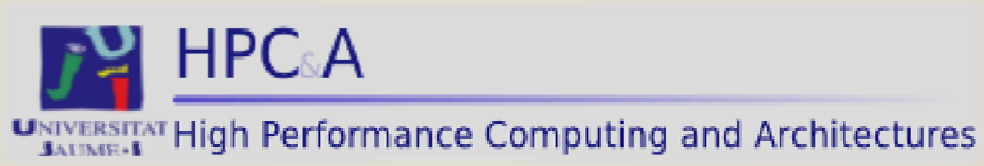
HPC SOLUTIONS FOR RNA-SEQ

Enrique S. Quintana-Ortí



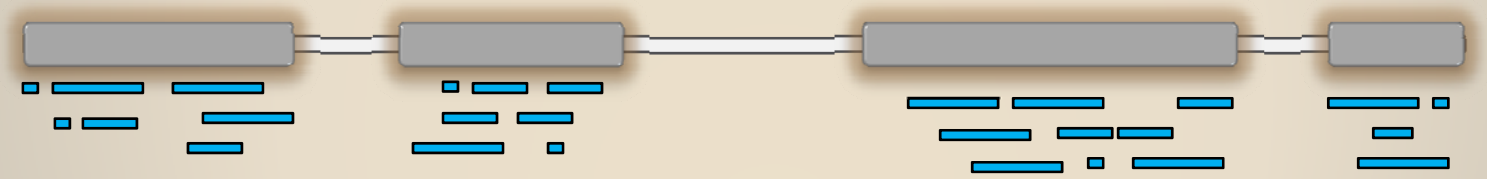
ONGOING COLLABORATION

- Ana Conesa
- Joaquín Dopazo
- Ignacio Medina
- Joaquín Tárraga

- Sergio Barrachina
- Maribel Castillo
- Héctor Martínez
- Enrique S. Quintana-Ortí

OVERVIEW RNA-SEQ

- Depending on technology, Next Generation Sequencing (NGS) provides millions of short sequence fragments (“reads”) with 35-150 (Illumina), 50-100 (SOLiD) or 400 (Roche) nts
- Objective: produce a genome-scale transcription map
- After sequencing, individual reads have to be mapped to the reference genome (proxy for transcriptome: complete set of transcripts in a cell, and their quantities)



OVERVIEW RNA-SEQ

- Mapping process delivers information on:
  - Gene/transcript/exon expression
  - Alternative splicing
  - Gene fusion
  - Nucleotide variations
  - Etc.

OVERVIEW RNA-SEQ

- Biology design issues/major problems of a mapper:
  - Allowed number of mismatches
  - Number of multi-hits
  - Exon unions (splice junctions)
  - Distance between exons

- Computer Science major problems:
  - Large collection of data (out-of-core)
  - Compute-intensive vs memory-bounded stages
  - Hybrid schemes of concurrency: data-parallelism, task-parallelism, loop-parallelism

STATE-OF-THE-ART FOR RNA-SEQ

- Short reads!
  - TopHat
  - MapSplice
  - SpliceMap
  - and dozens of others (BLASTN, Bowtie, ELAND, QPALMA…)

“Those are my principles, and if you don't like them... well, I have others.”

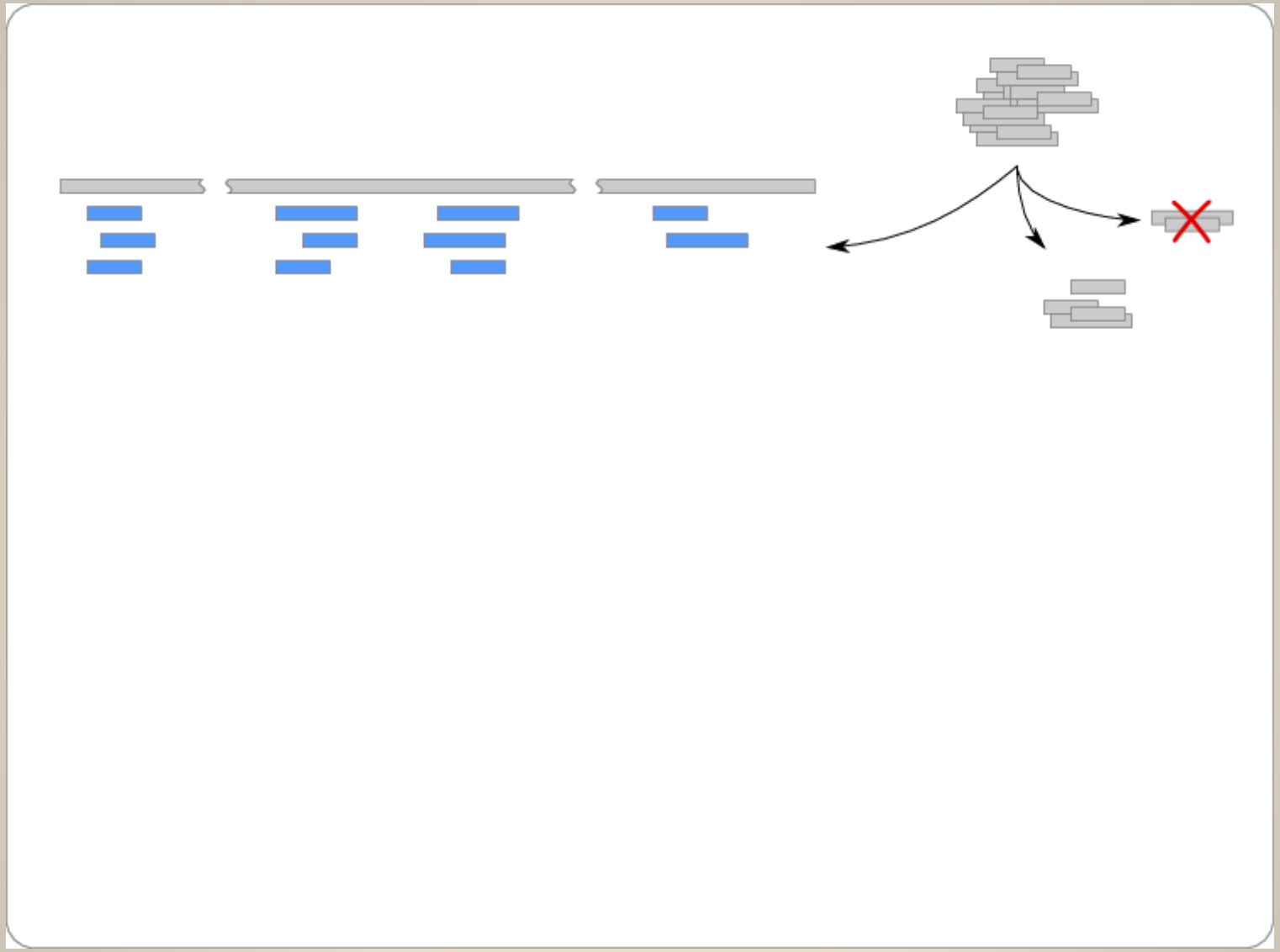
-- Groucho Marx

TOPHAT (http://tophat.cbcb.umd.edu)

