BiG Grid
the dutch e-science grid

Fork me on GitHub

Whole genome alignment and imputation pipelines on MOLGENIS compute pilot framework.

George Byelas, Pieter Neerincx, Martijn Dijkstra, Freerk van Dijk, Jan Bot, Mathijs Kattenberg,Tom Visser, Irene Nooren, eBioGrid, BBMRI-NL, NBIC,

Morris Swertz

umcg BBMRI • NL nbioassist Biobanking TarGet molgenis

Outline

- Introduction
- Results & How-to-use
- Concluding remarks

BiG Grid
the dutch e-science grid

Introduction

umcg

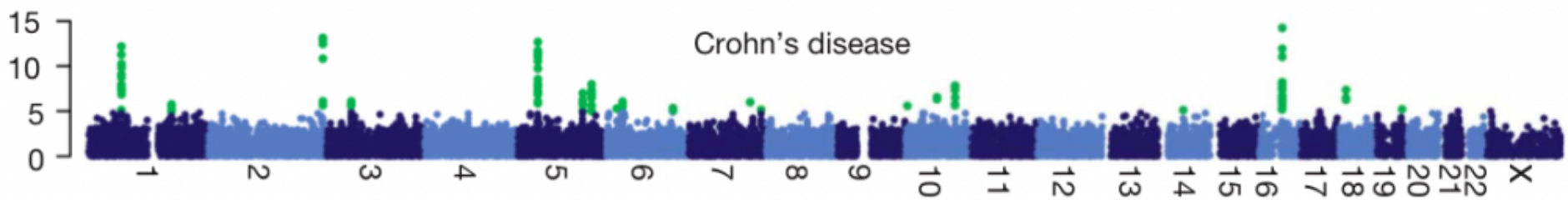
molgenis

Motivation: High-throughput biology

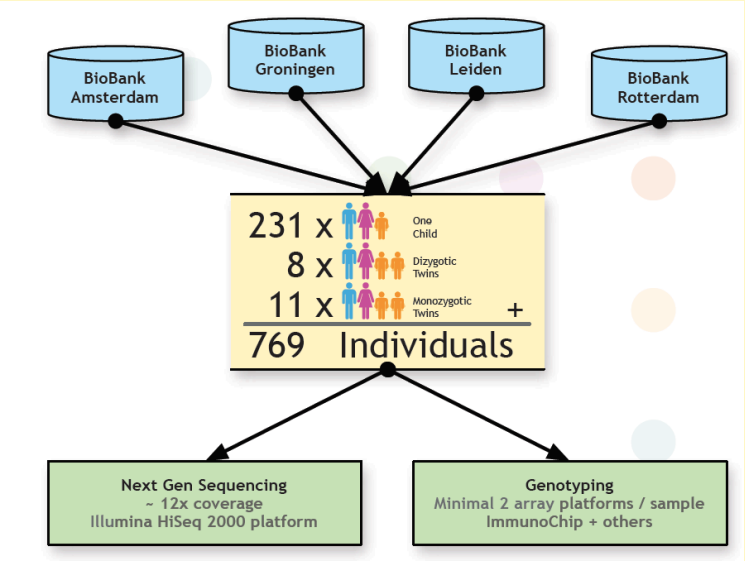
umcg

molgenis

Biobanking & Cohorts (BBMRI-NL)

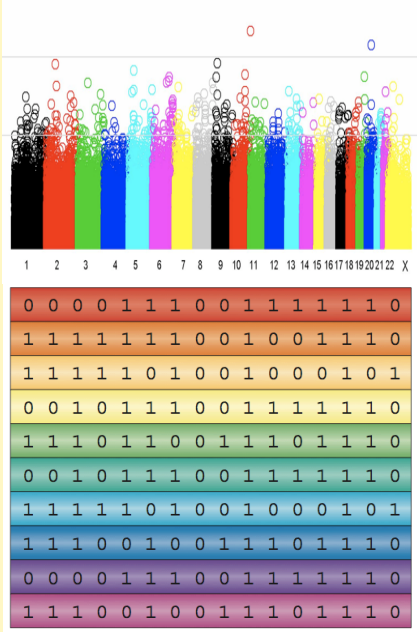


Whole Genome sequencing



~769 whole exome NGS data
1000s of exome samples 'in-house'

Imputation existing GWAS

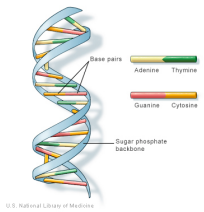


~100,000 Dutch samples with GWAS data

umcg

molgenis

Whole exome/genome sequencing



HiSeq

Raw data

10s-100s samples

analysis pipeline

20 – 200 days

Result data

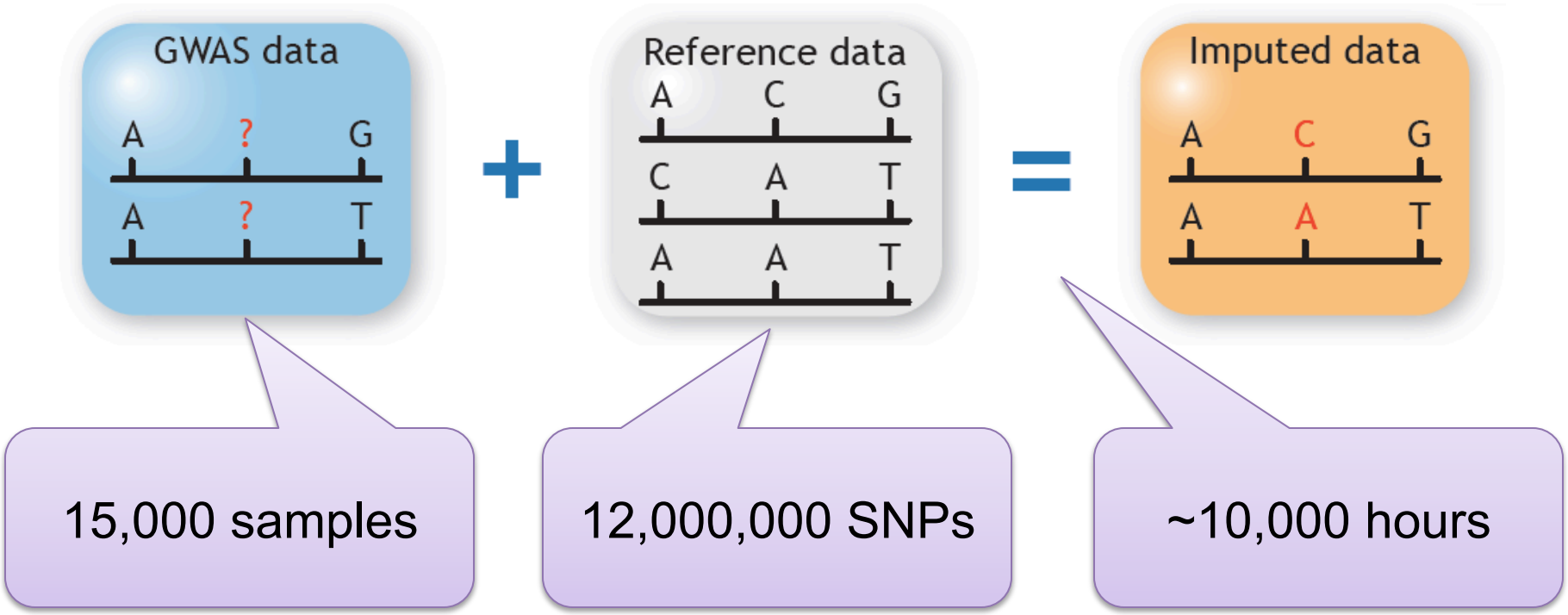
Per Project:

1. Raw sequence reads
2. Aligned reads
3. QC-report
4. SNP lists

umcg

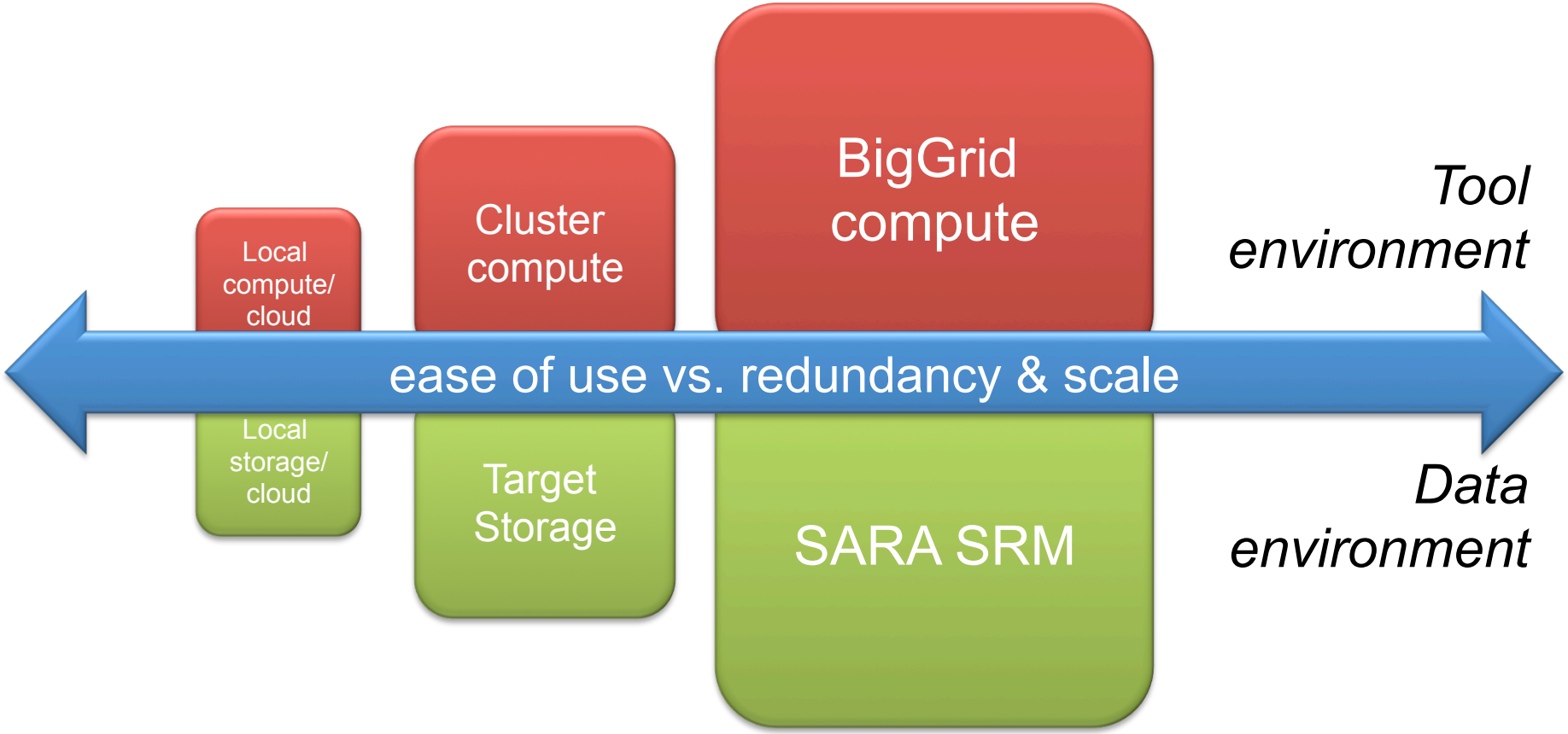
molgenis

Imputation & Genome Wide Association Studies



In NL >100,000 Dutch samples with GWAS data

Challenge: easily move pipeline between backends



Challenge: many, changing, analyses... hard to maintain???

This changes per backend

```
#!/bin/bash
#PBS -q test
#PBS -l nodes=1:ppn=4
#PBS -l walltime=08:00:00
#PBS -l mem=6gb
#PBS -e $GCC/test_compute/projects/batch4/intermediate/test1/err/err_test1_BwaElement1A102a_FC81D90ABXX_L7.err
#PBS -o $GCC/test_compute/projects/batch4/intermediate/test1/out/out_test1_BwaElement1A102a_FC81D90ABXX_L7.out

mkdir -p $GCC/test_compute/projects/batch4/intermediate/test1/err
mkdir -p $GCC/test_compute/projects/batch4/intermediate/test1/out
printf "test1_BwaElement1A102a_FC81D90ABXX_L7_started " >>$GCC/test_compute/projects/batch4/intermediate/test1/log_test1.txt
date "+DATE: %m/%d/%y%tTIME: %H:%M:%S" >>$GCC/test_compute/projects/batch4/intermediate/test1/log_test1.txt
date "+start time: %m/%d/%y%t %H:%M:%S" >>$GCC/test_compute/projects/batch4/intermediate/test1/
test1_BwaElement1A102a_FC81D90ABXX_L7.txt
echo running on node: `hostname` >>$GCC/test_compute/projects/batch4/intermediate/test1/
test1_BwaElement1A102a_FC81D90ABXX_L7.txt
```

This changes per pipeline

```
/target/gpfs2/gcc/tools//bwa-0.5.8c_patched/bwa aln \
/target/gpfs2/gcc/resources/hg19/indices/human_g1k_v37.fa \
$GCC/test_compute/projects/batch4/rawdata/110121_I288_FC81D90ABXX_L7_HUMrutRGADIAAPE_1.fq.gz \
-t 4 \
-f $GCC/test_compute/projects/batch4/intermediate/A102a_110121_I288_FC81D90ABXX_L7_HUMrutRGADIAAPE_1.fq.gz.sai
```

This changes per backend

```
printf "test1_BwaElement1A102a_FC81D90ABXX_L7_finished " >>$GCC/test_compute/projects/batch4/intermediate/test1/log_test1.txt
date "+finish time: %m/%d/%y%t %H:%M:%S" >>$GCC/test_compute/projects/batch4/intermediate/test1/
test1_BwaElement1A102a_FC81D90ABXX_L7.txt
date "+DATE: %m/%d/%y%tTIME: %H:%M:%S" >>$GCC/test_compute/projects/batch4/intermediate/test1/log_test1.txt
```

umcg

molgenis

Mission: e-science infrastructure for biobank bioinformatics

Dramatically reduce the expertise and time

To design, execute and change large workflows

Using standard scripts bioinformaticans know and love

Across BigGrid sites, PBS/SGE clusters, and local servers/clouds

Results

1. Generic ‘compute’ framework + operating procedures
   - Design: workflows, protocols, parameters
   - Run: worksheets, command-line submit, pilot jobs database
   - Deploy: harmonized tool and file management

2. Imputation pipelines
   - Preparing reference data (once) — 4 steps, 20min
   - QC and chunk the study data — 2 steps, 20min/chr
   - Phase the study data per chunk — 2 steps, 6h/500 sample/chr
   - Impute the study data and merge — 2 steps, 6h/500 sample/chr

3. Whole exome/genome sequencing pipelines
   - Sequence alignment, realignment — 17 steps, 5-6 days/lane/barcode
   - Variant calling & QC — 13 steps, 2 days/sample

