Supplementary material

NGSANE is a framework for advanced production informatics of Next Generation Sequencing libraries. Fig. 1 visualises NGSANE's functionality and structure and Tab. 1 contains a list of software with similar functionality as NGSANE. The following sections showcase the three steps to run NGSANE.

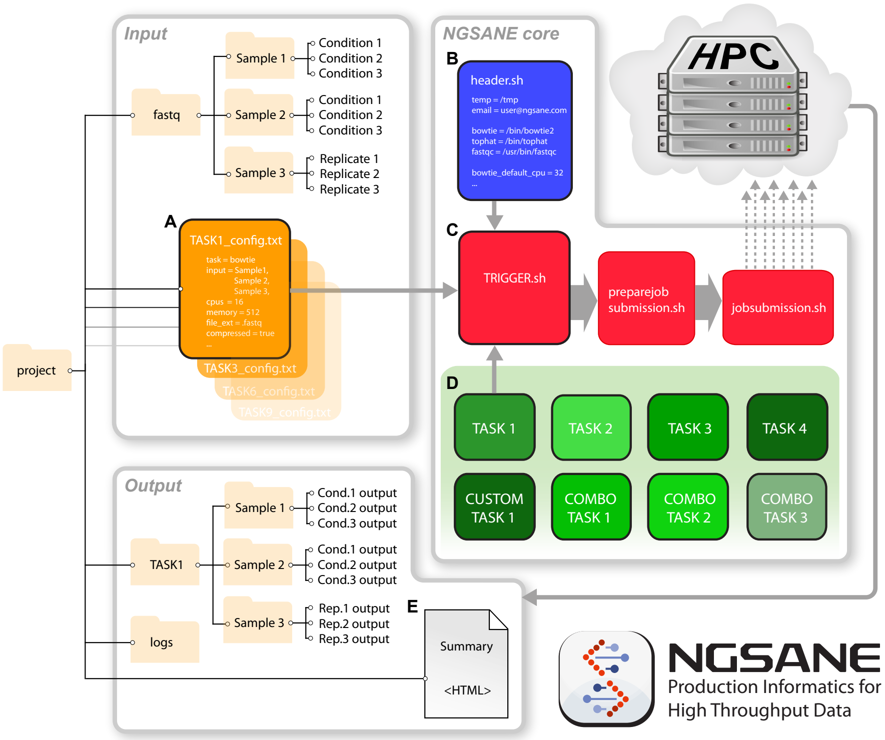


Figure 1: **Overview over NGSANE's functionality and structure**. Each project (Input) has a project specific config file **(A)** holding the necessary customisations for the planned analysis tasks. Note, each project can have multiple config files for each analysis task. Distinct from the project is the NGSANE core, which contains the pan-project configuration file called header.sh **(B)**. This file contains general system variables, platform-specific parameters, and paths to the various software binaries installed on a system. It should be configured once upon initial installation and updated as needed, for example when new software versions are installed. Also in the core is the trigger.sh script **(C)**, which is the main executable in NGSANE. It processes the variables and tasks specified in the configuration files, ensuring that all dependencies are met and invoking the core job submission protocols. In addition, it enables the user to selectively launch a test or 'dry' run, a full high performance computing run, or generate a summary report once the tasks have completed. **(D)** The mod files contain the generic analytic pipelines that are to be executed on the HPC cluster. Each mod corresponds to a specific analysis, a single task, or a series thereof. They include checkpoints to recover previous failed executions, as well as comprehensive logging of each step. Advanced users can create customised mods and include them in the framework. After execution, a concise summary of the results and a project card **(E)** can be generated. This usually includes general statistics of the results, including graphs, potential errors, and an itemised log of the checkpoints for each task.