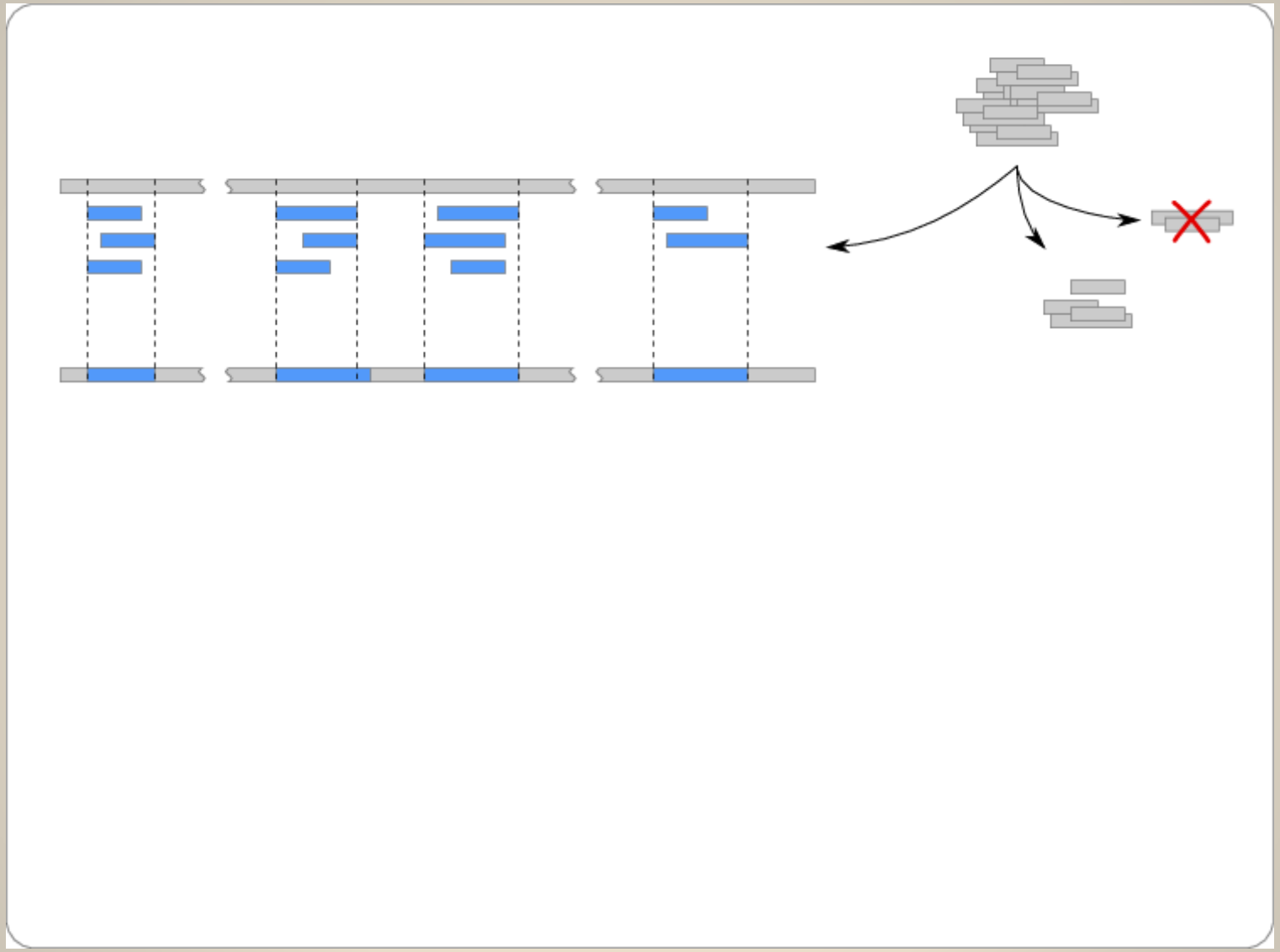
- Improved Bowtie with gap (intron) detection: ultrafast aligned with low memory consumption
- Align reads against reference genome detecting splice junctions
- Align segments with up to two mismatches
- Highly configurable

