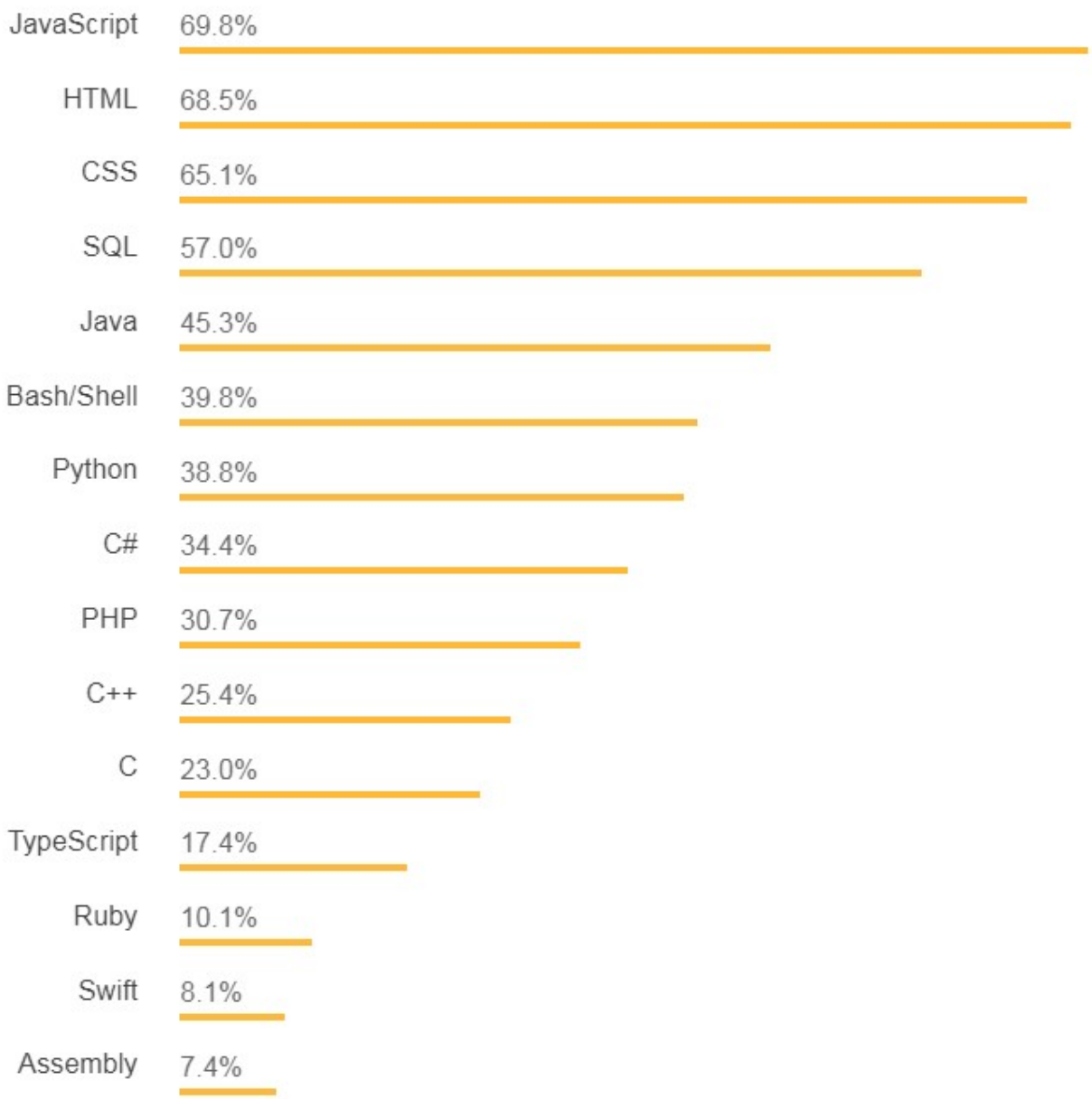
.NET for Apache Spark 预览版正式发布

2019年4月25日，微软的 Rahul Potharaju、Terry Kim 以及 Tyson Condie 在 Spark + AI Summit 2019 会议上为我们带来主题为 《Introducing .NET Bindings for Apache Spark 》的分享，并宣布 .NET for Apache Spark 预览版正式发布。