1.1 Setup environment using toy data

1. create fastq/Run directory

```
-bash-4.1$ mkdir -p fastq/Run
```

2. copy toy data to the fastq directory

```
-bash-4.1$ wget http://www.hpsc.csiro.au/users/bau04c/datahome/Sandbox/NGSANEDEMO/fastq/Run/RNA3kChr16.tar.gz
```

3. uncompress toy example (untar)

```
-bash-4.1$ tar -xvzf RNA3kChr16.tar.gz
```

4. move fastq files to fastq/Run directory

```
-bash-4.1$ mv RNA3kChr16* fastq/Run
```

5. list the content of the fastq folder; it should contain one read pair but the user can deposit multiple libraries

```
-bash-4.1$ ls fastq/Run
RNA3kChr16_read1.fastq  RNA3kChr16_read2.fastq
```

6. copy the configuration file for NGSANE

```
-bash-4.1$ wget http://www.hpsc.csiro.au/users/bau04c/datahome/Sandbox/NGSANEDEMO/config.txt .
```

7. display the configuration file; it is currently set up to perform mapping with Bowtie2 [13] (set RUNMAPPINGBWA=”1” activates mapping with BWA [15])

```
-bash-4.1$ cat config.txt
# author: Denis C. Bauer
# date: April 2013

#********************
# Tasks
#********************
RUNMAPPINGBOWTIE2="1" # mapping with bowtie2 set to 1 to run
RUNMAPPINGBWA="" # mapping with bwa

#********************
# Paths
#********************
SOURCE=$(pwd)
declare -a DIR; DIR=( Run )
OUT=$SOURCE
QOUT=$OUT/qout

READONE="_read1"
READTWO="_read2"
FASTQ=fastq
EXPID="Library"
LIBRARY="Provider"
PLATFORM="Illumina"

FASTA=$NGSANE_REFERENCE/b37/human_g1k_v37.fasta # needs to be set to appropriate path
```

1.2 Run NGSANE on toy data

1. test configuration (dry run)

```
-bash-4.1$ trigger.sh config.txt
[NGSANE] Trigger mode: [empty] (dry run)
[NOTE] Folders: Run
[Task] bowtie2
[NOTE] setup environment
[TODO] Run/RNA3kChr16_read1.fastq
[NOTE] proceeding with job scheduling...
[NOTE] Run/RNA3kChr16
[NOTE] make Run/bowtie2/RNA3kChr16.asd.bam.dummy
[ JOB]  /apps/gi/ngsane/0.2.0//mods/bowtie2.sh
-k /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/config.txt
-f /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/fastq/Run/RNA3kChr16_read1.fastq
-o /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/Run/bowtie2 --rgsi Run
```

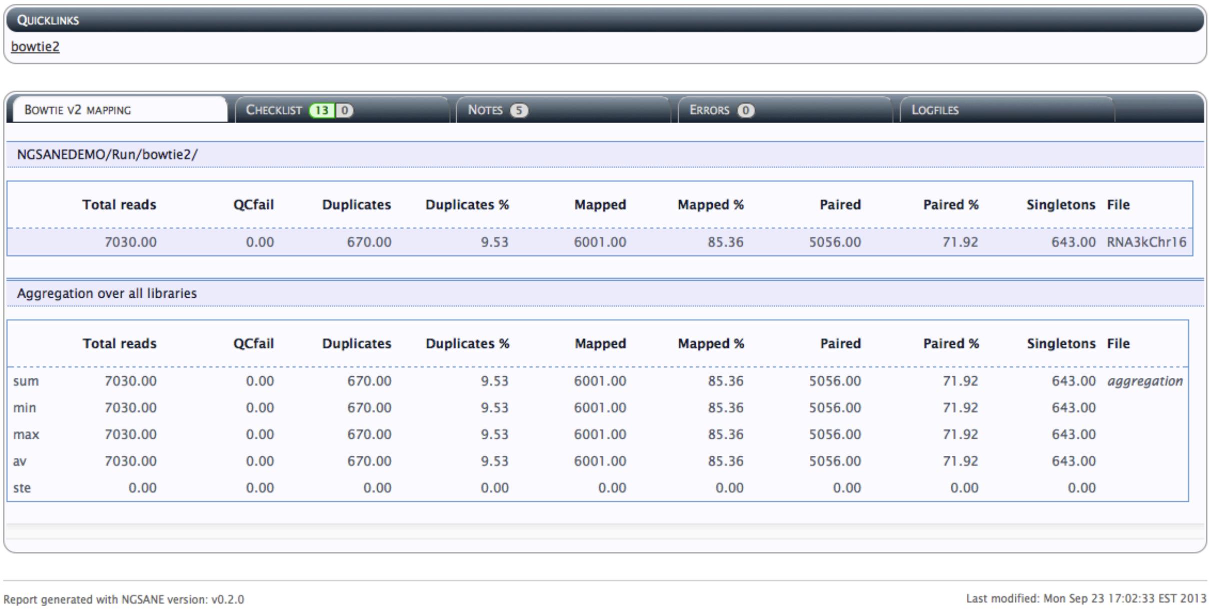
2. submit a bowtie job to the cluster (Jobnumber 2187179)

