What is pickling?

Pickle is used for serializing and de-serializing Python object structures, also called marshalling or flattening. Serialization refers to the process of converting an object in memory to a byte stream that can be stored on disk or sent over a network. Later on, this character stream can then be retrieved and de-serialized back to a Python object. Pickling is not to be confused with compression! The former is the conversion of an object from one representation (data in Random Access Memory (RAM)) to another (text on disk), while the latter is the process of encoding data with fewer bits, in order to save disk space.

What Can You Do With pickle?

Pickling is useful for applications where you need some degree of persistency in your data. Your program's state data can be saved to disk, so you can continue working on it later on. It can also be used to send data over a Transmission Control Protocol (TCP) or socket connection, or to store python objects in a database. Pickle is very useful for when you're working with machine learning algorithms, where you want to save them to be able to make new predictions at a later time, without having to rewrite everything or train the model all over again.

When Not To Use pickle?

If you want to use data across different programming languages, pickle is not recommended. Its protocol is specific to Python, thus, cross-language compatibility is not guaranteed. The same holds for different versions of Python itself. Unpickling a file that was pickled in a different version of Python may not always work properly, so you have to make sure that you're using the same version and perform an update if necessary. You should also try not to unpickle data from an untrusted source. Malicious code inside the file might be executed upon unpickling.

Pickle vs JSON

JSON stands for JavaScript Object Notation. It's a lightweight format for data-interchange, that is easily readable by humans. Although it was derived from JavaScript, JSON is standardized and language-independent. This is a serious advantage over pickle. It's also more secure and much faster than pickle.

However, if you only need to use Python, then the `pickle` module is still a good choice for its ease of use and ability to reconstruct complete Python objects.

An alternative is cPickle. It is nearly identical to `pickle`, but written in C, which makes it up to 1000 times faster. For small files, however, you won't notice the difference in speed. Both produce the same data streams, which means that Pickle and cPickle can use the same files.