TOPHAT

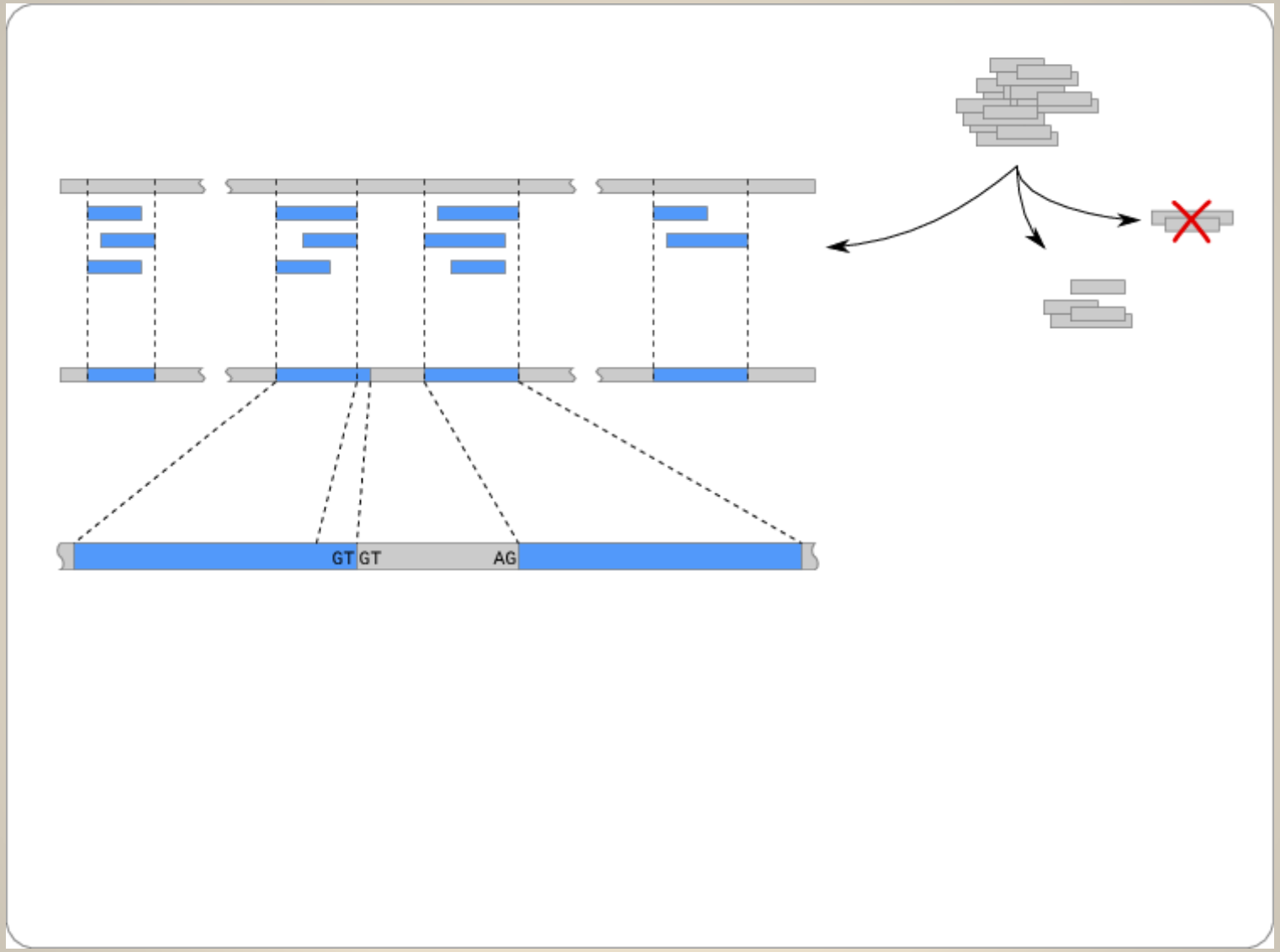
TOPHAT



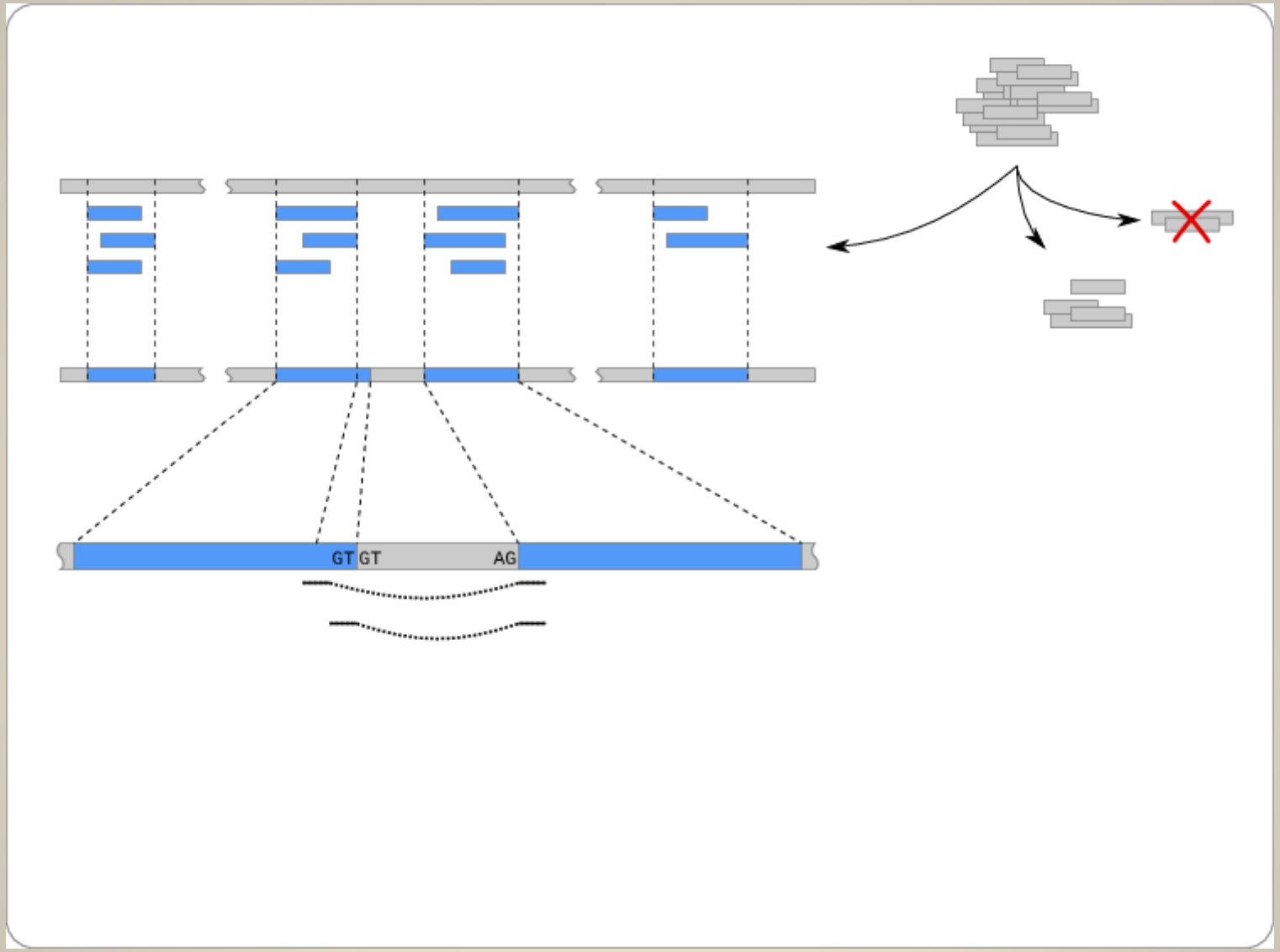
TOPHAT



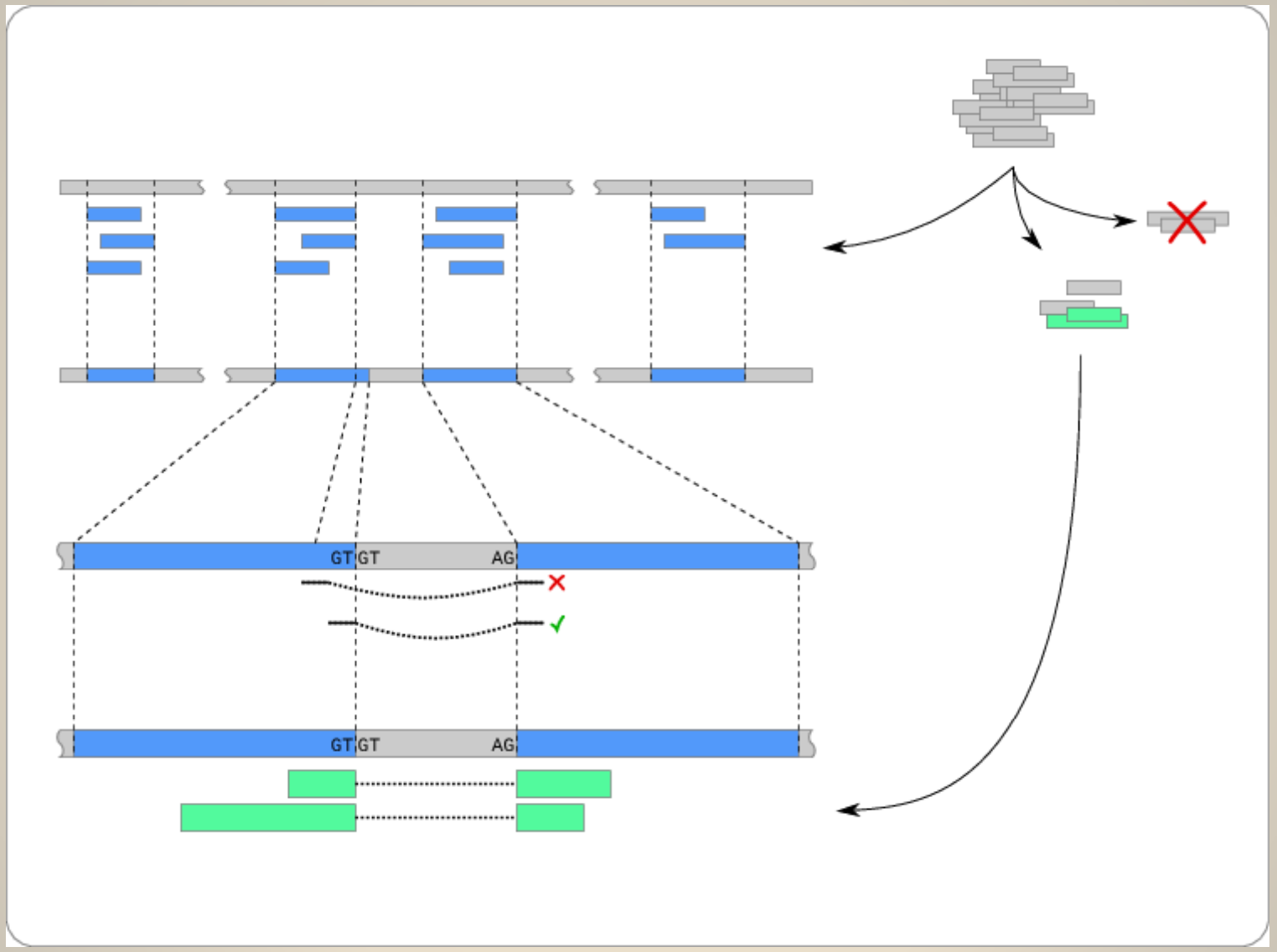
TOPHAT



TOPHAT

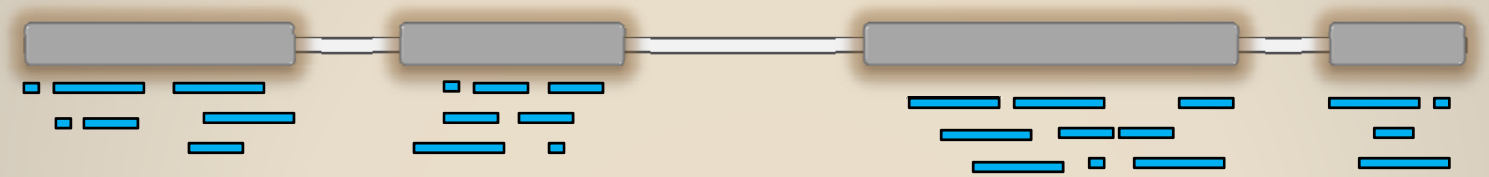


TOPHAT



TOPHAT

- Already parallelized for multicore processors: process reads concurrently, using one OpenMP thread per read



- Plenty of parallelism but… fast?

TOPHAT: EXPERIMENTATION

- Experimental setup (hardware)
  - 16 AMD Opteron(TM) Processor 6276 (2.3 GHz): 64 cores!
    - 4 Sockets
    - 2 nodes per socket
    - 4 cores per node
    - 2 CPUs per core
  - 6,144 Kbytes L3 cache (shared among all 4 cores in the same node)
  - 16 Gbytes main memory (split in 16 Gbytes modules, with one module per socket)

TOPHAT: EXPERIMENTATION

- Experimental setup (data and software)
  - Map chromosome 20 Homo Sapiens
  - File of reads in format FASTQ and Single End
    - 242 Mbytes
    - 99,9991 reads

- Experimental setup (software): tophat v2.0.0

```
tophat –p num_threads genome_index fastq_reads
```

TOPHAT: EXPERIMENTATION

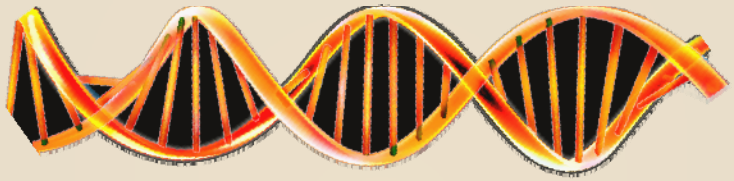
#Threads	Execution time	Speed-up
1	11.05 min.	-
2	7.23 min.	1.53
4	5.14 min.	2.15
8	4.28 min.	2.58

TOPHAT: EXPERIMENTATION

#Threads	Execution time	Speed-up
1	11.05 min.	-
2	7.23 min.	1.53
4	5.14 min.	2.15
8	4.28 min.	2.58
16	5.06 min.	2.18
32	17.45 min.	0.63
64	36.17 min.	0.31

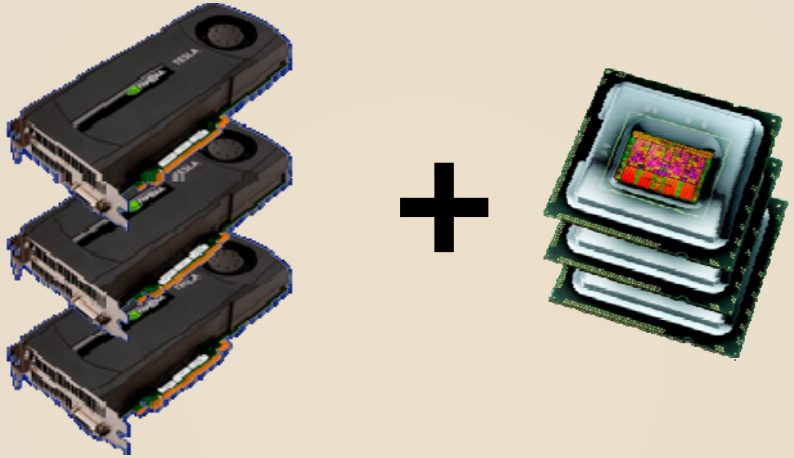
RNA-SEQ MAPPER

- Goals:
  - Map short reads to reference genome with splice junctions
  - High reliability
  - Fast
  - Generate files SAM/BAM for mapped reads and BED for splice junctions

