**Microsoft Machine Learning Server**

It includes a collection of packages, interpreters, and infrastructure for developing and deploying **R** **and Python**-based machine learning and data science solutions on a range of platforms

**Server Side:**

Available on premise (Supported Platforms):

Linux (Ubuntu), Windows (use server OS), Hadoop

available in the cloud as follows:

* [Machine Learning Server as preconfigured Azure virtual machine on Linux](https://docs.microsoft.com/en-us/machine-learning-server/install/machine-learning-server-azure-vm-on-linux)
* [Machine Learning Server as preconfigured Azure virtual machine on Windows](https://docs.microsoft.com/en-us/sql/advanced-analytics/r/provision-the-r-server-only-sql-server-2016-enterprise-vm-on-azure)
* [Machine Learning Server on Azure HDInsight](https://docs.microsoft.com/en-us/machine-learning-server/install/machine-learning-server-on-azure-hdinsight) (Hadoop, Spark)
* [R Server on the Microsoft Data Science Virtual Machine](https://docs.microsoft.com/en-us/machine-learning-server/install/r-server-vm-data-science)

Note: Machine Learning Server is coming soon on the Data Science VM.

**Client Side:**

Microsoft R Client is a free, data science tool for high performance analytics. R Client is built on top of Microsoft R Open so you can use any open source R packages to build your analytics. Additionally, R Client introduces the [powerful ScaleR technology](https://docs.microsoft.com/en-us/machine-learning-server/r/tutorial-revoscaler-data-import-transform) and its proprietary functions to benefit from parallelization and remote computing.

R Client allows you to work with production data locally using the full set of ScaleR functions, but there are some constraints. On its own, the data to be processed must fit in local memory, and processing is capped at two threads for RevoScaleR functions.

To benefit from disk scalability, performance and speed, push the compute context using rxSetComputeContext() to a production instance of Microsoft R Server (or R Server) such as [SQL Server Machine Learning Services](https://msdn.microsoft.com/en-us/library/mt604845.aspx) and Machine Learning Server for Hadoop.

(You can offload heavy processing to Machine Learning Server or test your analytics during their developmentYou by running your code remotely using [remoteLogin() or remoteLoginAAD()](https://docs.microsoft.com/en-us/machine-learning-server/r/how-to-execute-code-remotely) from the mrsdeploy package.)

The ability to switch a compute context means that you can push execution to an interpreter on another machine. Local is the default. Switching to a remote compute context is typically done to get better performance

**Components of R Server/ R Client:**

* R support includes [Microsoft R Open 3.4.1](https://mran.microsoft.com/open/), which is based on R-3.4.1. Python support is based on Anaconda 4.2 over Python 3.5

…

Operationalization refers to the process of publishing R models and code to Microsoft R Server in the form of web services and the consumption of these services within client applications to affect business results.

**Listed Pros/ Concepts**

# *Distributed and parallel computing* (breakdown of a complicated computation into pieces that can be performed independently, while maintaining a framework that allows for the results of those independent computations to be pulled together to create the final result.)

# *Web services:* R and Python code and models can be deployed as web services. models and code can be accessed and consumed in R, Python, programmatically using REST APIs, or using Swagger generated client libraries. Web services can be deployed from one platform and consumed on another. They can be consumed synchronously, in realtime, or in batch mode. Requirement! Before you can deploy and work with web services, you must have access to a Machine Learning Server instance [configured to host web services](https://docs.microsoft.com/en-us/machine-learning-server/operationalize/configure-start-for-administrators#configure-server-for-operationalization). two types of web services: standard and realtime.

# TO Do: Continue here: <https://docs.microsoft.com/en-us/machine-learning-server/r/concept-what-is-compute-context>

# Also look at Link Gabriel

# https://blogs.technet.microsoft.com/machinelearning/2017/09/25/introducing-microsoft-machine-learning-server-9-2-release/

# Good description at Concepts -> operationalization

# And draw picture of architecture for Joel