BiG Grid
the dutch e-science grid

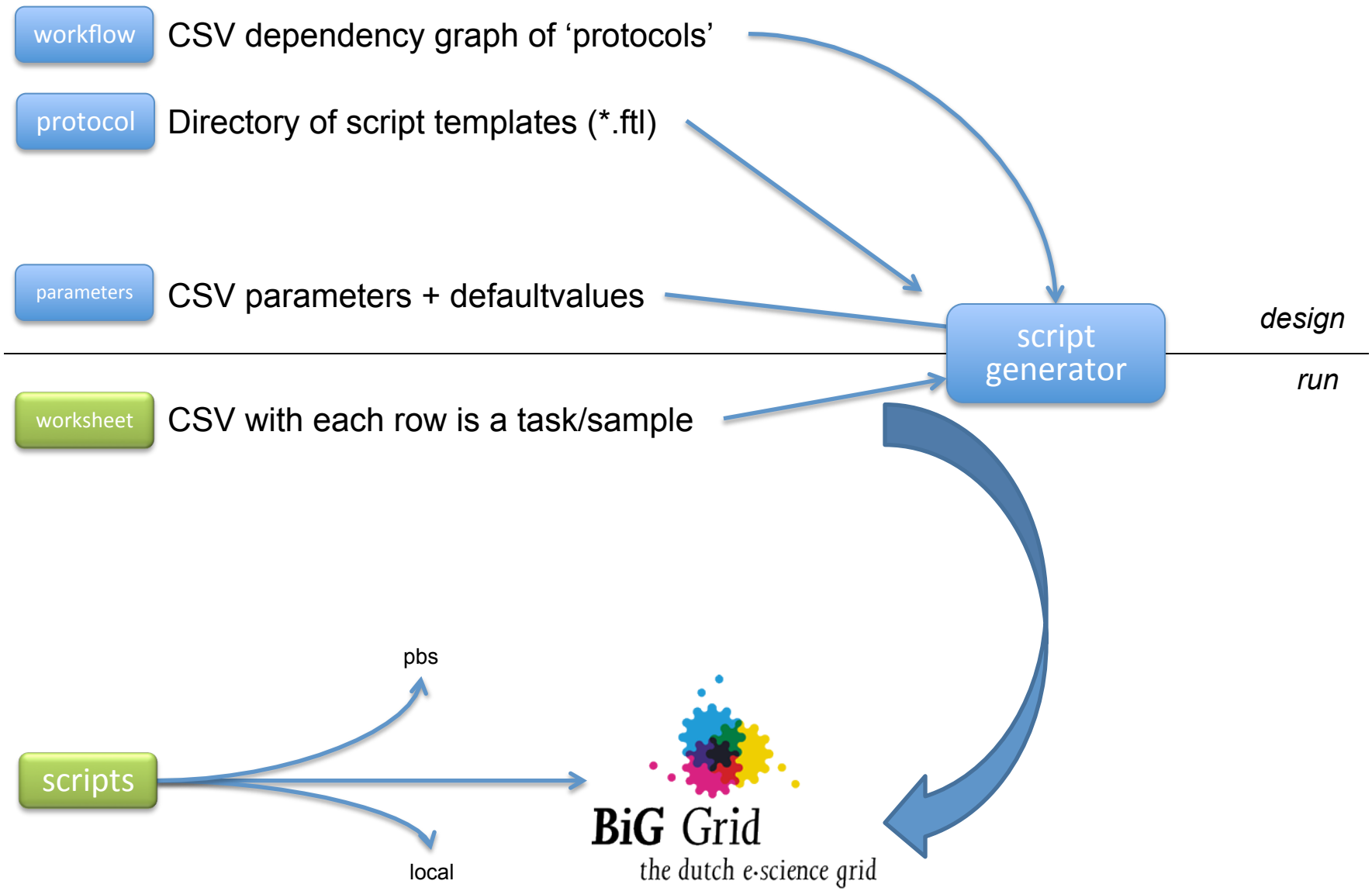
How to design

Once per pipeline

umcg

molgenis

Generic ‘compute’ framework



umcg

molgenis

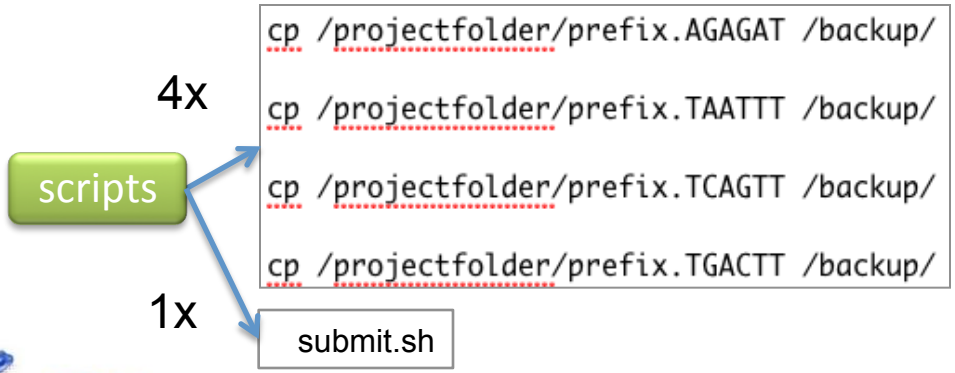
Example

Name	defaultValue
file	/projectfolder/prefix

design

run

project	contact	barcode
Microcephalie	Birgit Sikkema	AGAGAT
Microcephalie	Birgit Sikkema	TAATTT
Microcephalie	Birgit Sikkema	TCAGTT
Microcephalie	Birgit Sikkema	TGACTT