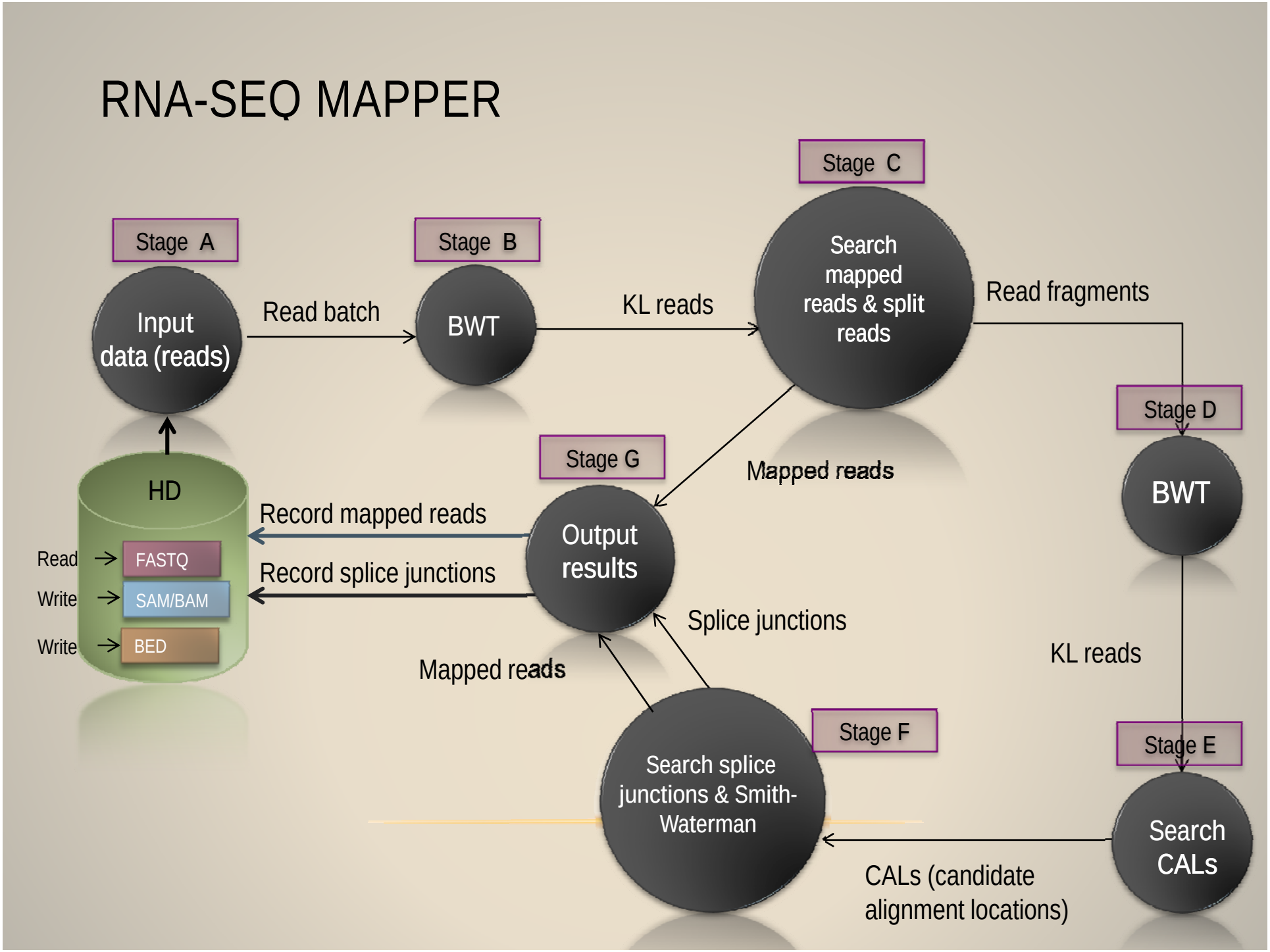


RNA-SEQ MAPPER

- "*Need for speed*"
  - Divide the problem into tasks/stages
  - Map tasks to different threads
  - Hybrid GPU-CPU computing



RNA-SEQ MAPPER



RNA-SEQ MAPPER

- Sequential processing:
  - One stage does not start till the previous one is completed
  - Consider, e.g.,

Stages	Units of time
A	1
B	2
C	1
D	2
E	3
F	2
G	1

RNA-SEQ MAPPER

- Sequential processing

Batches	U.T.																							
B1	A	B	B	C	D	D	E	E	E	F	F	G												
B2																								

RNA-SEQ MAPPER

- Sequential processing

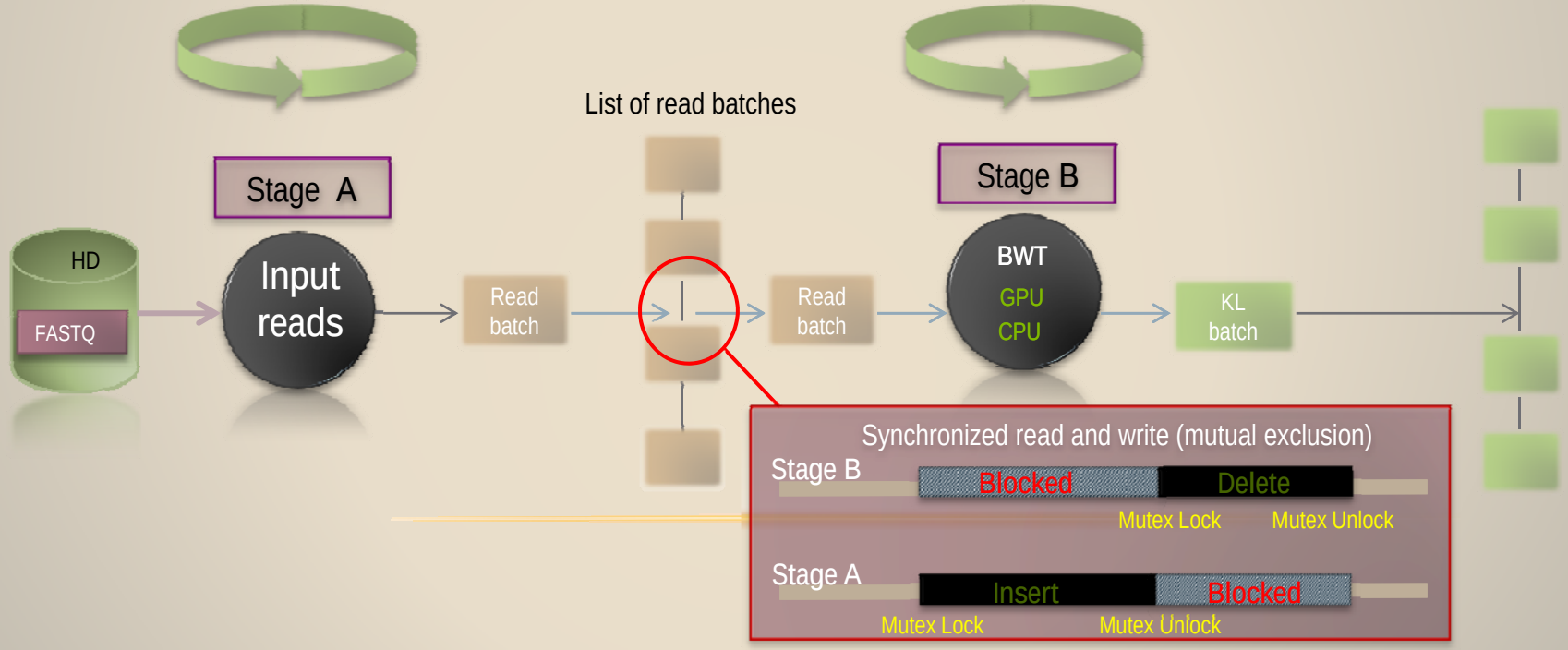
Batches	U.T.																							
B1	A	B	B	C	D	D	E	E	E	F	F	G												
B2													A	B	B	C	D	D	E	E	E	F	F	G

24 U.T.

RNA-SEQ MAPPER

- Pipelined (parallel) processing
  - Stages process new data as soon as possible
  - Needs careful synchronization of communication mechanisms

Batches	U.T.			
B1	A	B	B	
B2		A		



RNA-SEQ MAPPER

- Pipelined processing

Batches	U.T.																							
B1	A	B	B	C	D	D	E	E	E	F	F	G												
B2		A	X	B	B	C	D	D	X	E	E	E	F	F	G									

RNA-SEQ MAPPER

- Pipelined processing

Batches	U.T.																							
B1	A	B	B	C	D	D	E	E	E	F	F	G												
B2		A	X	B	B	C	D	D	X	E	E	E	F	F	G									

15 U.T.

- If there is a long list of batches, a new batch is processed every 3 U.T.
- Important to balance the cost of the stages

