Using Insyght
“In the cloud” with the IFB appliance
“Bacterial genomics”
-
User guide
(Version 2.1.1, Feb 2019)

Citation Insyght:
Lacroix T., Loux V., Gendrault A., Hoebeke M., and Gibrat J-F. (2014) Insyght: navigating amongst abundant homologues, syntenies and gene functional annotations in bacteria, it's that symbol! Nucleic Acids Res. 42(21):e162. PMID: 25249626

Content:

1 Getting started with visualizing your data ... 3
1.1 Launch a "Bacterial genomics" appliance from the IFB dashboard ... 3
1.2 Launch Insyght ... 3
1.3 How to connect to your VM in the cloud via a terminal ... 3
1.4 [Optional] Remove all current data in your instance of the database ... 4
1.5 Add your own data ... 4
1.5.1 Delete the temporary and log files generated by the pipeline from the previous run ... 4
1.5.2 [Optional] Download genome files from ncbi ... 5
1.5.3 Gather all the .gbk, .dat and .gbff genomes files you want to add in ${BANK_DIR} ... 5
1.5.4 Run the Insyght pipeline to add the data ... 6
1.5.5 See the results ... 8
2 Other information ... 9
2.1 Contact and report error ... 9
2.2 Overview of the pipeline ... 9
2.3 How to manually connect to the PostgreSQL database ... 10
2.4 Reclaim disk space of deleted rows and re-calculate the indexes of the database ... 10
2.5 Other useful command line tips ... 11
2.6 Changing the threshold parameters for the programs blast (used to find gene similarity) and align (used to compute the synteny) ... 11
2.7 Different ways to download a raw fasta or annotated genome file from the ncbi or ebi repositories ... 12
2.7.1 Download genome files from ncbi using script ... 12
2.7.2 Browse and download http content with a web browser ... 13
2.7.3 Browse and download ftp content with a web browser ... 14
2.8 Run prokka to quickly annotate raw fasta file ... 14
2.9 Using the IFB cloud cluster to speed up processing ... 15

1 Getting started with visualizing your data

~~ Remark: Insyght expects a transcript per CDS and is therefore only suitable for Bacteria and Archea.

1.1 Launch a “Bacterial genomics” appliance from the IFB dashboard

Login to the IFB Bioinformatics Cloud at https://biosphere.france-bioinformatique.fr/ and launch (“Advanced launch”) the appliance “Bacterial genomics (Insyght)”. Allow some time (30-45 minutes) for the virtual machine to start.

~~ Remark: It is recommended to create an appliance with minimum 2 vCPU and 8Go GB RAM for the “master”. The local disk space for the master VM depends on how many genomes you want to insert, see the benchmarking table below for details (i.e. 40Go GB for a dataset of ~150 genomes). You can choose to add 1 slave or more. Slaves will be used to speed up the processing of the data (parallel processing). Each slave can be as small as 1 vCPU, 512Mo GB RAM, 1Go GB local disk.

1.2 Launch Insyght

After the appliance starts, copy the “https” link from the “Access” tab to a web browser. You can access your instance of Insyght from any computer connected to the internet. The url provided by the “http” link is active as long as the VM is turned on.

~~ Remark: You may experience an error message if you try to connect to Insyght and the booting process of the VM is not fully completed. In that case, try to connect again in a few minutes.

~~ Remark: Depending on the deployment cloud, some of the ways to access the web application describe thereafter may work or not. If an http link does not work, please try modifying the port or trailing slash as shown in the example below:
http://ip_of_the_virtual_machine_provided_in_link** :80/Insyght/**
http://ip_of_the_virtual_machine_provided_in_link** :80/Insyght**
http://ip_of_the_virtual_machine_provided_in_link** :8080/Insyght/**
http://ip_of_the_virtual_machine_provided_in_link** :8080/Insyght**

If you already inserted some data previously, you should be able to browse them. If you haven’t, a message will appear saying that the database is empty. Follow this guide to populate the database with your data. Links are also available to showcase the tool’s features or to access different manuals and guides (https://sourcesup.renater.fr/frs/?group_id=3679).