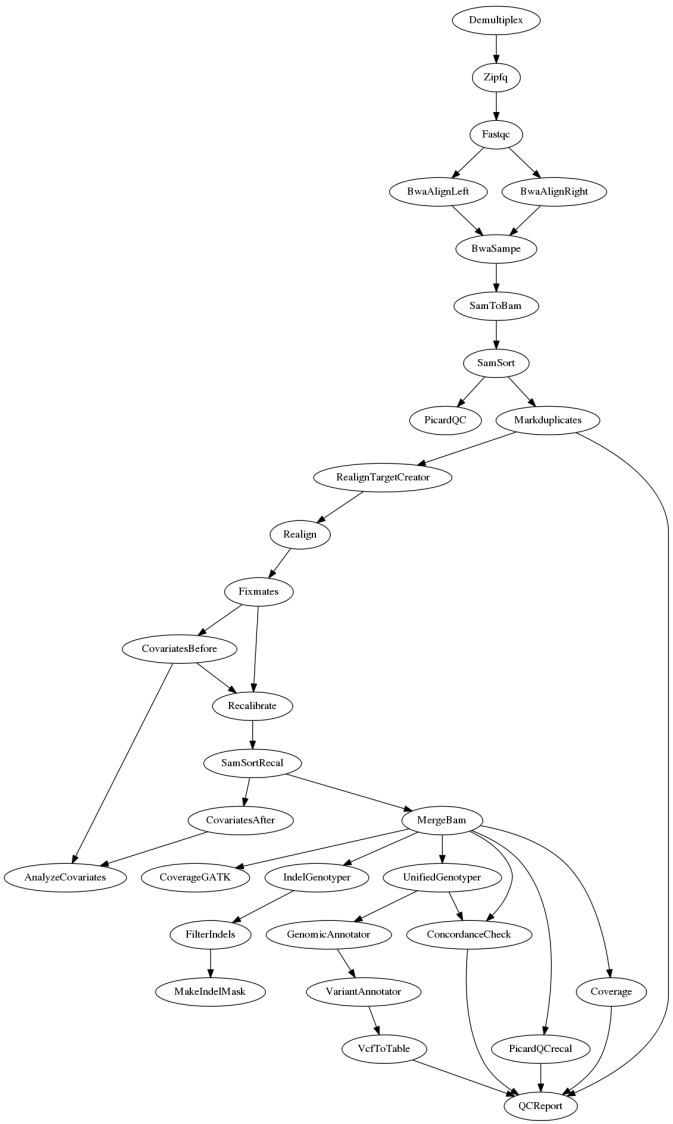


umcg

molgenis

Workflow = dependency graph

1	name	protocol_name	PreviousSteps_name
2	Demultiplex	Demultiplex	
3	Zipfq	Zipfq	Demultiplex
4	Fastqc	Fastqc	Zipfq
5	BwaAlignLeft	BwaAlignLeft	Fastqc
6	BwaAlignRight	BwaAlignRight	Fastqc
7	BwaSampe	BwaSampe	BwaAlignLeft, BwaAlignRight
8	SamToBam	SamToBam	BwaSampe
9	SamSort	SamSort	SamToBam
10	PicardQC	PicardQC	SamSort
11	Markduplicates	Markduplicates	SamSort
12	RealignTargetCreator	RealignTargetCreator	Markduplicates
13	Realign	Realign	RealignTargetCreator
14	Fixmates	Fixmates	Realign
15	CovariatesBefore	CovariatesBefore	Fixmates
16	Recalibrate	Recalibrate	Fixmates, CovariatesBefore
17	SamSortRecal	SamSortRecal	Recalibrate
18	CovariatesAfter	CovariatesAfter	SamSortRecal
19	AnalyzeCovariates	AnalyzeCovariates	CovariatesBefore, CovariatesAfter
20	PicardQCrecal	PicardQCrecal	SamSortRecal
21	Coverage	Coverage	SamSortRecal
22	IndelGenotyper	IndelGenotyper	SamSortRecal
23	FilterIndels	FilterIndels	IndelGenotyper
24	UnifiedGenotyper	UnifiedGenotyper	SamSortRecal
25	MakeIndelMask	MakeIndelMask	FilterIndels
26	VariantAnnotator	VariantAnnotator	SamSortRecal, UnifiedGenotyper
27	GenomicAnnotator	GenomicAnnotator	VariantAnnotator
28	VariantFiltration	VariantFiltration	GenomicAnnotator, MakeIndelMask
29	VcfToTable	VcfToTable	VariantFiltration
30	ConcordanceCheck	ConcordanceCheck	SamSortRecal, UnifiedGenotyper
31	QCReport	QCReport	PicardQC, ConcordanceCheck



umcg

molgenis

Protocol = freemarker templates

```
#MOLGENIS walltime=15:00:00 nodes=1 cores=4 mem=6
#FOREACH leftbarcodefqgz

module load bwa/${bwaVersion}

getFile ${indexfile}
getFile ${leftbarcodefqgz}

bwa aln \
${indexfile} \
${leftbarcodefqgz} \
-t ${bwaaligncores} \
-f ${leftbwaout} \

putFile ${leftbwaout}
```

resource management

tool management

data management

template of the actual analysis

data management

Parameters: csv file, can have freemarker too ☺

... 300 parameters

	A	B	C	D	E
1	Name	defaultValue	description	dataType	hasOne_nam
2	clusterQueue	gaf			
3	mem	4	Memory in GB		
4	walltime	23:59:00			
5	cores	1			
6	scheduler	GRID			
7	defaultInterpreter	#!/bin/bash			
8	library	${fileprefix}			
9	jobname	jobname		string	
10	root	$WORKDIR	the root to y	string	
11	bashrc	${root}/gcc.bashrc			
12	workflowHeader	NGSHeader.ftl			
13	tooldir	${root}/tools	root dir for t	string	
14	gafhome	/target/gpfs2/gaf	gaf home dir	string	
15	gaftools	${gafhome}/tools		string	
16	gafscripts	${gaftools}/scripts		string	
17	scriptdir	${tooldir}/scripts			
18	importscript	${scriptdir}/import.sh			
19	transferDataScript	${scriptdir}/transferData.sh			
20	demultiplexscript	${gafscripts}/demultiplex.R		string	
21	demultiplexWorkflowFile	${McDir}/workflows/in-house_workflow_demultiplex.csv		string	
22	workflowFile	${McDir}/workflows/in-house_workflow_realignmentAndSnpCalling.csv		string	
23	resdir	${root}/resources			
24	JAVA_HOME	/cm/shared/apps/sunjdk/jdk1.6.0_21/			
25	R_HOME	${tooldir}/R/			
26	R	${R_HOME}/bin/R			
27	rscript	${R_HOME}/bin/Rscript			
28	R_LIBS	${tooldir}/GATK-1.3-24-gc8b1c92/gsalib/			
29	intervallistdir	${resdir}/${genome}/intervals			
30	capturingKit				
31	baitintervals	${intervallistdir}/${capturingKit}_baits_${genome}_${indexfileIDtest}.interval_list			
32	targetintervals	${intervallistdir}/${capturingKit}_exons_${genome}_${indexfileIDtest}.interval_list			
33	baitsbed	${intervallistdir}/${capturingKit}_baits_${genome}_${indexfileIDtest}.bed			
34	tempdir	${root}/tmp/processing/			
35	tempprojectdir	${root}/tmp/			
36	genome	hg19			
37	indexfileID	human_g1k_v37			

nolgenis

Worksheet = parameter values (or ‘compute targets’)

A	B	C	D	E	F	G	H	I
externalSampleID	project	contact	sequencer	run	flowcell	lane	seqType	prepKit
Test_DNA	demo	researcher	SN163	457	BD0E5CACXX	4	PE	NEB

continued...

I	J	K	L	M	N
prepKit	capturingKit	arrayFile	arrayID	barcode	barcodeType
NEB	SureSelect_All_Exon_5	Alle_Samples	3	CAACCT	GAF

umcg

molgenis

Example: smart iteration using ‘foreach’

workflow

p1 ← p2

protocol

p2

```
42 #FOREACH project
43 Dear ${contact},
44 The barcodes in project ${project} are:<#list barcode as bc> ${bc} </#list>
```

parameters

Name	defaultValue	hasOne_name
file	/projectfolder/prefix	
project		contact

design

script generator

run

worksheet

project	contact	barcode
Microcephalie	Birgit Sikkema	AGAGAT
Microcephalie	Birgit Sikkema	TAATTT
Microcephalie	Birgit Sikkema	TCAGTT
Microcephalie	Birgit Sikkema	TGACTT

‘Folded’ on project and contact

project	contact	barcode
Microcephalie	Birgit Sikkema	AGAGAT, TAATTT, TCAGTT, TGACTT

1x

scripts

```
Dear Birgit Sikkema,
The barcodes in project Microcephalie are:AGAGAT TAATTT TCAGTT TGACTT
```

