The WATERS workflow is designed to characterize microbial populations. It produces phylogenetic trees, diversity metrics, Unifrac environment files, etc. as outputs in response to the inputs of one or more sequence libraries. The sequence libraries have various metadata commonalities, and diversity metrics are generated based on that metadata in addition to more inherent distinctions like sequence library.

Many of the steps in this workflow are computationally-intensive, and so it is beneficial to minimize the number of computations performed. The normal use of this workflow would be to run it on genetic library files A, and then, later in time, run it on genetic library files B, where B is a superset of A. This makes steps that run on a per-sequence basis quite worthwhile to cache.

The workflow aligns all the sequences, removes chimerical sequences then groups them by OTU and selects representative sequences. These sequences are then identified, and outputs are generated based off of them. Several steps, including tree generation and sequence alignment, have multiple interchangeable implementations.

Many external programs can run locally on the scientist's computer.

Due to the pipelined nature of COMAD, a great deal of tokens are being processed at any one time somewhere in the workflow. If the tokens are of non-negligible size, this can result in high memory usage and even running out of memory entirely. To circumvent this problem, information regarding full sequence data on demand.

The speed of database accesses is negligible, but memory usage of Kepler scales much better with larger datasets.

Running the workflow a second time on files B will perform the per-sequence and per-library operations on only those libraries newly introduced in B. The Group operation requires as input all the libraries, so it will have to be re-run on the entirety of B.

In Ptolemy's implementation of PN, queues are realized only when forced by a blocking write. This guarantees no write deadlocks and the queues use minimal memory. This approach has a drawback when used in a COMAD workflow with branching. If the default initial queue size of 1 is used, then only one token at a time will be sent from the actor directly before the branch point. As soon as one of the actors on a branch encounters a read scope match and begins executing, it will for a time stop processing tokens and will starve the other branch.

This work was supported by NSF awards IIS 0629846 (PPOD), IIS 0612326 (CHIP2), and OCI 0722079 (Kepler/CORE).