1.3 How to connect to your VM in the cloud via a terminal

From the IFB Cloud's dashboard, use the "ssh" link from the "Access" tab to connect into a terminal. An example of command you can type from a terminal is:

```
ssh ubuntu@**ip_of_the_virtual_machine_provided_in_link**
```

1.4 [Optional] Remove all current data in your instance of the database

Optionally, you can remove all the current data in your instance of the database before inserting new data. Insyght relies on this database to fetch any information it displays. Skip this section if you want to keep the current data and add yours along with them. The process of adding data in Insyght is incremental and can be done multiple times without deleting the current data.

If you wish to remove the current data in the database, type in the following command in a terminal:

```
perl -I $SCRIPTS_DIR $SCRIPTS_DIR/truncate_all_db_tables.pl
```

~~ *Warning: this action cannot be undone*

~~ *Remark: throughout the document, the text that is **highlighted in red** is to be copied/paste or typed in a terminal connected to the VM.*

~~ *Remark: The database should now be empty. If you open the Insyght web application, you should see that no organism is available.*

1.5 Add your own data

This section explains how to add your own data with Insyght.

~~ *Remark: we suggest that you try loading the demo data first to check that the pipeline is working correctly. See the steps below to load the demo data.*

~~ *Remark: The command thereafter uses "screen" therefore closing the terminal will not end the pipeline run. You can open another terminal anytime and re-attach the process of the pipeline as needed (see steps below). Using crtl-C in the terminal (or shutting down the VM) while the command runs will end it.*

~~ *Benchmark: benchmarking was carried out with the parameters: VM master 2CPU, 8GB RAM ; VM slaves = 1CPU, 2GB RAM.*

1.5.1 Delete the temporary and log files generated by the pipeline from the previous run

From a terminal connected to the VM, copy the following commands:

```
rm -f -r ${DATA_DIR}/*
rm -f -r ${LOG_DIR}/*
rm -f -r ${BANK_DIR}/*
```

1.5.2 [Optional] Download genome files from ncbi

See the section "*Different ways to download a raw fasta or annotated genome file from the ncbi or ebi repositories*" for more details. if you want to annotate raw fasta files, you can quickly annotate them with prokka which will automatically generate a genbank file. Prokka (http://www.vicbioinformatics.com/software.prokka.shtml) is a software tool for the rapid annotation of prokaryotic genomes developed by the Victorian Bioinformatics Consortium. For more details, see the section "Run prokka to quickly annotate raw fasta file".

1.5.3 Gather all the .gbk, .dat and .gbff genomes files you want to add in ${BANK_DIR}

Copy or move the genomes files (.gbk, .dat, and .gbff) in the directory ${BANK_DIR}.

```
cp ***PATH_TO_YOUR_FILE*** ${BANK_DIR}
```

Repeat the command for each file to copy. For example, if you did download the complete genomes files from Refseq for Enterococcus faecalis, Acidobacteriia, and Escherichia albertii following the command line given as an example in "*Download genome files from ncbi using script*", do:

```
cp ~/Downloads/* ${BANK_DIR}
```

List the files that are going to be inserted:

```
ll $BANK_DIR/
```

~~ *Remark: A set of demo genome files can be found under the directory ${SCRIPTS_DIR}/../../ORIGAMI/DEMO_FILES/ANNOTATED_GENOMES/BATCH1/ If you want to load the demo data to check that the pipeline is working correctly, use the command:*

```
cp ${SCRIPTS_DIR}/../DEMO_FILES\
/ANNOTATED_GENOMES/BATCH1/* ${BANK_DIR}/
```