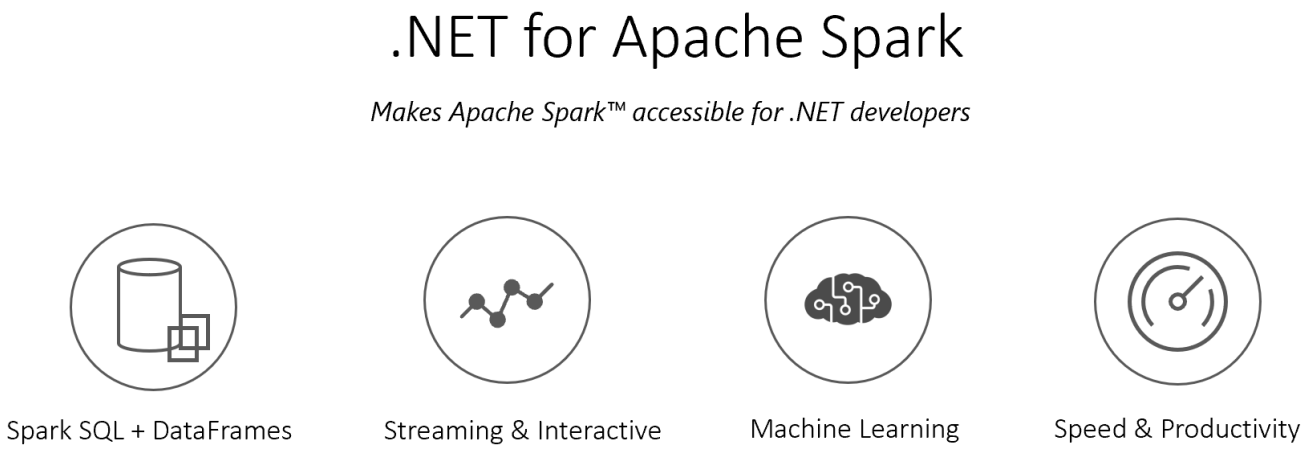
.NET 框架是由微软开发，一个致力于敏捷软件开发、快速应用开发、平台无关性和网络透明化的免费软件框架，用于构建许多不同类型的应用程序。就当前的编程语言排名可以看出，.NET 也是世界上使用人数最多的编程语言之一。其旗舰编程语言 C# 在各种文章和统计数据中被列为最受欢迎的编程语言之一：



如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公众号：iteblog_hadoop

从上图可以看出，C# 在 stackoverflow 调查的最流行编程语言中排名第八位，具体可以参见 这里 。同时，C# 在 2018年 GitHub 最流行的编程语言中排名第六位，参见 这里 。虽然有这么多的开发者使用 C#，但是目前没有很好的大数据解决方案，基于这些问题，微软为我们带来了 .NET for Apache Spark。

很明显，.NET for Apache Spark 的目标就是使得 .NET 的开发者们能够使用到 Apache Spark 的所有 API，因为目前 Apache Spark 仅仅支持 Scala, Java, Python 以及 R 编程语言。微软最近几年在开源项目上做了很多贡献，所以 .NET for Apache Spark 当然也是开源的项目（项目地址：https://github.com/dotnet/spark），不过是基于 MIT 许可证发行。