RNA-SEQ MAPPER

- More concurrency? Superscalar processing

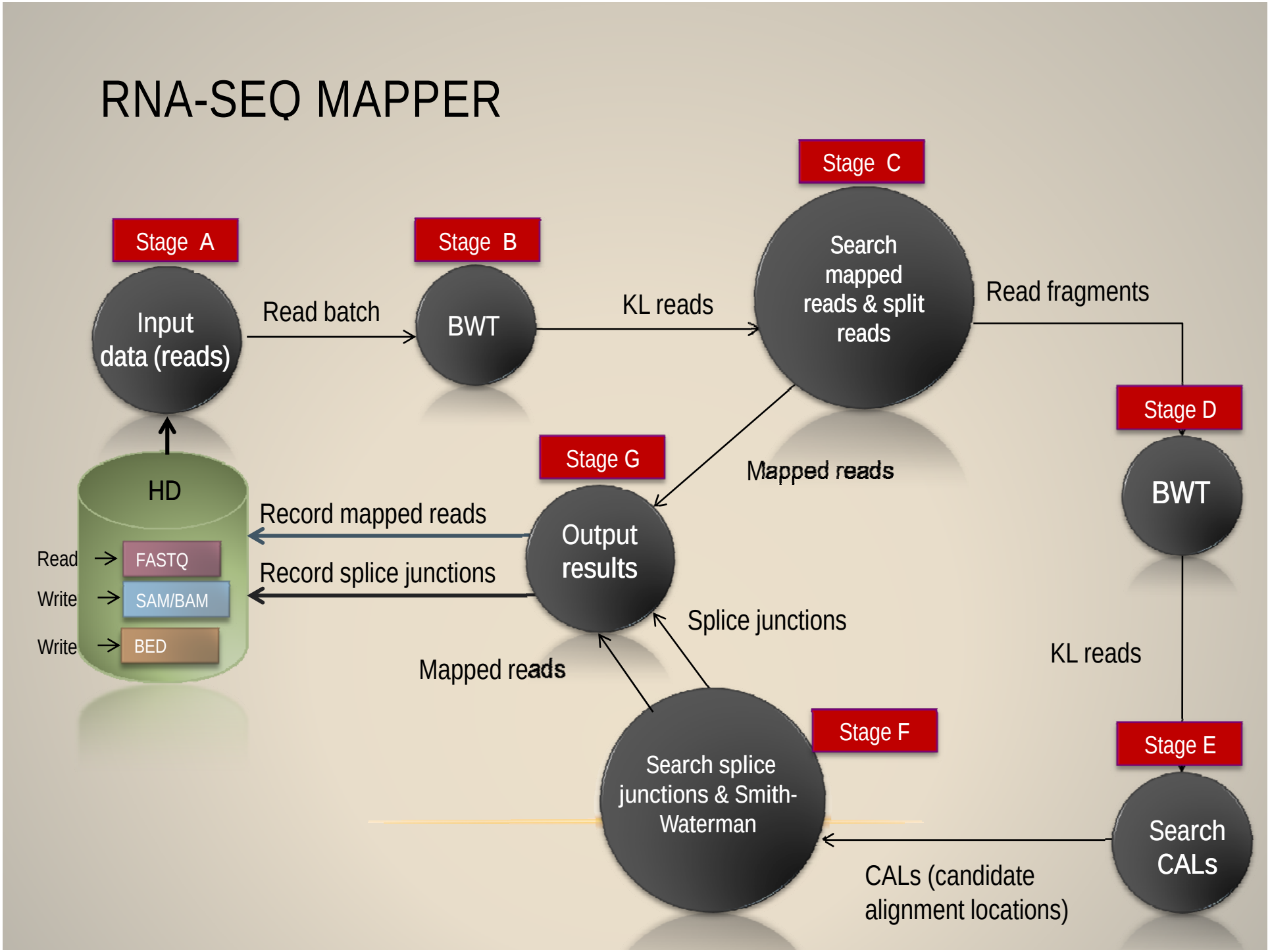
RNA-SEQ MAPPER

- More concurrency? Superscalar processing

Batches	U.T.																							
B1	A	B	B	C	D	D	E	E	E	F	F	G												
B2	A	B	B	C	D	D	E	E	E	F	F	G												
B3		A	X	B	B	C	D	D	X	E	E	E	F	F	G									
B4		A	X	B	B	C	D	D	X	E	E	E	F	F	G									

15 U.T.

- …but not all stages can proceed in parallel. Example: read from/write to disk
- Exploit intrastage parallelism

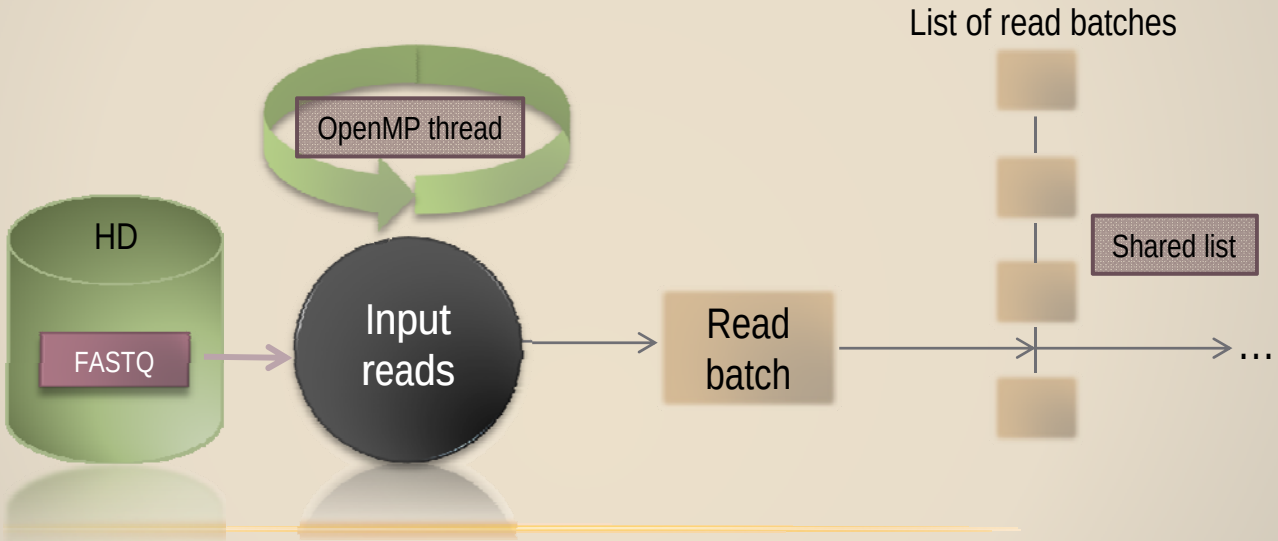

RNA-SEQ MAPPER
Stage A
Input data (reads)
Read batch
Stage B
BWT
KL reads
Stage C
Search mapped reads & split reads
Read fragments
Stage D
BWT
KL reads
Stage E
Search CALs
CALs (candidate alignment locations)
Stage F
Search splice junctions & Smith-Waterman
Mapped reads
Splice junctions
Stage G
Output results
Record mapped reads
Record splice junctions
Mapped reads
HD
Read
FASTQ
Write
SAM/BAM
Write
BED

RNA-SEQ MAPPER: "HD READER" THREAD

- Functions:
  - Input reads from HD to main memory, creating batches
  - Enable parallel processing of batches in main memory

RNA-SEQ MAPPER: "HD READER" THREAD

- Implementation:
  - A single OpenMP thread to input reads (bottleneck is disk, do not use more threads!)
  - Stores batches in a list, shared with the "**BWT Server**" thread
    - Decouple problem dimension-RAM capacity, prefetch, stream instead of disk,…

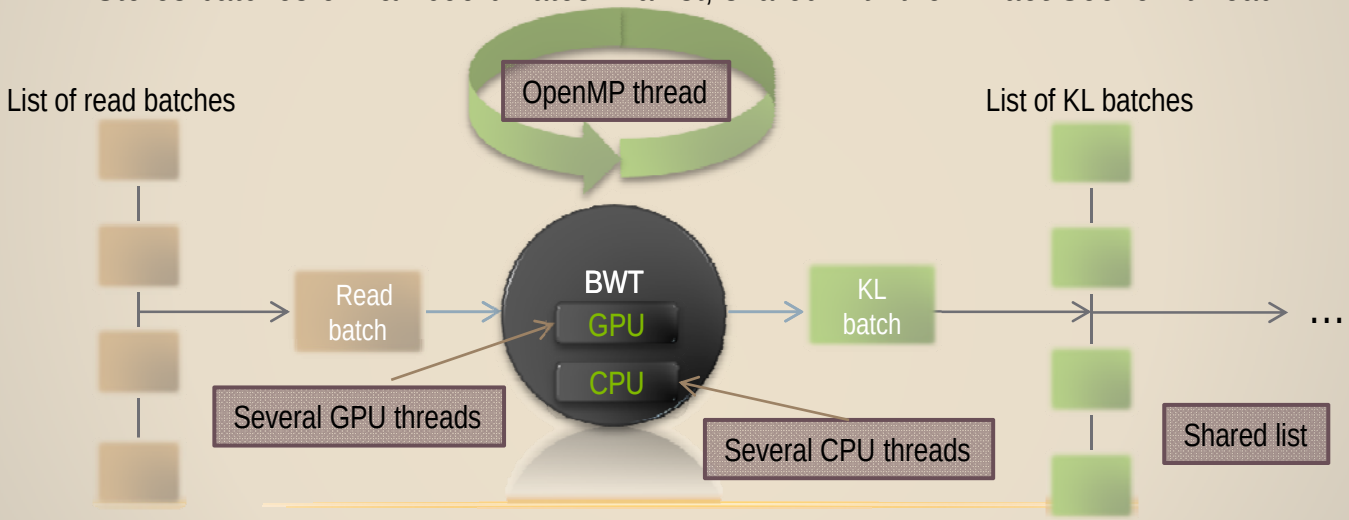


RNA-SEQ MAPPER: "BWT SERVER" THREAD

- Functions:
  - Input read batches (in main memory)
  - Apply BWT (CPU or GPU) to individual batches, possibly in parallel
  - Produce batches of K & L coordinates for mapped reads (coordinates)
  - Enable parallel processing of batches of K & L coordinates

