Cachepy

Cachepy is a framework for disk based caching of arbitrary python function calls. It has been conceived primarily with various forms of compilation caching in mind, but it should be useful in other contexts as well. The two obvious benefits are the elimination of duplicate work, and the elimination of dependencies, by generating a distributable cache.

Features:

* Caches are thread/process/NFS\* safe. Multiple processes will coordinate to execute the cached operation no more often than necessary.
* No possibility of hash collisions; a unique representation of the cached arguments is stored entirely, and hashes are only used to accelerate cache lookup
* Arbitrary key objects are allowed, and serialized in a unique and deterministic\* way. This makes caching of functions mapping complex datastructures to complex datastructures painless
* Caches will recompile given different function arguments, but will also optionally monitor the state of a given environment description, which may have relevance to the outcome of the cached process
* Easy to use API; either annotate functions\* or subclass CachedOperation
* Tools for distribution\* of caches. This may free the end user of cumbersome dependencies, such as compiler installations.
* Automatic cache cleanup\*: The cost of executing the cached operation is monitored, and in conjuction with usage statistics of cache entries, a disksize target is maintained.

\*planned/WIP features

Note that cachepy is similar in intent to joblib; but considering it has somewhat different aims, a new project has been started. Both projects are sufficiently similar that ideally, the features of both should be merged into one package in the future. But for now, implementing the specific needs of compilation caching would clash with backwards compatibility in joblib.