submit.sh

1x

umcg

molgenis

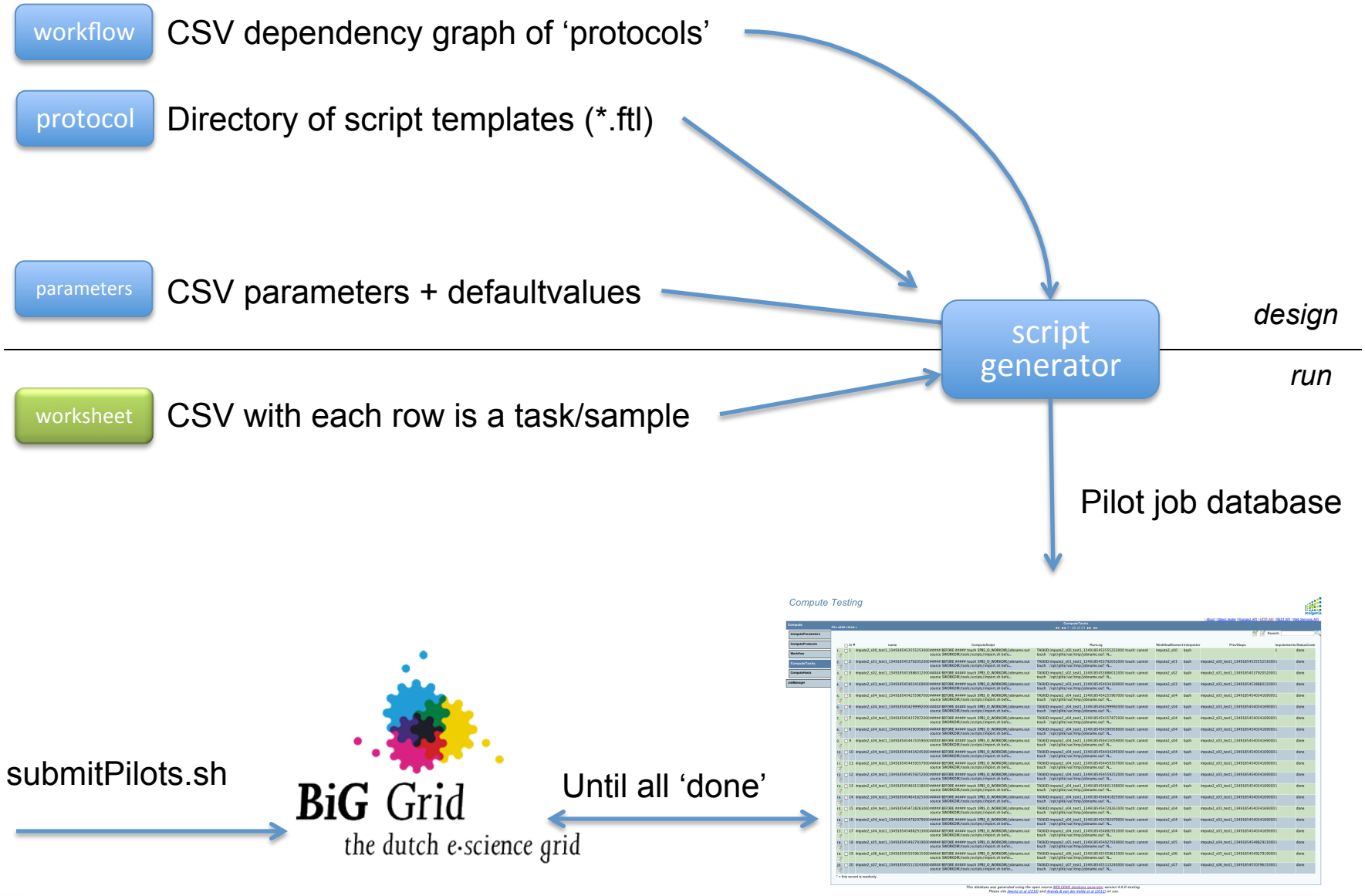
BiG Grid
the dutch e-science grid

How to run @ BigGrid

umcg

molgenis

Job pilot framework



umcg

molgenis

Generic ‘compute’ framework

- Manual: github.com/molgenis -> molgenis_apps/doc/compute

molgenis_apps / doc / compute / **01_compute_introduction.md**

mswertz 2 hours ago minor fixes to docs; added README.md

1 contributor

file | 462 lines (314 sloc) | 26.593 kb

Edit | Raw | Blame | History

% Manual Molgenis/Compute % Genomics Coordination Center % December 16, 2012

Compute framework overview

MOLGENIS compute (or *Compute*) is 'just enough' to rapidly generate analysis workflows that can run locally, on parallel compute clusters and the grid.

- Users can use build on their standard expertise in (shell) scripts
- Users can rapidly share their workflows to accross Linux servers
- Users can easily view the scripts generated for provenance and debugging.
- Users can customize *Compute* to fit their local practices

To use *Compute*, you need the following.

- `workflow.csv` : a file that describes steps to be executed in order.
- `protocols` : a directory in which each file is a script template describing a step.

Tutorial workflow: ‘invitation for a party’

1. Create worksheet

guest,	group,	organizer
Charly,	child,	Oscar
Cindy,	child,	Oscar

2. Generate your analysis run

```
sh molgenis_compute \
—workflowDir=invitationWorkflow \
—worksheet=myworksheet.csv \
—run=myrun —host=ui.grid.sara.nl
```

3. Submit: run on local, cluster or as grid pilot jobs

```
sh runPilots.sh ui.grid.sara.nl <user> <password> grid
```

umcg

molgenis

Simply monitor using the web user interface (incl. logs)

- http://localhost:8080/compute

Compute Testing

molgenis

| About | Object model | R-project API | HTTP API | REST API | Web Services API

Compute

File • Edit • View •

- ComputeParameters
- ComputeProtocols
- Workflow
- ComputeTasks
- ComputeHosts
- JobManager

ComputeTasks

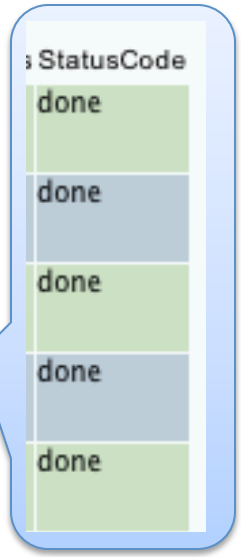
1 - 20 of 21

Search:

	id	name	ComputeScript	RunLog	WorkflowElement	Interpreter	PrevSteps	requirements	StatusCode
1.	1	impute2_s00_test1_1349185453555253000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s00_test1_1349185453555253000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s00	bash		1	done
2.	2	impute2_s01_test1_1349185453792052000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s01_test1_1349185453792052000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s01	bash	impute2_s00_test1_1349185453555253000	1	done
3.	3	impute2_s02_test1_1349185453886012000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s02_test1_1349185453886012000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s02	bash	impute2_s01_test1_1349185453792052000	1	done
4.	4	impute2_s03_test1_1349185454034169000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s03_test1_1349185454034169000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s03	bash	impute2_s02_test1_1349185453886012000	1	done
5.	5	impute2_s04_test1_1349185454255967000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454255967000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
6.	6	impute2_s04_test1_1349185454299992000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454299992000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
7.	7	impute2_s04_test1_1349185454357872000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454357872000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
8.	8	impute2_s04_test1_1349185454390958000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454390958000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
9.	9	impute2_s04_test1_1349185454415059000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454415059000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
10.	10	impute2_s04_test1_1349185454434245000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454434245000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
11.	11	impute2_s04_test1_1349185454459357000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454459357000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
12.	12	impute2_s04_test1_1349185454559252000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454559252000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
13.	13	impute2_s04_test1_1349185454601338000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454601338000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
14.	14	impute2_s04_test1_1349185454641825000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454641825000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
15.	15	impute2_s04_test1_1349185454728261000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454728261000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
16.	16	impute2_s04_test1_1349185454782979000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454782979000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
17.	17	impute2_s04_test1_1349185454882913000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befo...	TASKID:impute2_s04_test1_1349185454882913000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s04	bash	impute2_s03_test1_1349185454034169000	1	done
18.	18	impute2_s05_test1_1349185454927919000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befor...	TASKID:impute2_s05_test1_1349185454927919000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s05	bash	impute2_s04_test1_1349185454882913000	1	done
19.	19	impute2_s06_test1_1349185455059615000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befor...	TASKID:impute2_s06_test1_1349185455059615000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s06	bash	impute2_s05_test1_1349185454927919000	1	done
20.	20	impute2_s07_test1_1349185455113243000	##### BEFORE ##### touch $PBS_O_WORKDIR/jobname.out source $WORKDIR/tools/scripts/import.sh befor...	TASKID:impute2_s07_test1_1349185455113243000 touch: cannot touch `/opt/glite/var/tmp/jobname.out': N...	impute2_s07	bash	impute2_s06_test1_1349185455059615000	1	done

* = this record is readonly.

This database was generated using the open source MOLGENIS database generator version 4.0.0-testing.
Please cite Swertz et al (2010) and Arends & van der Velde et al (2012) on use.

