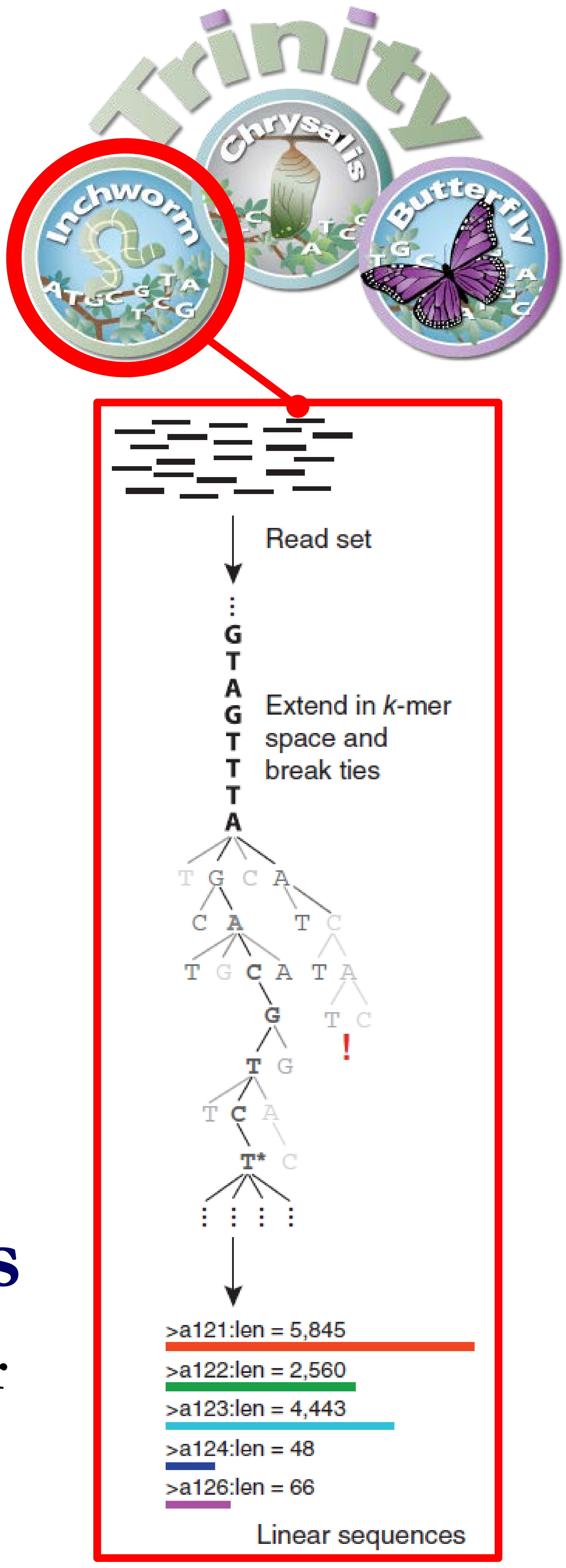
ACP in De Novo Transcript Assembly

- Concurrent *k*-mer Dictionary on Distributed Memory -

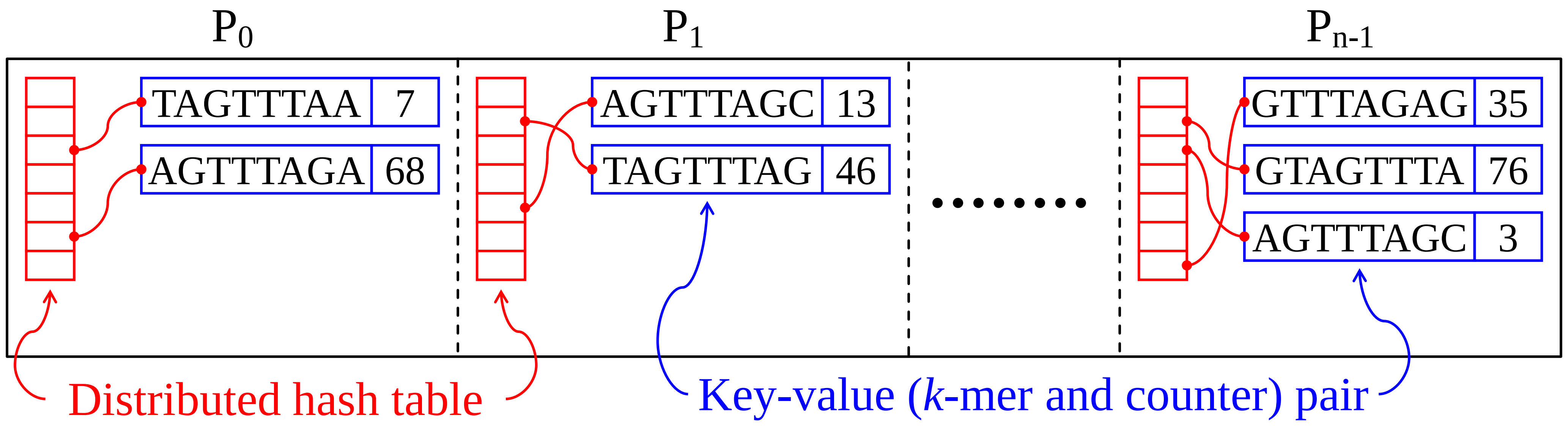
Overview

Trinity is a major de novo transcript assembly software, and Inchworm is the most memory consuming process of Trinity. To use distributed memory, we parallelized Inchworm using ACP.



Construction of *k*-mer dictionary

NGS short reads are split into small *k*-mers and stored into distributed dictionary implemented using map data-structure of ACP. Each entry of the dictionary holds the number of appearances.



Splicing *k*-mers to build long sequences

Inchworm splices *k*-mers by selecting the adjacent *k*-mer that has the highest number of appearances.
The connected *k*-mer is deleted from the dictionary.

Porting Inchworm to Python

Transcriptome data include numerous individual disconnected graphs. In the dictionary construction, there is an opportunity to place a *k*-mer to a process likely to splice it. Intending to implement such intelligence, we ported Inchworm to Python.

Live demo on a Raspberry Pi cluster

The memory efficiency of ACP enables not only exascale with brawny computing nodes, but also cluster computing with tiny computing nodes. Please visit our demonstration of Trinity on a Raspberry Pi cluster.

Advanced **C**ommunication Library for **E**xa Project