   ```
   -bash-4.1$ trigger.sh config.txt armed
   [NGSANE] Trigger mode: armed
   Double check! Then type safetyoff and hit enter to launch the job: safetyoff
   ... take cover!
   [NOTE] Folders: Run
   [Task] bowtie2
   [NOTE] setup enviroment
   [TODO] Run/RNA3kChr16_read1.fastq
   [NOTE] proceeding with job scheduling...
   [NOTE] Run/RNA3kChr16
   [NOTE] make Run/bowtie2/RNA3kChr16.asd.bam.dummy
   [ JOB]  /apps/gi/ngsane/0.2.0//mods/bowtie2.sh
   -k /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/config.txt
   -f /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/fastq/Run/RNA3kChr16_read1.fastq
   -o /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/Run/bowtie2 --rgsi Run
   Jobnumber 2187179
   ```

1.3 Aggregate results in a summary report

To have a one page overview of job success, and results we now generate the Project Card

```
-bash-4.1$ trigger.sh config.txt report
[NGSANE] Trigger mode: report
>>>>> Generate HTML report
>>>>> startdate Mon Sep 23 17:02:31 EST 2013
>>>>> hostname burnet-login
>>>>> makeSummary.sh -k /datastore/cci/bau04c/Documents/datahome/Sandbox/NGSANEDEMO/config.txt
--R          --
 R version 3.0.0 (2013-04-03) -- "Masked Marvel" Copyright (C) 2013 The R Foundation for Statistical Computing
 Platform: x86_64-unknown-linux-gnu (64-bit)
--Python     --
 Python 2.7.2
QC - bowtie2
>>>>> Generate HTML report - FINISHED
>>>>> enddate Mon Sep 23 17:02:33 EST 2013
```



NGSANEDEMO/Run/bowtie2/

Total reads	QCfail	Duplicates	Duplicates %	Mapped	Mapped %	Paired	Paired %	Singletons	File
7030.00	0.00	670.00	9.53	6001.00	85.36	5056.00	71.92	643.00	RNA3kChr16

Aggregation over all libraries

	Total reads	QCfail	Duplicates	Duplicates %	Mapped	Mapped %	Paired	Paired %	Singletons	File
sum	7030.00	0.00	670.00	9.53	6001.00	85.36	5056.00	71.92	643.00	*aggregation*
min	7030.00	0.00	670.00	9.53	6001.00	85.36	5056.00	71.92	643.00	
max	7030.00	0.00	670.00	9.53	6001.00	85.36	5056.00	71.92	643.00	
av	7030.00	0.00	670.00	9.53	6001.00	85.36	5056.00	71.92	643.00	
ste	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

Report generated with NGSANE version: v0.2.0

Last modified: Mon Sep 23 17:02:33 EST 2013

Figure 2: **Example project card**

The Project Card (Fig. 2) can be created at the very end of the execution or at any intermittent step allowing human quality control throughout the stages of a project. The "Notes" and "Error" tabs specifically highlight, which (if any) subset of files contain error and the "Logfile" tab facilitates easy access to the specific log files to identify the source of the problem. Once the faulty files are identified and the error is removed NGSANE allows the automated resubmission of the file-subset starting from the point of error.

