GNATY: A tools library for faster variant calling and coverage analysis

B. Wolf, P. Kuonen, *University of Applied Sciences Western Switzerland*
T. Dandekar, *University of Würzburg, Germany*

Introduction

In consequence of the speed increases in next generation sequencing over recent years, the proportion of time spent in sequence analysis compared to sequencing has increasingly shifted towards sequence analysis.
While certain analysis steps, such as sequence alignment, were able to benefit from various speed increases, others, equally important steps, like variant calling or coverage analysis, did not receive the same improvements.
Analysing NGS data remains a complicated and time consuming process, requiring a substantial amount of computing power. Most current approaches to address the increasing data quantity rely on the usage of more powerful hardware or offload calculations to the cloud.
In this poster we show that by using modern software development techniques such as stream processing, those additional analysis steps can be sped up without changing the analysis results. We demonstrate this by implementing a variant caller based on the Varscan 2 model, as well as a coverage analysis tool based on the BEDtools 2 model.
GNATY, which is based on the code used in GensearchNGS is a free and available at gnaty.phenosystems.com.

Methods

The variant caller as well as the coverage analysis tool have been implemented in Java, using HTSjdk to access the alignment files. The code has been implemented using a stream based approach, using various independent modules that abstract the different analysis steps. Not only did this allow to share a lot of code between both tools (as seen in Figure 1), but by running the different modules in independent threads, it separated I/O tasks from calculation heavy tasks, thus optimizing the resource usages.

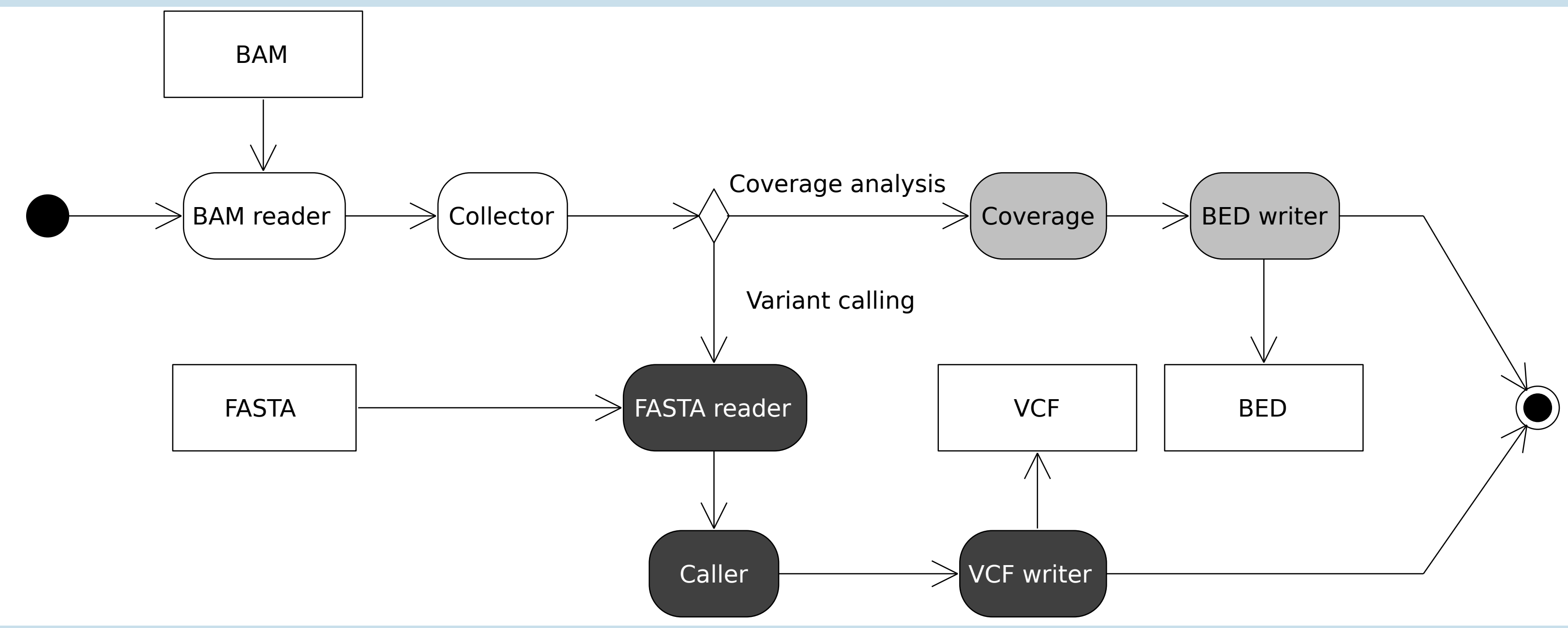

Figure 1: UML diagram of the shared tools architecure

The variant calling reproduces most of the options found in Varscan 2, with one notable exception which is the multi sample variant calling. This feature will be added at a later stage. The implementation focuses not only on speed, but also on reproducing the same results as Varscan 2 and BEDtools. Varscan 2.3.7 and BEDtools 2.21.0 where used as the target versions. For Varscan 2 a special compatibility mode has been implemented to reproduce some of its behaviour.