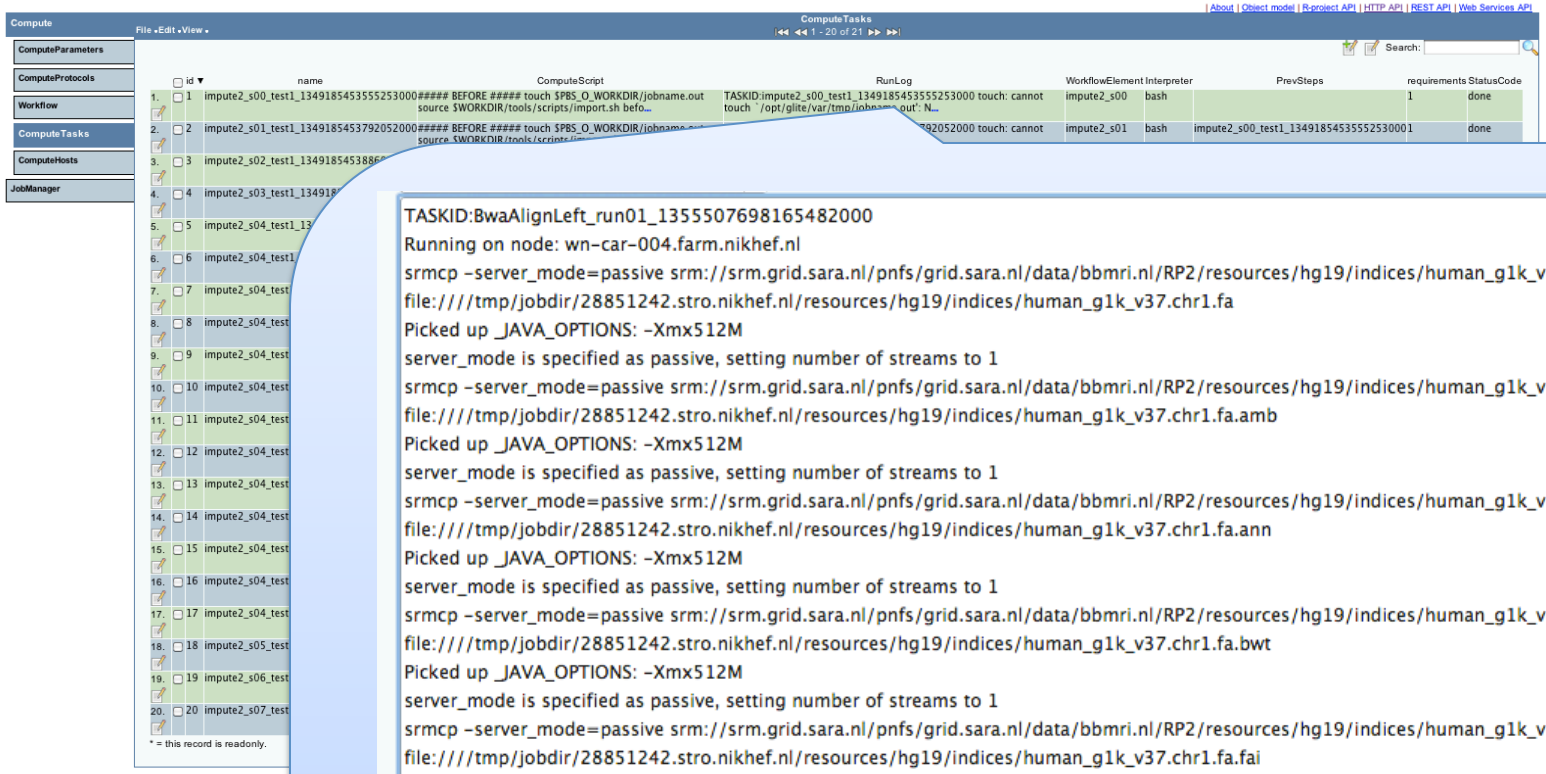


umcg

molgenis

Debugging and ‘retry’

- Just mark the job as ‘ready’ to start from middle
- All logs are easily available in database




```
TASKID:BwaAlignLeft_run01_1355507698165482000
Running on node: wn-car-004.farm.nikhef.nl
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.amb
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.amb
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.ann
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.ann
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.bwt
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.bwt
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.fai
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.fai
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.pac
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.pac
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
srmcp -server_mode=passive srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/RP2/resources/hg19/indices/human_g1k_v37.chr1.fa.rbwt
file:////tmp/jobdir/28851242.stro.nikhef.nl/resources/hg19/indices/human_g1k_v37.chr1.fa.rbwt
Picked up _JAVA_OPTIONS: -Xmx512M
server_mode is specified as passive, setting number of streams to 1
```

umcg

Alignment & SNP calling pipeline

- Manual: github.com/molgenis -> molgenis_apps/doc/compute

molgenis_apps / doc / compute / **03_compute_ngs.md**

mswertz an hour ago minor fixes to docs; added README.md

1 contributor

file | 193 lines (131 sloc) | 10.77 kb

Edit | Raw | Blame | History

Next-generation sequencing pipeline

Next-generation sequencing methods produce a growing volume of data, leading to increasing difficulties in analysing this data. This manual describes how one can simplify, parallelize and distribute such analysis across high performance compute architecture by using a standardized pipeline and the Molgenis Compute framework.

The pipeline is comprised of best-practice open-source software packages used in multiple institutions leading to 23 analysis steps. The four main parts of the pipeline are:

- *Alignment:* here alignment is performed using Burrows-Wheeler Aligner BWA. The produced SAM file is converted to a binary format using Picard and sorted afterwards.
- *Realignment:* in this part of the pipeline duplicate reads are marked using Picard. Afterwards realignment around known insertions and deletions (indels) from the Mills-Devine^1 dataset using the Genome Analysis ToolKit GATK is performed. If reads are re-aligned, the fix-mates step will update the coordinates of the reads mate.