RNA-SEQ MAPPER: "BWT SERVER" THREAD

- Implementation:
  - A single OpenMP thread as the BWT server
  - Multiple worker threads to apply BWT concurrently (parallel processing of reads)
    - CPU or GPU threads
  - Stores batches of K&L coordinates in a list, shared with the "**Exact Seeker**" thread

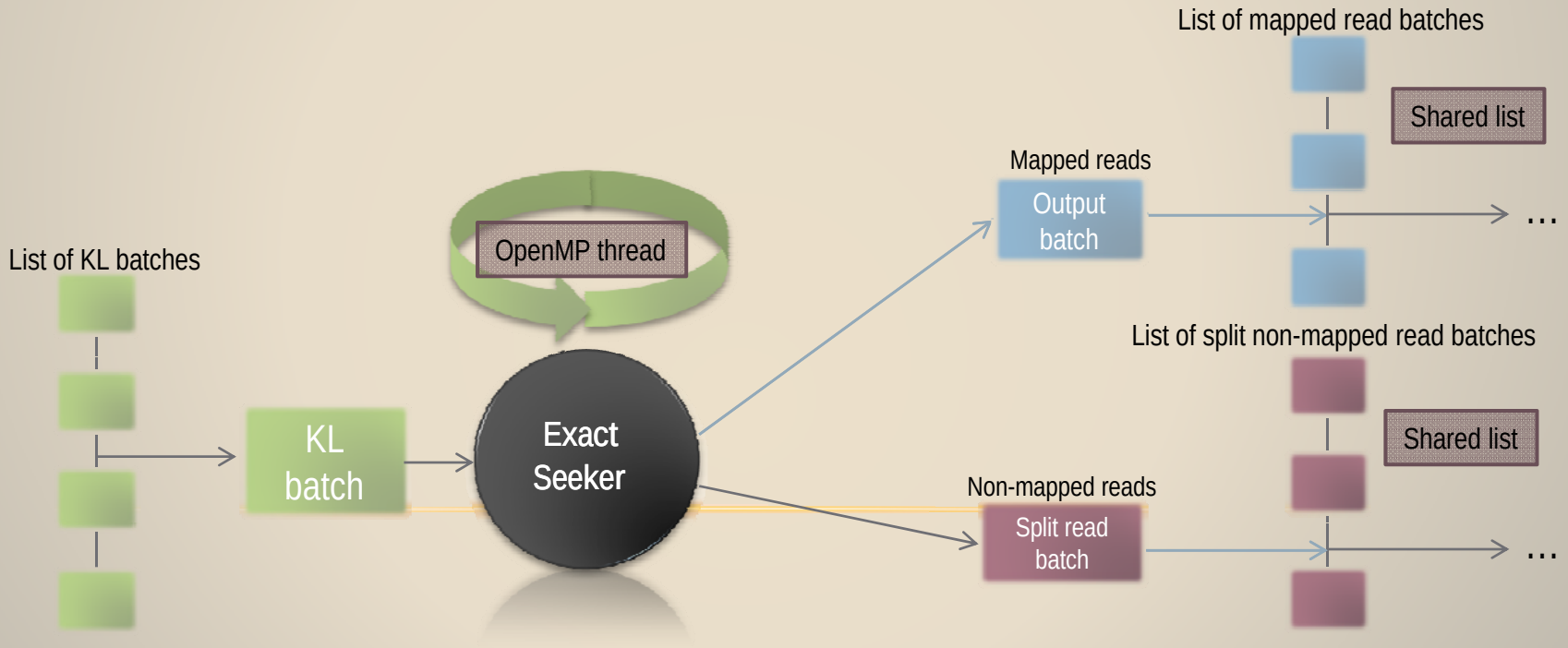


RNA-SEQ MAPPER: "EXACT SEEKER" THREAD

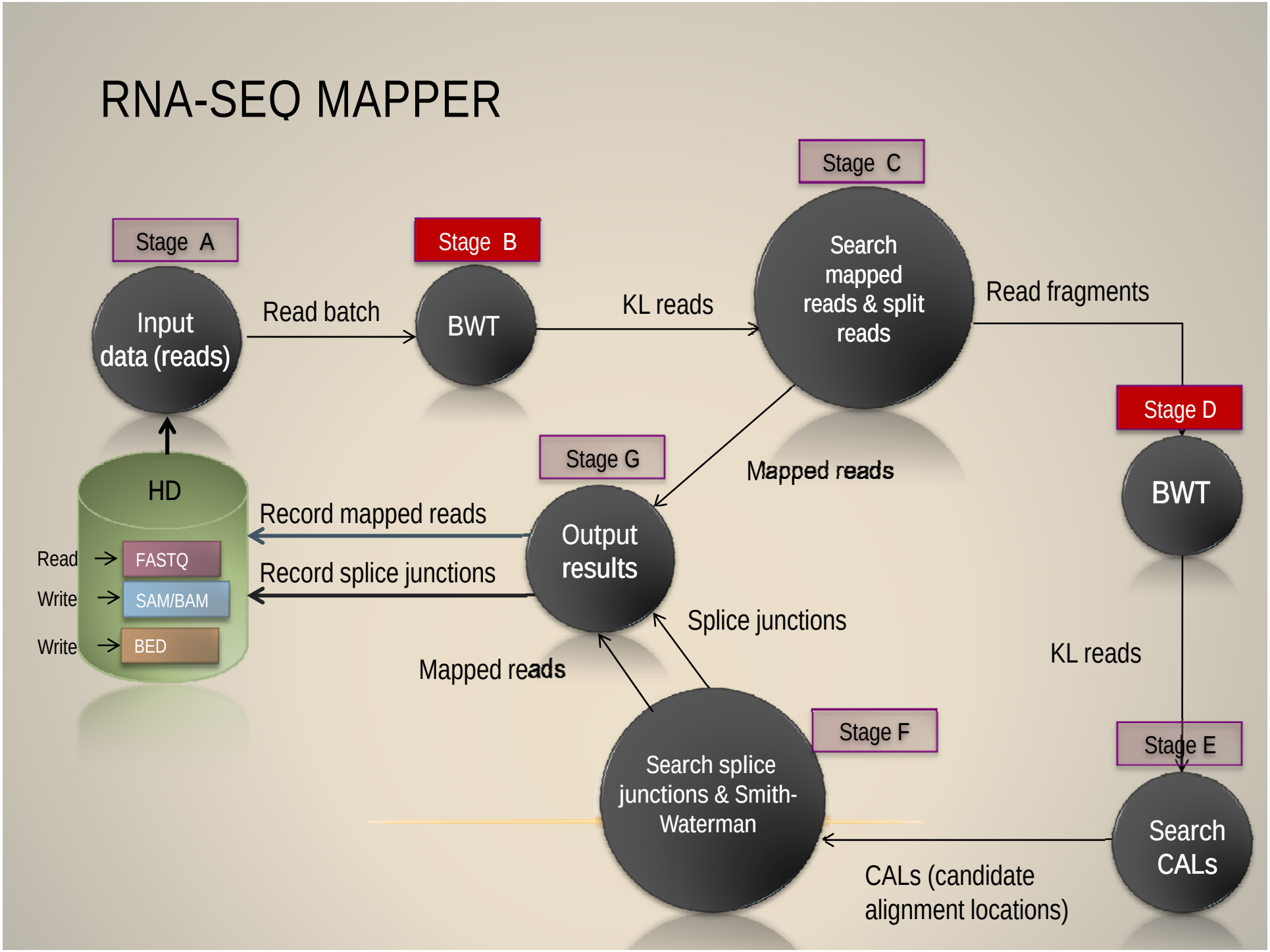
- Functions:
  - Input K & L batches
  - Obtain output batches with mapped reads
  - Split and store non-mapped reads
  - Enable parallel store of mapped reads in HD
  - Enable parallel processing of batches of split non-mapped reads

