The Microsoft .NET Framework was from the beginning designed to support broad range of languages on a Common Language Runtime (CLR). CLR provides shared services such as garbage collection, JIT and tools integration. The other benefit is that these languages can work together and use libraries written in any of them as well as .NET Base class library (BCL).

The CLR didn't have the support for dynamic languages. Their dynamic nature makes the compilation uneasy and places high demands on the language runtime. Unlike static languages as C# which don't require runtime support other than CLR itself. How difficult was it to make the dynamic language on .NET can be seen in the open-source implementation of PHP language on .NET called Phalanger. Its code is really complex and hard to survey. This is a serious problem for an open-source project, because it's hard to contribute.

The new Dynamic Language Runtime (DLR) makes a difference. It adds a lot of support for dynamic languages on .NET, that makes implementing the dynamic languages much easier and it also enables the interoperability between the dynamic languages built on DLR and standard static languages on .NET.

This work focuses on features of PHP dynamic language and discusses how they can be implemented in DLR. A part of this work is a pilot implementation of PHP language on DLR; the target of this implementation is to prove some new concepts, find advantages and disadvantages that DLR brings and serves as an example for implementing the dynamic language on DLR.