More comprehensive reports that include summary graphics area available from:
`http://www.hpsc.csiro.au/users/bau04c/datahome/Sandbox/smokebox_ngsane/smokebox/result/`

Table 1: Available academic or commercial software for NGS data analysis

Name	Date	comercial	Domain specific	language	subset files	recovery	pipelining	paralleli-zation	hpc	hadoop	automatic summary	URL
NGSANE	2013	academic	no	BASH	yes	yes	yes	yes	yes	not yet	yes	https://github.com/BauerLab/ngsane
Nestly [18]	2013	academic	no	python	yes	no	no	no	no	no	yes	http://github.com/fhcrc/nestly
RUbioSeq [4]	2013	academic	bisulfite and exome variants	perl			yes	yes	SGE	no	no	http://rubioseq.sourceforge.net/.
Bpipe [22]	2012	academic	no	Groovy		yes	yes	yes	yes	no	yes	http://bpipe.org
GeneProf [10]	2012	academic	no	Webserver								http://www.geneprof.org/GeneProf/
GenomicTools [23]	2012	academic	no	C++	yes	no	yes	yes	no	no	no	http://code.google.com/p/ibm-cbc-genomic-tools
PGAP [24]	2012	academic	pan-genomic analysis	Perl								http://pgap.sourceforge.net/
Snakemake [12]	2012	academic	no	python	yes		yes		not yet			https://code.google.com/p/snakemake/
NARWHAL [6]	2012	academic	de-multiplexing	python, BASH, c			yes	yes	no	no	yes	https://trac.nbic.nl/narwhal/
SeqWare [19]	2010	academic		HBase based	yes	yes	yes	yes	yes	yes	yes	http://seqware.github.io
CLoVR [5]	2011	academic	yes	Webserver				yes	no	yes		http://clovr.org/
Conveyor [17]	2011	academic	no	.NET				yes	no			http://conveyor.cebitec.uni-bielefeld.de
Knime4Bio [16]	2011	academic	yes	Webserver								http://code.google.com/p/knime4bio/
PaPy [7]	2011	academic	no	python			yes	yes				http://muralab.org/PaPy/
Pyicos [3]	2011	academic	no	python								http://regulatorygenomics.upf.edu/pyicos
NGS_backbone [2]	2011	academic	no				yes	yes				http://bioinf.comav.upv.es/ngs_backbone/
Galaxy [8]	2010	academic	no	Webserver	yes	yes	yes	no	no		no	
Ruffus [9]	2010	academic	no	python	yes	yes						http://www.ruffus.org.uk/
CrossBow [14]	2009	academic	SNP calling	C++						yes		http://bowtie-bio.sourceforge.net/crossbow/index.shtml
Mobyle [1]	2009	academic	no	Webserver	yes		no					
Ipython [21]	2007	academic	no	python	yes	yes	yes	yes	yes		no	http://ipython.org/
Taverna [20]	2004	academic	no	Webserver	yes	yes	yes	no	no			http://www.taverna.org.uk/
Biopipe [11]	2003	academic	no	Webserver								
GATK Queue		academic	GATK commands	JAVA								http://gatkforums.broadinstitute.org/discussion/1306/overview-of-queue
Gene Pattern		academic		Webserver								http://www.broadinstitute.org/cancer/software/genepattern
ISGA		academic	prokaryotic annotation and prokaryotic assembly	Webserver								http://gmod.org/wiki/ISGA
Kepler		academic		Webserver	yes		no		yes			
Pegasus		academic		Webserver	yes				yes			
GeneSpring		Agilent										http://www.genomics.agilent.com
Avadis NGS		Avadis NGS										http://www.avadis-ngs.com/
Genomics Workbench		CLC Bio										http://www.clcbio.com/
DNASTAR		DNASTAR										http://www.dnastar.com
CASAVA		illumina										http://www.illumina.com/software/genome_analyzer_software.ilmn
NextGENe		NextGENe										http://www.softgenetics.com/NextGENe.html
Partek		Partek										http://www.partek.com/?q=ngs
geospiza		Perkin-Elmer										http://www.geospiza.com/&

Table 2: **Software similar to** Ngsane**.** Tools are listed in no particular order and the list may not be comprehensive. See https://github.com/BauerLab/ngsane/wiki/Similar-Projects for an up-to-date list.

References

[1] (2009). Mobyle: a new full web bioinformatics framework. *Bioinformatics*, **25**(22), 3005–3011.

[2] (2011). ngs_backbone: a pipeline for read cleaning, mapping and snp calling using next generation sequence. *BMC Genomics*, **12**, 285.

[3] (2011). Pyicos: a versatile toolkit for the analysis of high-throughput sequencing data. *Bioinformatics*, **27**(24), 3333–3340.