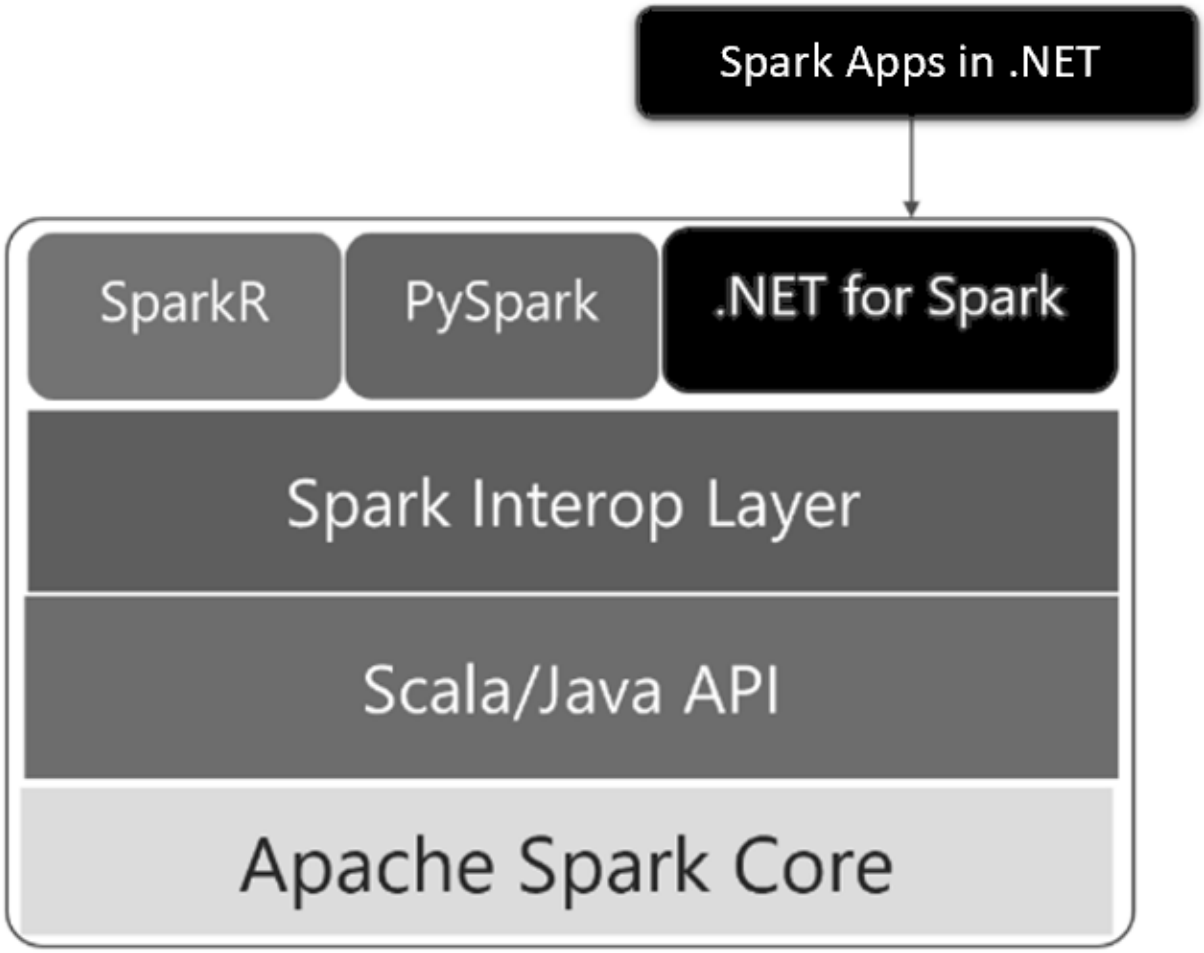


如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公众号：iteblog_hadoop

.NET for Apache Spark 是什么

.NET for Apache Spark 为 C# 或 F# 的开发者提供了高性能的 API 来访问 Apache Spark。 使用此 .NET API，用户可以访问 Apache Spark 的所有组件，包括 Spark SQL，DataFrames，Streaming，MLLib 等。并且这个项目允许 .NET 开发者重用已有的所有知识，技能，代码和库。

让 Spark 支持 C#/F# 是基于一个新的 Spark 互操作层（interop layer），这个层提供了更容易的扩展性。从长远来看，这种可扩展性可用于在 Spark 中添加对其他语言的支持。具体可以参见 SPARK-26257。.NET for Apache Spark 的具体框架如下：



如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公众号：iteblog_hadoop

.NET for Apache Spark 符合 .NET Standard 2.0，可以在 Linux，macOS 和 Windows 上使用，就像 .NET 的其余部分一样。 .NET for Apache Spark 在 Azure HDInsight 中默认可用，并且可以安装在 Azure Databricks 等中。

使用 .NET for Apache Spark

.NET for Apache Spark 的使用之前需要安装一些软件，具体参见 这里。这样我们就可以使用 C# 或 F# 来编写 Spark 应用程序了，下面是分别使用 C# 和 F# 编写的 WordCount 程序：
C# 版本的 WordCount

```
// Create a Spark session
var iteblog_spark = SparkSession
    .Builder()
    .AppName("word_count_sample")
```

```
    .GetOrCreate();

// Create a DataFrame
DataFrame dataFrame = iteblog_spark.Read().Text("input.txt");

// Manipulate and view data
var words = dataFrame.Select(Split(dataFrame["value"], " ").Alias("words"));

words.Select(Explode(words["words"])
    .Alias("word"))
    .GroupBy("word")
    .Count()
    .Show();
```