RNA-SEQ MAPPER: "EXACT SEEKER" THREAD

- Implementation:
  - A single thread as the Exact Seeker
  - Store split non-mapped reads in a list, shared with the "**BWT Seeker**" thread
  - Store mapped reads in a list, shared with the "**HD Writer**" thread



RNA-SEQ MAPPER

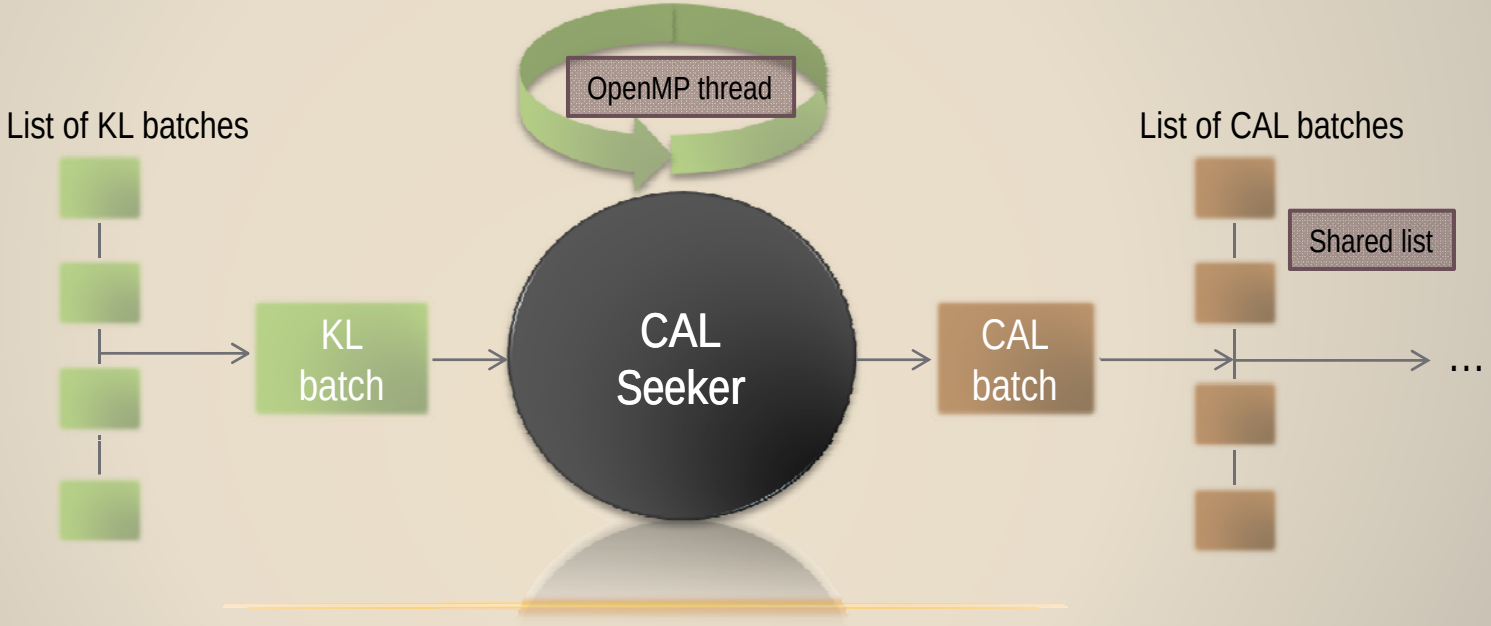


RNA-SEQ MAPPER: "CAL SEEKER" THREAD

- Functions:
  - Run multiple "CAL Seeker" theads
  - Read K & L batches
  - Generate batches of CALs
  - Enable parallel processing of batches of CALs

RNA-SEQ MAPPER: "CAL SEEKER" THREAD

- Implementation:
  - Run multiple OpenMP "**CAL Seeker**" threads (parallel processing of batches)
  - Store batches of CALs in a list, shared with the "**RNA server**" thread

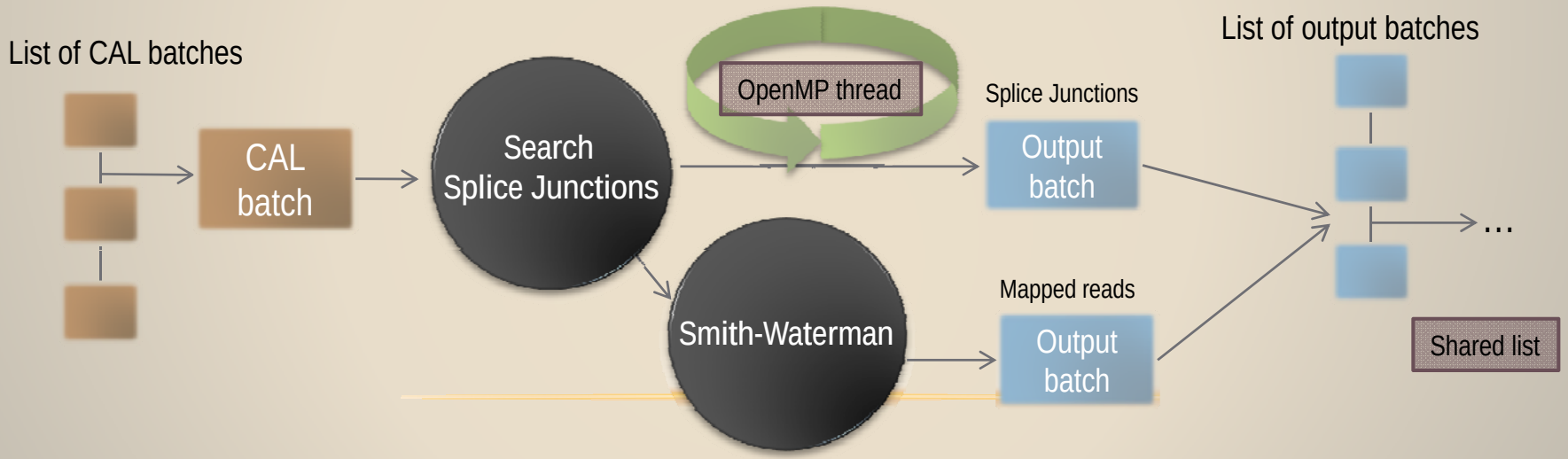


RNA-SEQ MAPPER: "RNA SERVER" THREAD

- Functions:
  - Run multiple "RNA Server" threads
  - Read CAL batches
  - Map reads to CALs and search for splice junctions
  - Create output batches with mapped reads
  - Create batches with splice junctions
  - Enable parallel output of batches of mapped reads and splice junctions to HD

RNA-SEQ MAPPER: "RNA SERVER" THREAD

- Implementation:
  - Run multiple OpenMP "RNA Server" threads (parallel processing of batches)
  - Search for intron marks (GT-AG, etc.) and splice junctions
  - Apply Smith-Waterman to CALs per read
  - Write mapped reads and splice junctions to list, shared with "**HD Writer**" thread