[4] (2013). Rubioseq: a suite of parallelized pipelines to automate exome variation and bisulfite-seq analyses. *Bioinformatics*, **29**(13), 1687–1689.

[5] Angiuoli, S. V., Matalka, M., Gussman, A., Galens, K., Vangala, M., Riley, D. R., Arze, C., White, J. R., White, O., and Fricke, W. F. (2011). Clovr: a virtual machine for automated and portable sequence analysis from the desktop using cloud computing. *BMC Bioinformatics*, **12**, 356.

[6] Brouwer, R. W. W., van den Hout, M. C. G. N., Grosveld, F. G., and van Ijcken, W. F. J. (2012). Narwhal, a primary analysis pipeline for ngs data. *Bioinformatics*, **28**(2), 284–285.

[7] Cieslik, M. and Mura, C. (2011). A lightweight, flow-based toolkit for parallel and distributed bioinformatics pipelines. *BMC Bioinformatics*, **12**, 61.

[8] Goecks, J., Nekrutenko, A., Taylor, J., and Galaxy Team (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biol*, **11**(8), R86.

[9] Goodstadt, L. (2010). Ruffus: a lightweight python library for computational pipelines. *Bioinformatics*, **26**(21), 2778–2779.

[10] Halbritter, F., Vaidya, H. J., and Tomlinson, S. R. (2012). Geneprof: analysis of high-throughput sequencing experiments. *Nat Methods*, **9**(1), 7–8.

[11] Hoon, S., Ratnapu, K. K., Chia, J.-M., Kumarasamy, B., Juguang, X., Clamp, M., Stabenau, A., Potter, S., Clarke, L., and Stupka, E. (2003). Biopipe: a flexible framework for protocol-based bioinformatics analysis. *Genome Res*, **13**(8), 1904–1915.

[12] Köster, J. and Rahmann, S. (2012). Snakemake – a scalable bioinformatics workflow engine. *Bioinformatics*, **28**(19), 2520–2522.

[13] Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with bowtie 2. *Nat Methods*, **9**(4), 357–359.

[14] Langmead, B., Schatz, M. C., Lin, J., Pop, M., and Salzberg, S. L. (2009). Searching for snps with cloud computing. *Genome Biol*, **10**(11), R134.

[15] Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics*, **25**(14), 1754–1760.

[16] Lindenbaum, P., Le Scouarnec, S., Portero, V., and Redon, R. (2011). Knime4bio: a set of custom nodes for the interpretation of next-generation sequencing data with knime. *Bioinformatics*, **27**(22), 3200–3201.

[17] Linke, B., Giegerich, R., and Goesmann, A. (2011). Conveyor: a workflow engine for bioinformatic analyses. *Bioinformatics*, **27**(7), 903–911.

[18] McCoy, C. O., Gallagher, A., Hoffman, N. G., and Matsen, F. A. (2013). Nestly – a framework for running software with nested parameter choices and aggregating results. *Bioinformatics*, **29**(3), 387–388.

[19] O'Connor, B. D., Merriman, B., and Nelson, S. F. (2010). Seqware query engine: storing and searching sequence data in the cloud. *BMC Bioinformatics*, **11 Suppl 12**, S2.

[20] Oinn, T., Addis, M., Ferris, J., Marvin, D., Senger, M., Greenwood, M., Carver, T., Glover, K., Pocock, M. R., Wipat, A., and Li, P. (2004). Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, **20**(17), 3045–3054.

[21] Pérez, F. and Granger, B. E. (2007). IPython: a System for Interactive Scientific Computing. *Comput. Sci. Eng.*, **9**(3), 21–29.

[22] Sadedin, S. P., Pope, B., and Oshlack, A. (2012). Bpipe: a tool for running and managing bioinformatics pipelines. *Bioinformatics*, **28**(11), 1525–1526.

[23] Tsirigos, A., Haiminen, N., Bilal, E., and Utro, F. (2012). Genomictools: a computational platform for developing high-throughput analytics in genomics. *Bioinformatics*, **28**(2), 282–283.

[24] Zhao, Y., Wu, J., Yang, J., Sun, S., Xiao, J., and Yu, J. (2012). Pgap: pan-genomes analysis pipeline. *Bioinformatics*, **28**(3), 416–418.