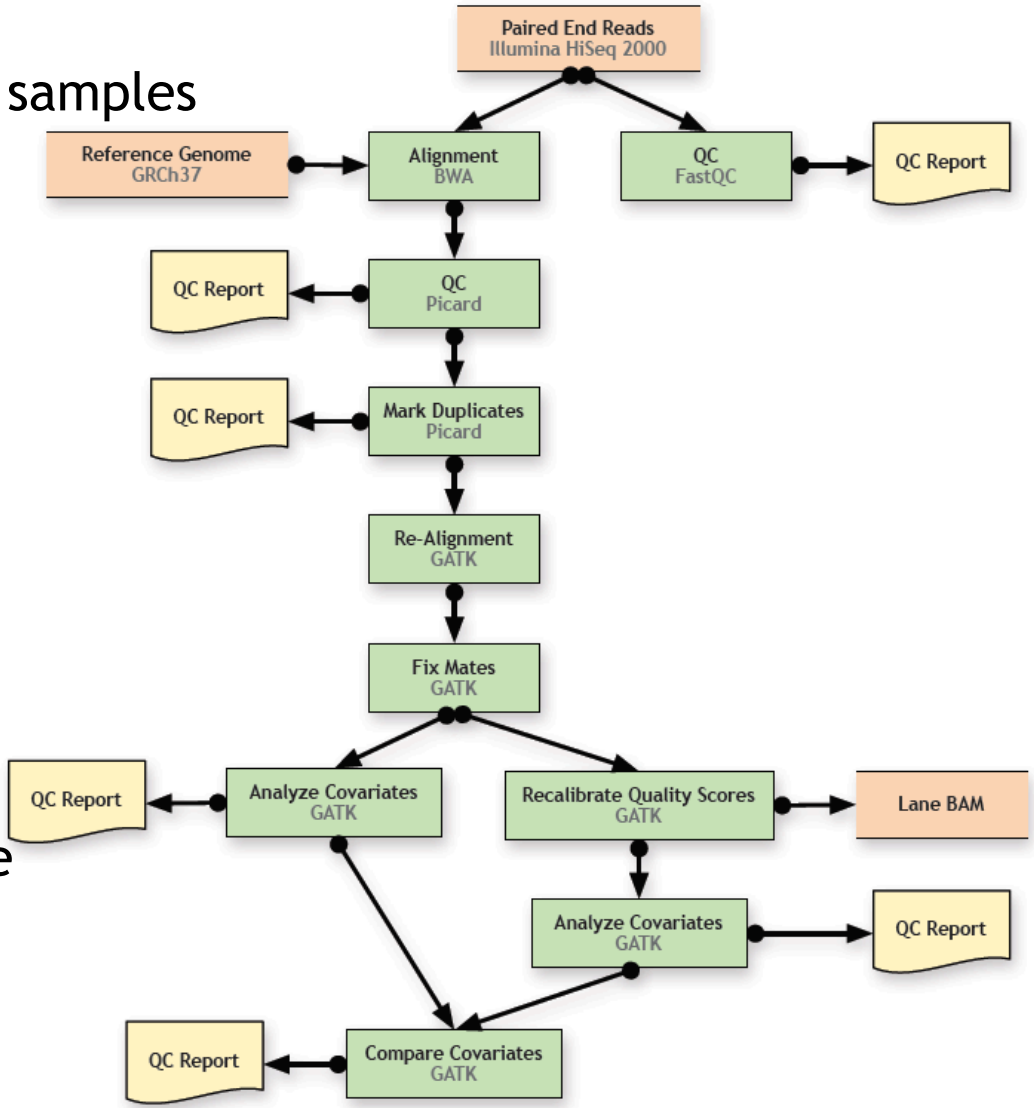
umcg

molgenis

Alignment & SNP calling pipeline

31 steps, ≥ 2 days per sample

- Inputs
  - Worksheet describing the samples
  - Sample *.fq files
  - Alignment reference VCF
- Outputs
  - BWA (re-)aligned BAMs
  - GATK SNP vcf
  - QC report
- Workflow
  - SRM upload data
  - Generate and run pipeline
  - SRM download data



umcg

Alignment & SNP calling pipeline

1. Create worksheet

A	B	C	D	E	F	G	H	I
externalSampleID	project	contact	sequencer	run	flowcell	lane	seqType	prepKit
Test_DNA	demo	researcher	SN163	457	BD0E5CACXX	4	PE	NEB

I	J	K	L	M	N
prepKit	capturingKit	arrayFile	arrayID	barcode	barcodeType
NEB	SureSelect_All_Exon_5	Alle_Samples	3	CAACCT	GAF

2. Generate your analysis run

```
sh molgenis_compute \
—workflowDir=workflows/ngs/alignAndSnpCall \
—worksheet=myworksheet.csv \
—run=myrun —host=ui.grid.sara.nl
```

3. Submit: run on local, cluster or as grid pilot jobs

```
sh runPilots.sh ui.grid.sara.nl <user> <password> grid
```

umcg

molgenis

Imputation pipeline

- Manual: github.com/molgenis -> molgenis_apps/doc/compute

molgenis_apps / doc / compute / **02_compute_imputation.md**

mswertz 2 hours ago re-added imputation docs

1 contributor

file | 304 lines (226 sloc) | 16.546 kb

Edit | Raw | Blame | History

Imputation pipeline

This manual explains how one can do imputation using the minimac^3 analysis pipeline in the Molgenis Compute framework. To run the analysis efficient it's needed to devide the analysis in four steps:

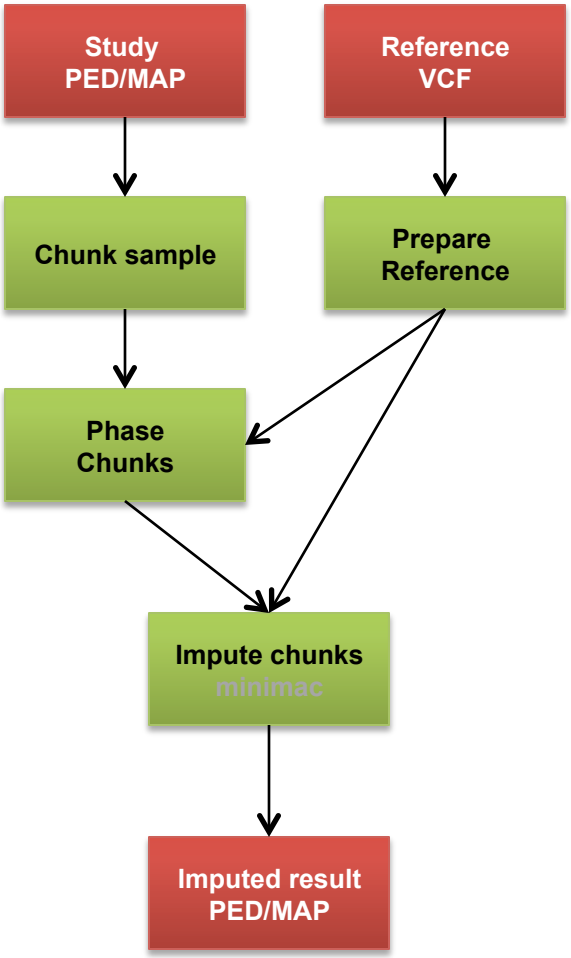
- *Preparing the reference:* here the reference is created from VCF (once per reference)
- *Preparing and QCing the study data:* aligns all alleles to the reference genome and performs quality control using ImputationTool ^7, chunks the study data in a user specified number of samples and splits the chromosome in chunks by splitting on a specified number of SNPs. This extensive chunking is needed to parallelize the analysis, leading to a total analysis time of approximately 10 hours per chunk of 2000 SNPs and 500 samples.
- *Phasing:* phases the data using MaCH ^6. The phasing only has to be done once for a specific study.
- *Imputation:* consist of imputing the phased data and concatenates the results per chromosome. Since the phasing is independant of the reference panel one only has to run the third step again when imputing with a different reference panel.

Using the above explained method imputation is parallelized in 'chunks' and ready to use on your cluster. For additional compute resources one can use the national Computing infrastructure for life Sciences, eBioGrid. How to setup Molgenis Compute for this grid infrastructure is explained in chapter six.

Minimac Imputation pipeline (within BBMRI-NL VO)

4 workflows, ≥ 12 hours per 500 sample/chr

- Inputs:
  - Worksheet
  - Study PED/MAP
  - Reference VCF
- Outputs:
  - Imputed PED/MAP
- Workflow
  - SRM upload study & reference
  - Prepare reference (wf1, once)
  - Chunk samples (wf2)
  - Phase & Impute (wf3+4)
  - SRM download imputed



umcg

molgenis

Generic ‘compute’ framework

1. Create worksheet

project	studyInputDir	prePhasingResultDir	imputationPipeline	genomeBuild	chr	autostart
projectname	directory	directory	beagle/mach/impute2	b36/b37	chromosome number	TRUE/FALSE

2. Generate your analysis run

```
sh molgenis_compute \
—workflowDir=workflows/prephase \
—worksheet=myref.csv \
—run=myrun  —host=ui.grid.sara.nl
```

3. Submit: run on local, cluster or as grid pilot jobs

```
sh runPilots.sh ui.grid.sara.nl <user> <password> grid
```

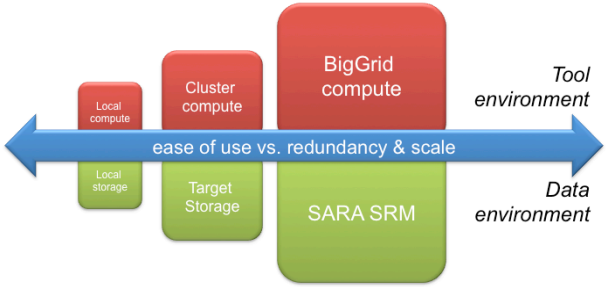
umcg

molgenis

BiG Grid
the dutch e-science grid

How to deploy

Harmonize the ‘backends’



umcg

molgenis

Harmonized file management: getFile/Putfile

- getFile and pu
  - are back-enc
  - now, we
    - check if th
    - do srm/lfr
- Input

```
getFile ${s
```

