High-Level Synthesis of a Genomic Database Search Engine

Rasha Karakchi, Jordan Bradshaw, **Jason D. Bakos**

UNIVERSITY OF SOUTH CAROLINA

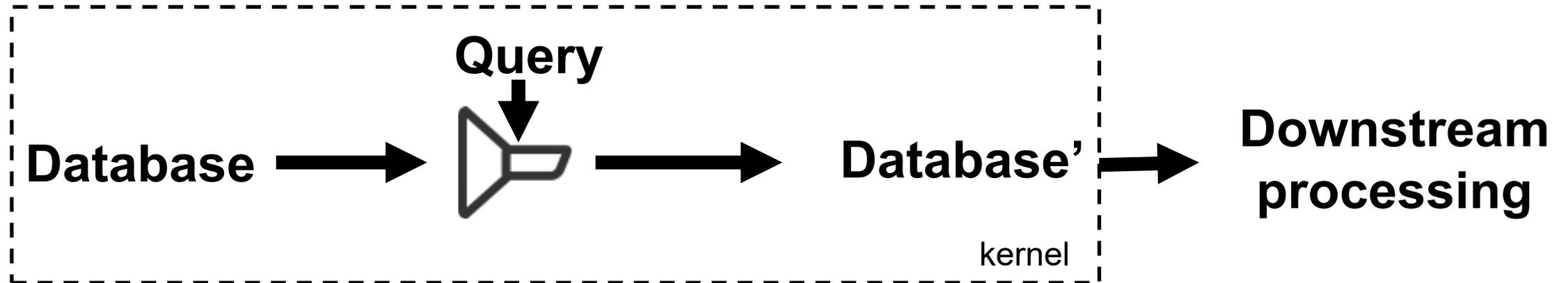Heterogeneous and Reconfigurable Computing Group

Motivation

- Genomic databases large and growing fast

- NCBI BLAST widely used for database search
  - Must scan perform entire database for each query
  - I/O (disk) bound

Background

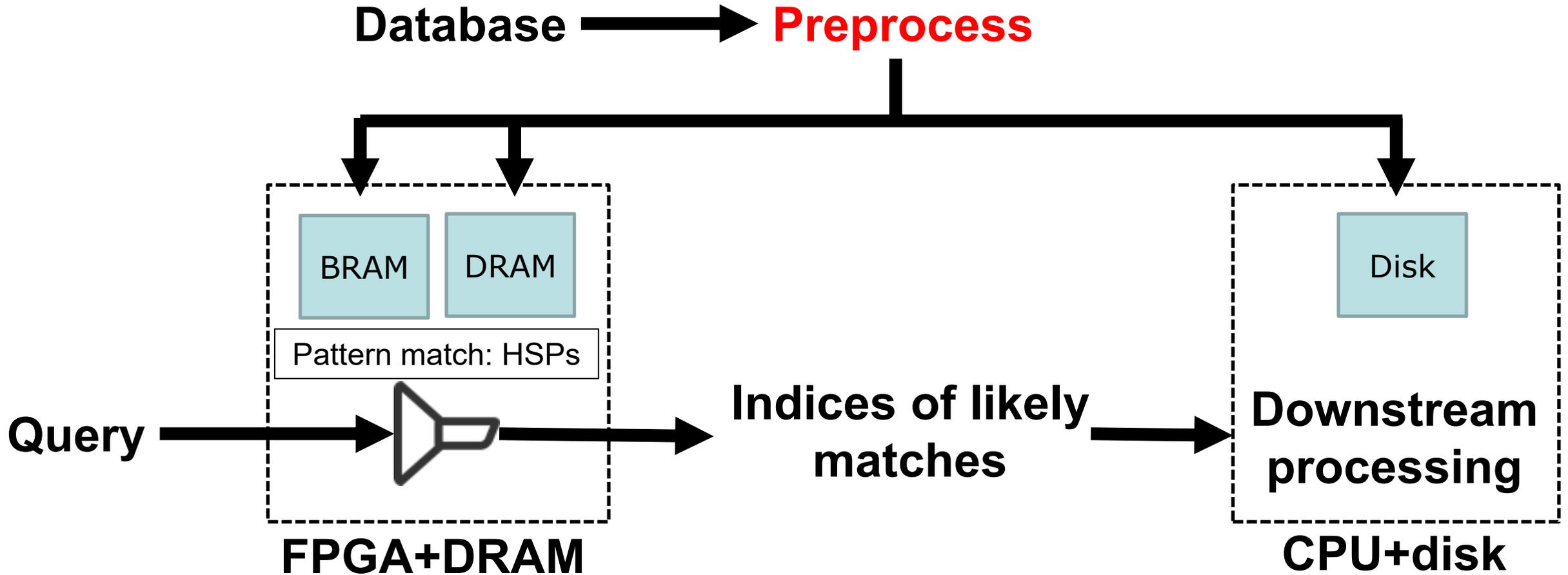## BLAST Filter:

1) Decompose **query**

2) Identify **seeds**

A A C B B A V ➡ | A A C | A C B | C B B | ~~B B A~~ | B A V |

3) Identify **High Scoring Pairs** (HSPs):

| A A C | A C B | C B B | ~~B B A~~ | B A V |

4) Look for HSPs of the same total length in each DB record

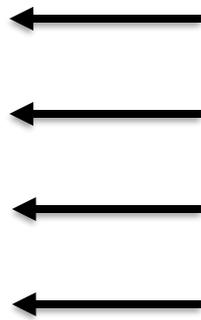- Existing FPGA implementations are direct implementations of software

Our Approach

Filter Design

Index

Record	Start
0	2
0	2
0	4
0	4

**Disk
(10X size of DB)**

Table of Contents

HSP / Length	Start	Length
ACBC-8	0	1
ACCV-9	1	1
CBBC-6	2	1
CBCV-7	3	1

**DRAM
(160 MB)**

Suffix Table

	AC	BC	CB	CV
AC	0	1	0	1
BC	0	0	0	0
CB	0	1	0	1
CV	0	0	0	0

**On-chip
(1 Mb)**

UNIVERSITY OF
SOUTH CAROLINA

Filter Design

- Synthesized with Vivado HLS
- Multiple PEs →

- When found, schedule access to DRAM

- All PEs together synthesized as single pipeline
  - Performs at WCET
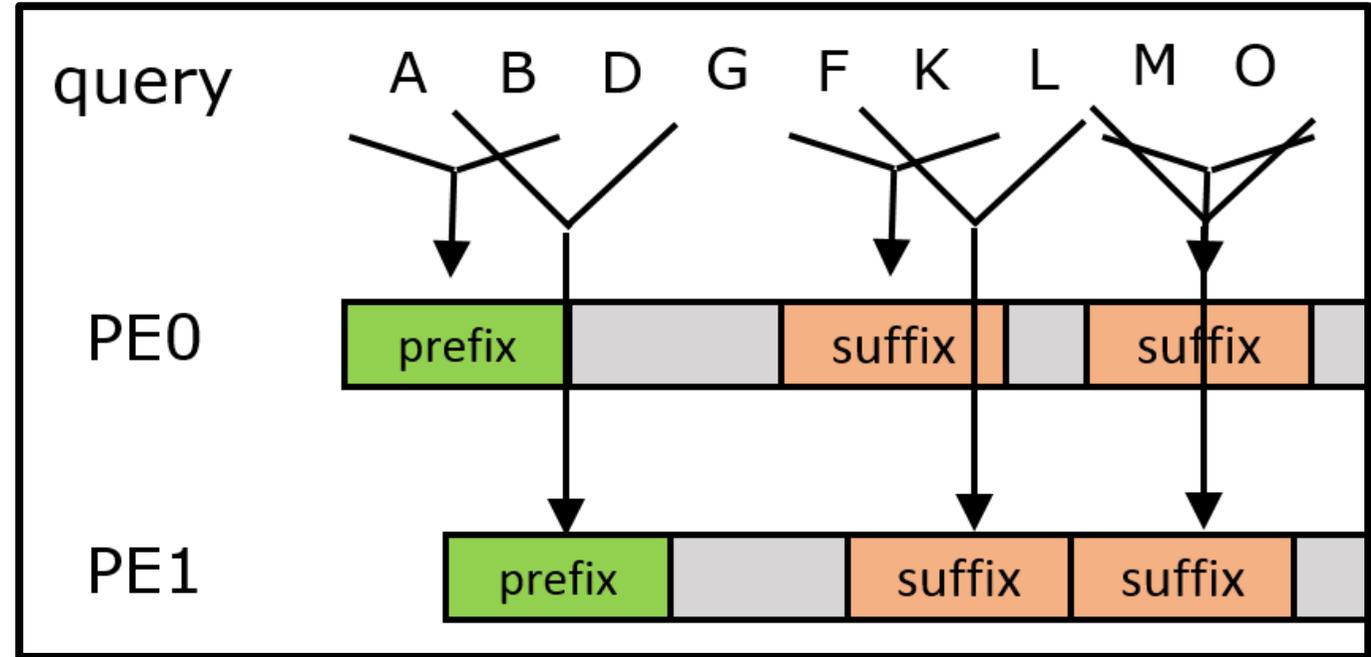
- Detects HSP patterns within fixed windows

Results

FPGA Performance

# PEs	II	Pipeline Depth	Throughput (Mchars/s)
12	12	40	9.2
16	16	44	8.4
20	20	48	7.7
24	24	52	7.1
28	28	56	6.6
32	32	60	6.2
36	36	64	5.8
40	40	68	5.4
44	44	72	5.1

CPU Baseline Performance

Threads	upper bound Thrp't. (Mc/s)	lower bound Thrp't. (Mc/s)
1	8.7	2.4
2	11.5	3.9
3	8.3	4.6
4	7.0	4.6
5	6.1	3.7
6	5.0	3.6
7	4.7	3.4
8	4.2	3.1

Faster for high-hit rate queries despite having less than half DRAM b/w

UNIVERSITY OF
SOUTH CAROLINA

Thank you!

See our poster in the next poster session