Results

Both tools have been tested with 2 standardized Datasets, one single ended 150x and one paired end 123x. For both the variant calling and coverage analysis, GNATY produced the same results as Varscan2 and BEDtools 2. Figure 2 shows the runtimes for both benchmarks, averaged over 3 runs. We see an overall speed increase of 18 times when calling variants and 52% when doing coverage analysis.

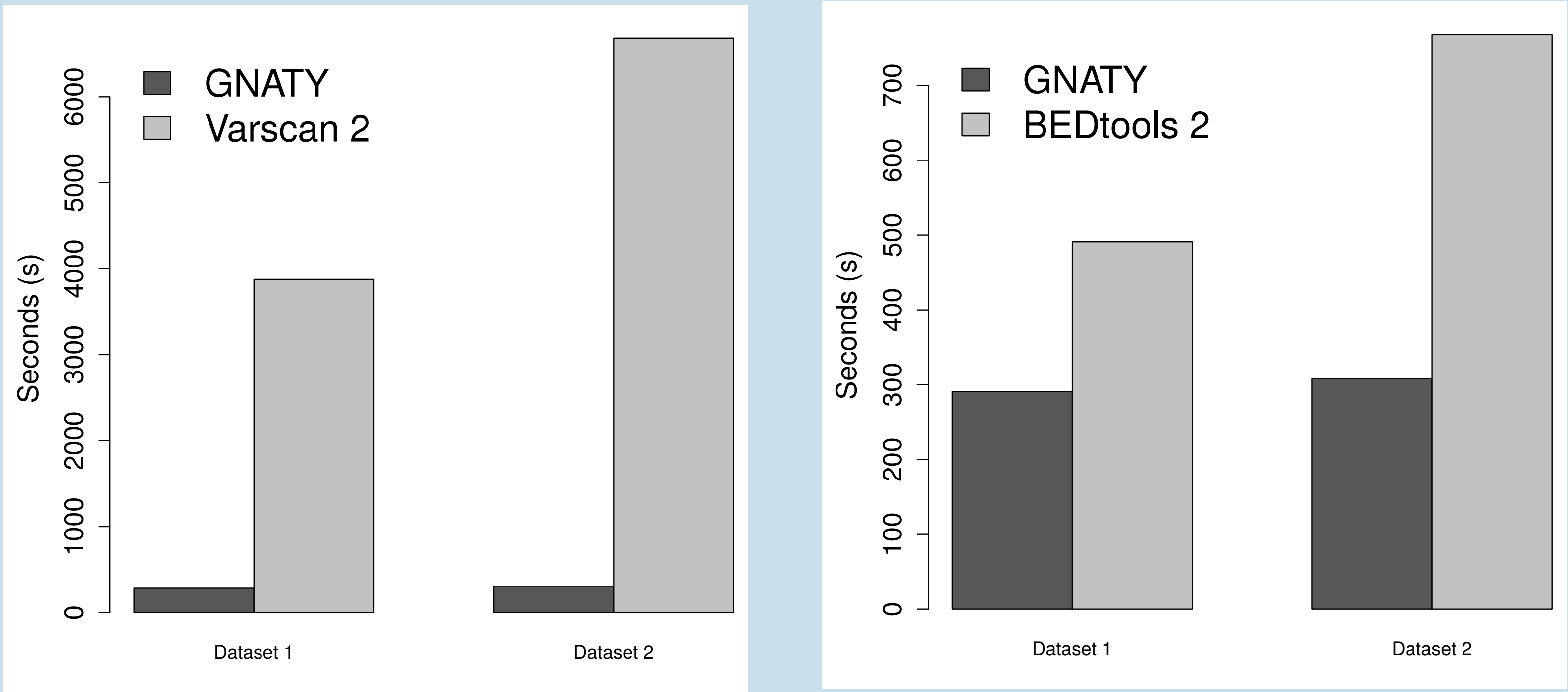

Figure 2: Benchmark times for variant calling(l) and coverage analysis (r)

Considering the time required to align both datasets with BWA of 164 minutes, reducing the post alignment analysis time from 196 minutes to 20 minutes is indeed a big change. This makes the analysis not only more time efficient, but also allows research to experiment with different settings for the data analysis, without having to wait a long time for the results. Pushing the performance further would require a change in hardware, as GNATY is mainly limited by the hard disk speed.

Conclusion

We were able to show that decoupling I/O operations from the calculation heavy parts allows to drastically decrease the time needed for the analysis, without changing the results.
The speed increases in GNATY range from 52% for coverage analysis compared to BEDtools 2, up to a 18 fold speed increase for variant calling compared to Varscan 2.
The speed increase of GNATY compared to Varscan 2 in variant calling transforms the critical step of variant calling from a time intensive processing step to one that no longer takes a critical amount of time in the complete workflow.
This shows that there is still a lot of potential for speed increases in NGS analysis pipelines, without having to invest in faster hardware.
Future work on GNATY includes, the introduction of a probabilistic variant calling method, similar to the one used in samtools and GATK, variant calling on multiple samples simultaneously and additional performance improvements.
Having demonstrated that there is still a lot of optimization potential in default NGS data analysis tools, we will also investigate the possibility of optimizing other tools.
In the context of this paper, the conversion of unsorted SAM files to sorted BAM files appears to be an interesting target for future optimizations.

Contact: Beat wolf, beat.wolf@hefr.ch

Website GensearchNGS:
http://gnaty.phenosystems.com/