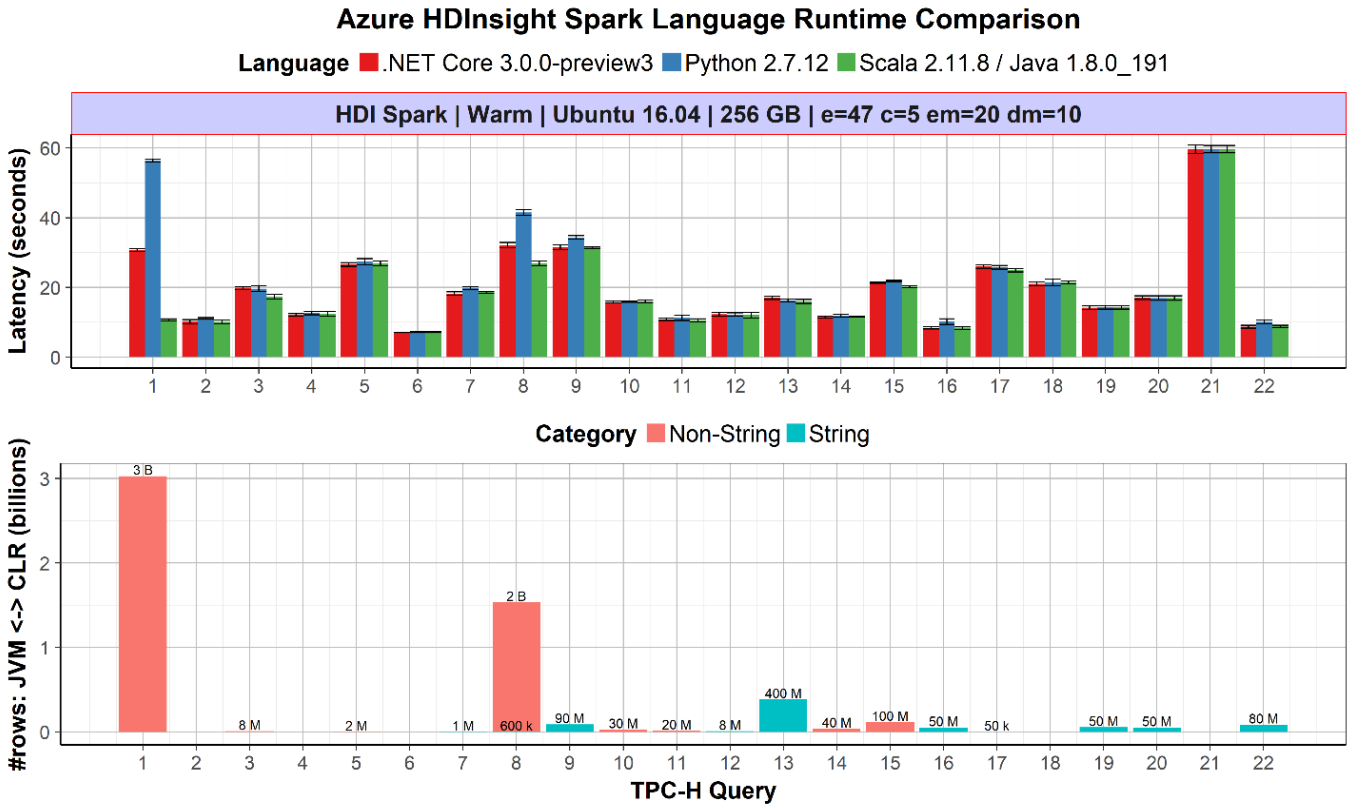
F# 版本的 WordCount

```
// Create a Spark session
let iteblog_spark =
    SparkSesstion.Builder()
        .AppName("word_count_sample")
        .GetOrCreate()

// Create a DataFrame
let df = iteblog_spark.Read().Text("input.txt")

let words = df.Select(Split(df.["value"], " ").Alias("words")

words.Select(Explode(words["words"]).Alias("word"))
    .GroupBy("word")
    .Count()
```

可以看出，这个和 Spark 原生的 API 还是很类似的。

.NET for Apache Spark 性能如何

经过微软官方的测试，.NET for Apache Spark 的第一个预览版本在流行的 TPC-H 基准测试中表现良好。TPC-H 基准包含一套面向业务的查询。 下图说明了在 TPC-H 查询集上 .NET Core 与 Python 和 Scala 的性能对比。



QT	UDF	QT	UDF	QT	UDF	QT	UDF	QT	UDF
1	Numeric	6	None	11	Numeric	16	Numeric | Regex	21	None
2	None	7	Numeric | String	12	Numeric | String	17	Numeric	22	Regex | String
3	Numeric	8	Numeric | String	13	Regex	18	None		
4	None	9	Numeric | String	14	Numeric | String	19	Numeric | Regex		
5	Numeric	10	Numeric	15	Numeric	20	String		

如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公众号：iteblog_hadoop

上图显示了 .NET for Apache Spark 与 Python 和 Scala 的每个查询性能对比。.NET for Apache Spark 相比较于 Python 和 Scala 运行良好。 此外，在 UDF
性能至关重要的情况下，例如查询1，其中在 JVM 和 CLR 之间传递30亿行非字符串数据，.NET for Apache Spark 比 Python 快2倍。

目前 .NET for Apache Spark 正在快速发展中，后续发展可以参见这里。