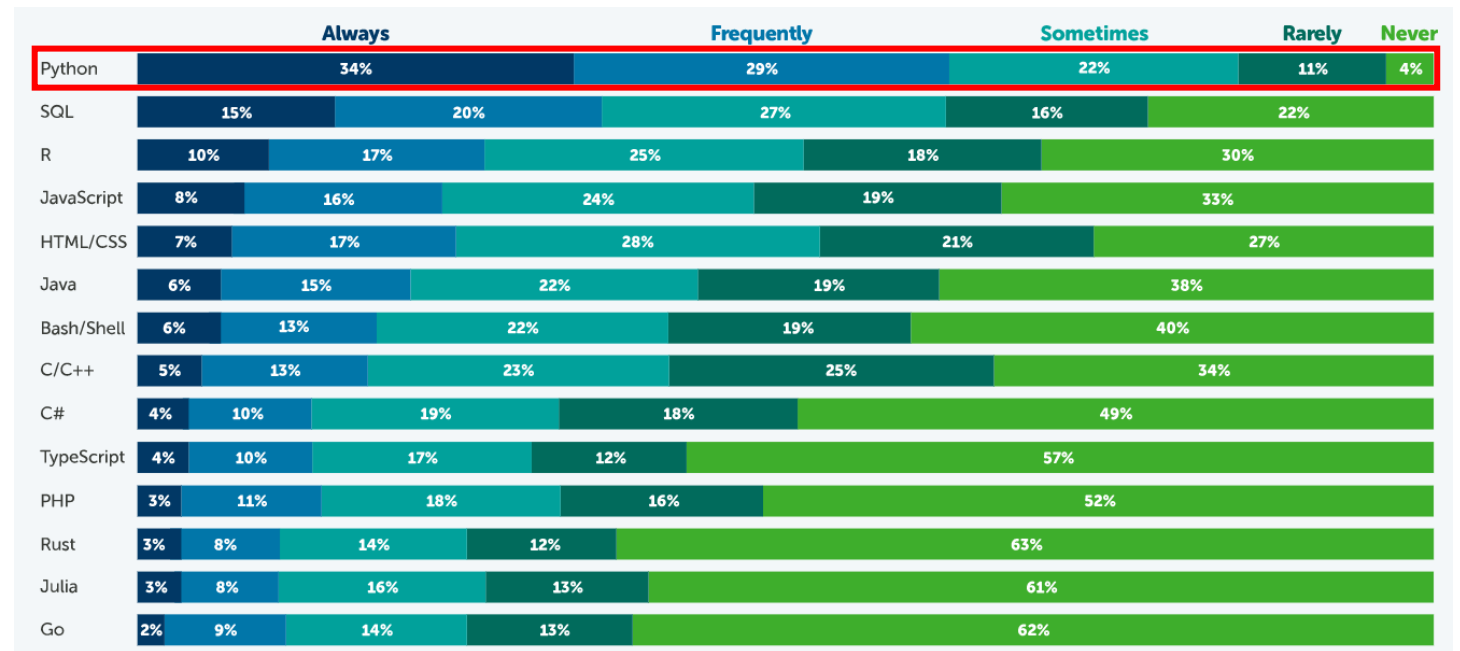
ONNX



How often do you use the following languages?

“**63%** of respondents said they always or frequently use Python”

– Anaconda State of Data Science 2021



What do you use Python for the most?

- Web development
- **Machine learning**
- **Data analysis**
- DevOps / System administration / Writing automation scripts
- Educational purposes

Favorite Python features

- Simple syntax, syntactic sugar, easy to learn
- Easy to write & read code, high-level language
- List comprehension, generator
- Versatility, libraries for any problem
- Dynamic typing, duck typing

Desired Python features

- Static typing, strict type hinting
- Performance improvements
- Better concurrency & parallelism
- Pattern matching, switch statement
- Official Python compiler, JIT compiler

```

1  #r "nuget:FSharp.Data" ← Convenient dependency management / Libraries for any problem
2
3  open FSharp.Data
4
5  let getHtml (source: string) = // string -> Async<option<HtmlDocument>>
6      async {
7          try
8              let! html = HtmlDocument.AsyncLoad(source)
9              return Some html
10         with
11         | _ ->
12             printfn $"Can't get html"
13             return None
14     }
15
16 let getLinks (document: HtmlDocument option) = // option<HtmlDocument> -> seq<HtmlNode>
17     match document with
18     | Some html -> html.Descendants [ "a" ]
19     | None -> Seq.empty
20
21 let urls = [ "invalid link"; "https://focus.dotnetconf.net/" ] // list<string>
22
23 [ for url in urls do
24     |> getHtml url ] // Async<HtmlDocument option> list
25 |> Async.Parallel // Async<HtmlDocument option []>
26 |> Async.RunSynchronously // HtmlDocument option []
27 |> Array.map getLinks // seq<HtmlNode> []

```

Static typing

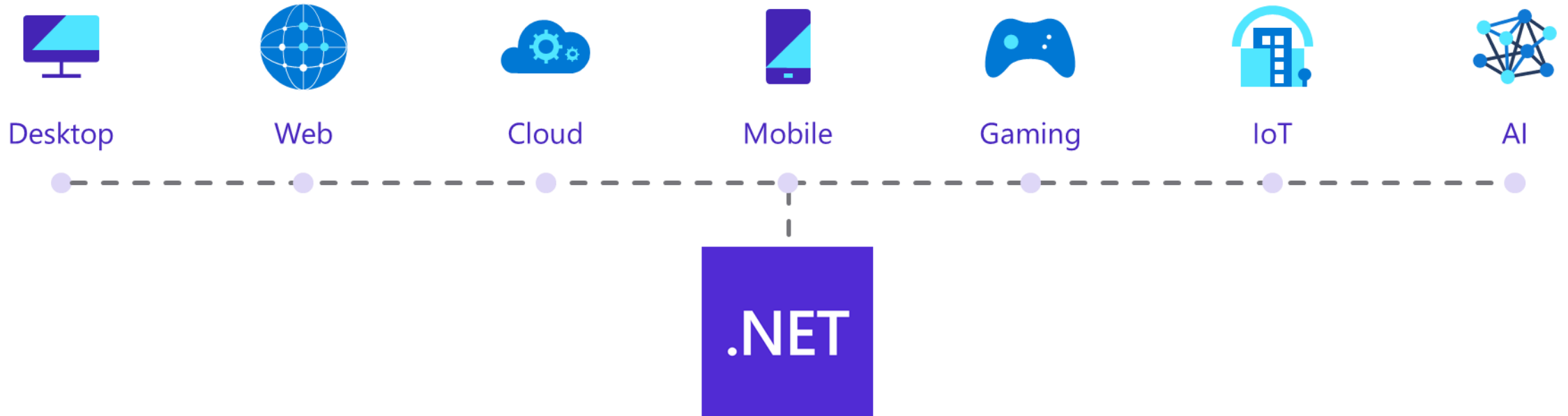
Type inference

Pattern matching

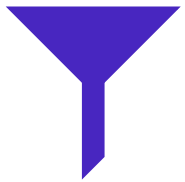
List comprehension / generator

Concurrency & parallelism built-in

Why .NET?



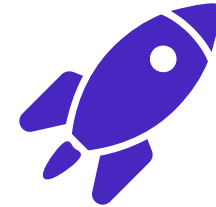
Data analytics & ML landscape .NET



Prepare Data



Modeling



Deploy



ONNX



Demo

End-to-End Machine Learning Workflow

Resources

<https://www.theurlist.com/dotnet-conf-fsharp-ml>

Questions?