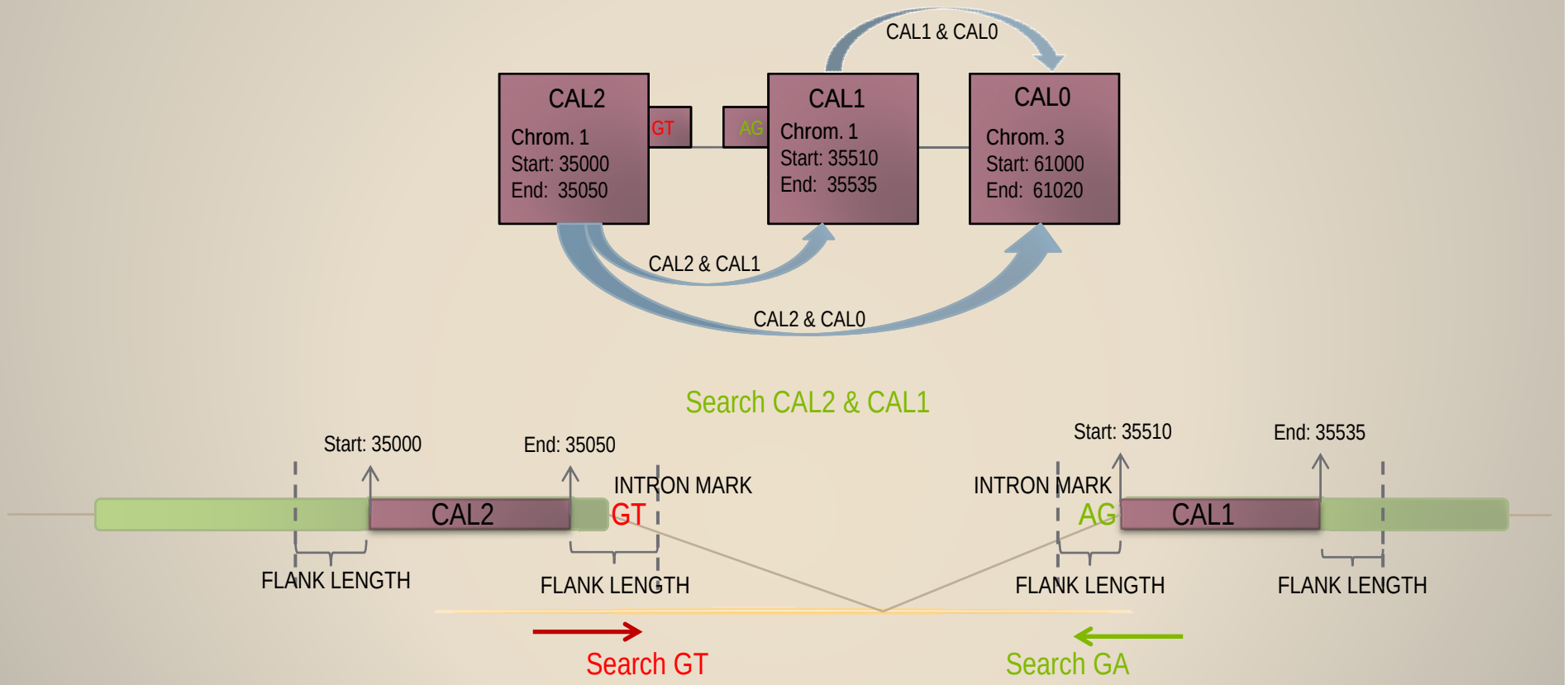


RNA-SEQ MAPPER: "RNA SERVER" THREAD

1. Read CALs for each read
2. To correctly search for GT-AG pairs (or other combinations), CALs must appear in order
   - Natural ordering: chromosome, start, end

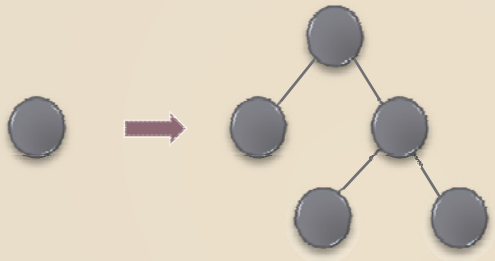
RNA-SEQ MAPPER: "RNA SERVER" THREAD

3. Search CALs with splice junctions

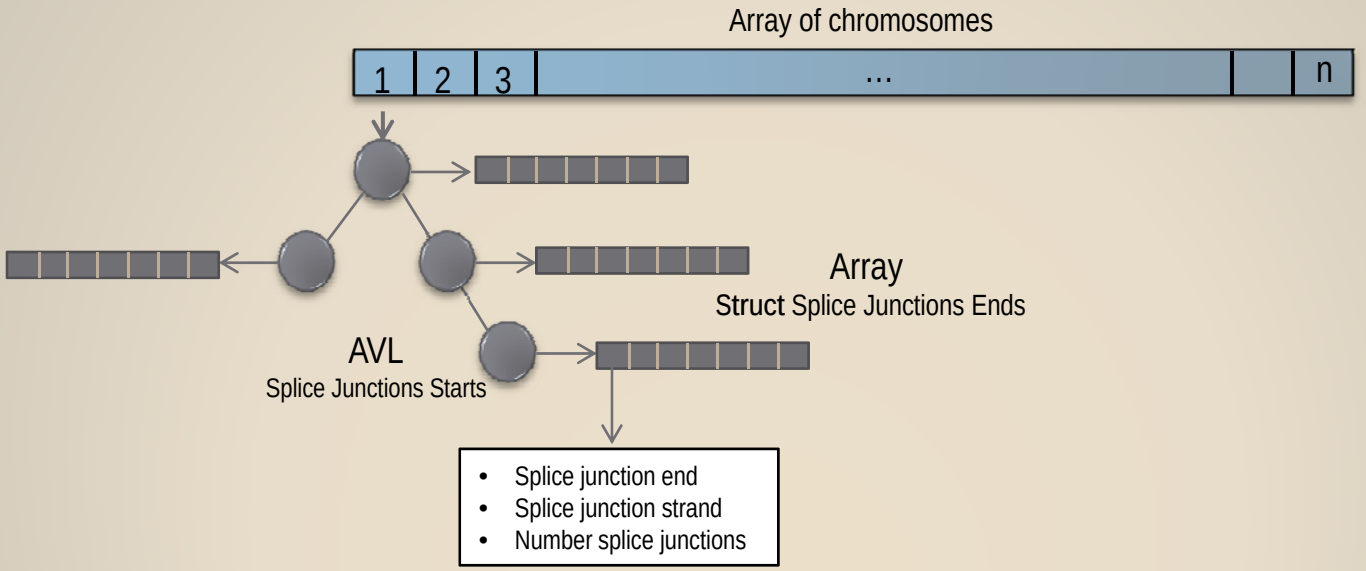


RNA-SEQ MAPPER: "RNA SERVER" THREAD

4. Record splice junctions
    - Objectives:
        - Store splice junctions in main memory, ordered according to start
        - Efficient search and insertion
    - Implementation:
        - Use AVLs (ordered insertion and search in O(log n))



RNA-SEQ MAPPER: "RNA SERVER" THREAD



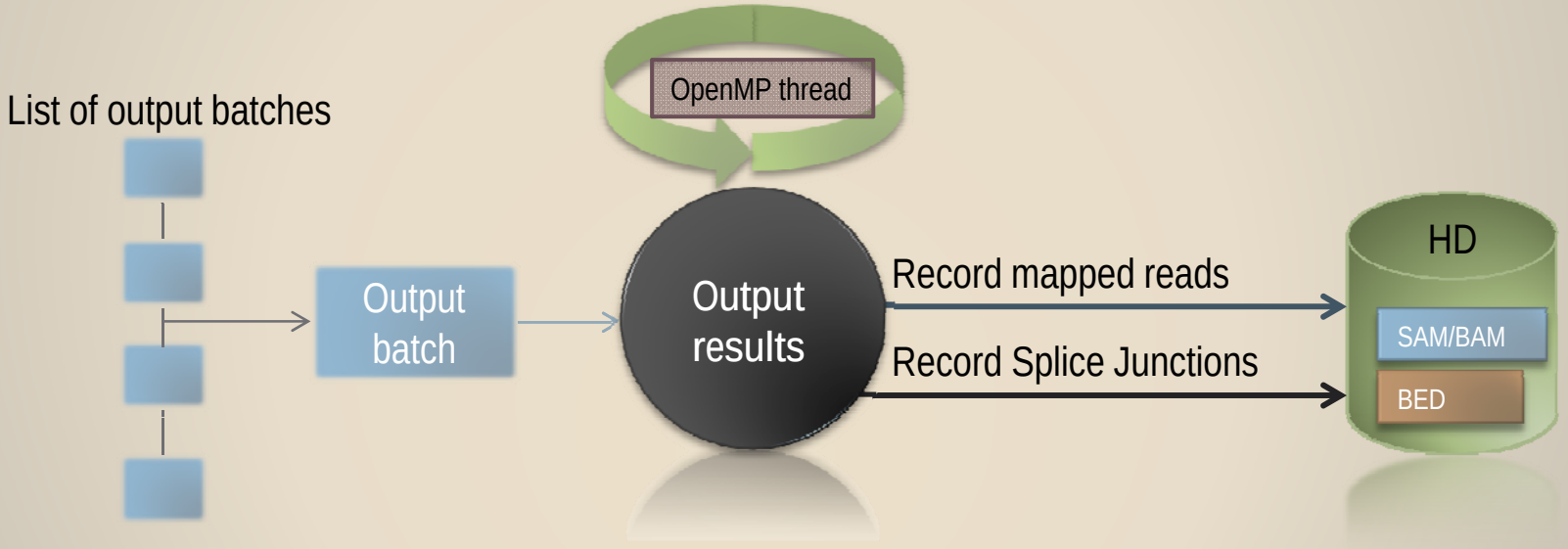
4. Smith-Waterman for each CAL of the read
5. Store results in output batches

RNA-SEQ MAPPER: "HD WRITER" THREAD

- Functions:
  - Read output batches
  - Write batches to the appropriate file, concurrently with data processing

RNA-SEQ MAPPER: "HD WRITER" THREAD

- Implementation:
  - A single OpenMP thread (bottleneck is disk, do not use more threads!)
  - Multiple streams of results: serialize writes



RNA-SEQ MAPPER: CURRENT STATUS

- Under development, with preliminary evaluation
  - Much faster than TopHat (one-two orders of magnitude)
  - Still not so “accurate”
  - Need to improve workload balance among CPU threads and GPU

RNA-SEQ MAPPER: SUMMARY

- Parallelization of RNA-SEQ mapper
  - CPU-bounded:
    - Data parallel, task parallel, loop parallel… an optimal solution will leverage a mixture of them!
    - GPU, multicore, SIMD, etc.
  - Memory-bounded (data intensive):
    - Network, disk, main memory, etc.

QUESTIONS?

- Ana Conesa
- Joaquín Dopazo
- Ignacio Medina
- Joaquín Tárraga

- Sergio Barrachina
- Maribel Castillo
- Héctor Martínez
- Enrique S. Quintana-Ortí