- Generated out

```
getFile $WO
imputationB
chr20.map
```

```bash
#!/bin/bash

getRemoteLocation()
{
    ARGS=($@)
    myFile=${ARGS[0]}
    remoteFile=srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/bbmri.nl/byelas${myFile:`expr length $TMPDIR`}
    echo $remoteFile
}

getFile()
{
    ARGS=($@)
    NUMBER="${#ARGS[@]}";
    if [ "$NUMBER" -eq "1" ]
    then

    myFile=${ARGS[0]}
    remoteFile=`getRemoteLocation $myFile`

    # 1. myPath = getPath( myFile ) will strip off the file name and return the path
    mkdir -p $(dirname "$myFile")

    # 2. cp srm:.../remoteFile myFile
    echo "srmcp -server_mode=passive $remoteFile file:///$myFile"
    srmcp -server_mode=passive $remoteFile file:///$myFile
    chmod 755 $myFile

    else
    echo "Example usage: getData \"\$TMPDIR/datadir/myfile.txt\""
    fi
}

putFile()
{
    ARGS=($@)
    NUMBER="${#ARGS[@]}";
    if [ "$NUMBER" -eq "1" ]
    then
    myFile=${ARGS[0]}
    remoteFile=`getRemoteLocation $myFile`
    echo "srmcp -server_mode=passive file:///$myFile $remoteFile"
    srmcp -server_mode=passive file:///$myFile $remoteFile
    else
```

umcg

“Harmonized” tool management

Tool in input sandbox
“getFile(‘tool.zip’)”

In $WORKDIR

- Download

Tool deployed as
“load module”

In $VO_BBMRI_NL_SW_DIR

- Download
- Configure
- Compile
- Install

Static linked binary
without dependencies

Dynamic linked binary
With complex dependencies

‘Harmonized’ tool management: modules

- Build on standard ‘modulecmd’ (apt-get install ....)
- http://www.bbmriwiki.nl/svn/ebiogrid/modules/

```
#%Module1.0#####################################################################
##
## bwa 0.5.8c_patched modulefile
##

proc ModulesHelp { } {
        global toolversion
        global toolname

        puts stderr "Set up the environment for $toolname version $toolversion\n"
}

# for Tcl script use only
set toolname         bwa
set toolversion      0.5.8c_patched
set tooldir          "$env(VO_BBMRI_NL_SW_DIR)/tools/$toolname-$toolversion"

module-whatis   "Sets $toolname environment."
prepend-path    PATH    "$tooldir/"
setenv          BWADIR  "$tooldir"
```

umcg

molgenis

Modules installed (should we coordinate with others?)

- GATK/1.0.5069
- Python/2.7.3
- R/2.14.2
- bwa/0.5.8c_patched
- capturing_kits/SureSelect_All_Exon_30MB_V2
- capturing_kits/SureSelect_All_Exon_50MB
- capturing_kits/SureSelect_All_Exon_G3362
- fastqc/v0.10.1
- fastqc/v0.7.0
- gtool/v0.7.5_x86_64
- impute/v2.2.2_x86_64_static
- jdk/1.6.0_33
- mach/1.0.18
- minimac/beta-2012.10.3
- picard-tools/1.61
- plink/1.07-x86_64
- plink/1.08

Deployment is a b$tch

- Debugging on the grid is hard
  - Can’t login to each cluster to see why it fails
  - Typical deployment script is > 100 lines
- NB: the UI is *not* the same as the grid sites re modules
- Should we instead have gone for *one* module
  - in the form of a VM??

Pitfalls

- File management
  - File permissions
  - SRM does not know directories as we do
  - ‘normal’ users still ask us to do their file management
- How many files do we have now?
  - diskpool info: ~143 TB
- How many cpu hours did we burn yet?
  - not measured: Jan/Barbara aligned; Jan/Laurent/Kai have called indels; Alex has imputed

BiG Grid
the dutch e-science grid

Concluding remarks

umcg

molgenis

Project

Done:

- D01: Backend MOLGENIS/GRID integration based on pilot framework
- D02: BBMRI-NL alignment pipelines integrated and running
- D03: Minimal data tracking and pipeline orchestration dashboard
- D04: Education: collaborators assisted to use alignment
- D05: Imputation pipelines added; pipelines running on D01

Partly done (still ongoing):

- D06: Workflow wizards and user friendly monitor tools added on D03
- D07: Education: BBMRI-NL team assisted to impute 100,000 GWAS

Comments:

- We dramatically *underestimated* the work to deploy files and tools on the grid. We did not realize the grid is not homogeneous.
- We had to shorten our project plan from 2 to 1.5 years because end of eBioGrid. This cancelled our Galaxy connectivity plans.

When to use ‘compute’ versus e.g. ‘galaxy/taverna’

- Use compute when
  - You are a programming bioinformatician
  - Working with big data files
  - Running large relative routine pipelines
  - Need to debug (all scripts + logs are fully available)
  - You assume you will want to change the scripts

- Use galaxy/taverna when
  - You prefer graphical user interfaces
  - Working with limited amount of data
  - Running ad-hoc pipelines in a exploratory fashion
  - Don’t expect to debug (scripts + logs are hidden)
  - You asume that everything just works

Read more

- Publications
  - Byelas et al, Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms, Barcelona, Spain, 11 - 15 February, 2013
  - Byelas et al, Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms, Vilamoura, Algarve, Portugal, 1 - 4 February, 2012
  - Byelas et al, Proceedings of the 19th International Euromicro Conference on Parallel, Distributed and Network-based Processing, PDP 2011, Ayia Napa, Cyprus, 9-11 February 2011.
- Demo sessions
  - MOLGENIS/compute workshop at BioAssist meeting, 12 oct '12
  - MOLGENIS/compute demo at the 4th IWSG4LS, 23-25 may '12
  - MOLGENIS/compute demo at NBIC-2012, 23 apr '12
- Posters
  - Netherlands Proteomics Conference, NPC-2011, 13 jan '11
  - Connecting Biobanks - Towards a Dutch Biobanking Hub, 19 nov '12

Further steps

- We are using it daily
  - Impute 15-20k samples on BigGrid
  - Move in-house analysis to BigGrid (requires private VO)
  - Support BBMRI-NL RP-3
- Further improve grid usage
  - Smarter pilots with automatic module detection
  - Deploy pilot database on SARA cloud?
- Polish supporting tools
  - Add smart defaults and better warnings
  - Automatic execution logs evaluation (instead of by hand)
  - File storage synchronization system (with CTMM, SARA?)
  - Graphical analysis of big workflows (visual analytics)
  - Execution statistics (especially for on the grid)
- Support is welcome ☺

Acknowledgements

eBioGrid biobanking team

- George Byelas
- Pieter Neerincx
- Martijn Dijkstra
- Feerk van Dijk
- Ger Strikwerda
- Wil Bruins-Koetsier
- Morris Swertz
- Jan Bot
- Mathijs Kattenberg
- Tom Visser
- Barbara van Schaik
- David van Enckevort
- Irene Nooren
- eBioGrid/BBMRI-NL/NBIC

And you, our local, national and international collaborators

umcg

molgenis

We use ‘compute’ daily, you are welcome to join ☺

Fork me on GitHub

http://www.molgenis.org/wiki/ComputeStart

1. Generic ‘compute’ framework + operating procedures
   - Design: workflows, protocols, parameters
   - Run: worksheets, command-line submit, pilot jobs database
   - Deploy: uniform tool and file management

2. Imputation pipelines
   - Preparing reference data (once) — 4 steps, 20min
   - QC and chunk the study data — 2 steps, 20min/chr
   - Phase the study data per chunk — 2 steps, 6h/500 sample/chr
   - Impute the study data and merge — 2 steps, 6h/500 sample/chr

3. Whole exome/genome sequencing pipelines
   - Sequence alignment, realignment — 17 steps, 5-6 days/lane/barcode
   - Variant calling & QC — 13 steps, 2 days/sample

umcg

Contact: m.a.swertz@rug.nl

molgenis