~~ *Remark: To download genome files from ncbi or ebi, see the section "Different ways to download a raw fasta or annotated genome file from the ncbi or ebi repositories". If you use the script download_genome_files_from_ncbi.pl, the files will be downloaded directly into the ${BANK_DIR} directory so there is no need to copy them.*

~~ *Remark: If you did annotate some fasta file with prokka, do not forget to copy them as well, typically with the command:*

```
cp /root/mydisk/prokka_output/*.gbk ${BANK_DIR}/
```

~~ *Remark: Compressed files with the .gz extension are supported by the pipeline, no need to extract them.*

1.5.4 Run the Insyght pipeline to add the data

This script will perform preliminary preparations of the genome files, retrieve primary data (organisms, genes, features, etc.), compute the cross comparison of all the CDSs, infer the syntenies, and insert the data into the database. For an overview of the pipeline, see the section "*Overview of the pipeline*". To run the script:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -PROGRAM_TO_USE PLAST
```

This script may take some time hence the use of screen to detach the process from the terminal as needed and to avoid closing the process if the terminal closes. To detach the process, you can close the terminal that runs this script or type "ctrl-a d". At a later time, open another terminal and re-attach the process of the pipeline as needed by typing:

```
screen -D -R
```

Closing the terminal will not end the screen command but using crtl-C in the terminal while the command runs or shutting down the VM will end it.
If you have detached the process and want to check whether the script is still running or not, use the command:

```
screen -ls
```

If the script is done, you should see an output such as:

```
No Sockets found …
```

If the script is still running, you should see an output such as:

```
There are screens on…
```

Alternatively, you can use **top** to see the list of all the programs that are running or check the log file for this step (see below).

~~ *Remark: Information will be printed on the screen while the script run. If the script is successful, you should see:*

```
run_Insyght_pipeline.pl successfully completed at DATE
```

printed on the last line. To access the log file, use

```
less ${LOG_DIR}/run_Insyght_pipeline/run_Insyght_pipeline.log
```

~~ *Benchmark: The time and resources needed to process the data grow with the number of genomes to compare. When using Plast (see below):*

Number of genomes	*Processing time*	*Final size of database*	*Max size needed for OS (~10 GB) + tmp files*	*Min local disk space when choosing master VM*
17 (demo dataset)	*~1h*	*~325 MB*	*~14 GB*	*20 GB*
50	*~8h45min*	*~1.7 GB*	*~18 GB*	*20 GB*
100	*~34h*	*~4.8 GB*	*~22 GB*	*40 GB*
150	*~108h*	*~10 GB*	*~33 GB*	*40 GB*
200	*~152h*	*~17 GB*	*~45 GB*	*80 GB*
250	*~259h*	*~25 GB*	*~65 GB*	*80 GB*

When using multiple parallel processes (multicore VM or cluster, see below), it approximately divides the time it takes to process the genomes by the number of jobs in parallel (option -MAX_JOBS, see below) plus an extra hour. For example, with -MAX_JOBS 20 (20 processes in parallel, either on local CPU or cluster) it takes ~7 hours to process 150 genomes instead of 108 hours with a single thread. The pipeline can be used incrementally without deleting previous entries. In this case, each new organism will to be compared to new and previous entries.

~~ Recommended options for this script (you can combine them):

-PROGRAM_TO_USE {PLAST,BLAST}
As an alternative to Blast, you have the option to use Plast (https://plast.inria.fr/). With the default parameters and for most cases, Plast is about 5 times faster than Blast but less sensitive (about 2 times less alignments found). However, regarding BiDirectional Best Hits (BDBH), this decrease in sensitivity is not significant. As Insyght relies primarily on BDBH, Plast therefore gives results comparable to Blast for our purpose. Blast is the default sequences alignment tool if you omit the -PROGRAM_TO_USE option or if you use:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -PROGRAM_TO_USE BLAST
```

Using Blast, this step took ~6 hours for the 17 organisms provided as demo.
If you wish to use Plast instead, use the following option:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -PROGRAM_TO_USE PLAST
```

Using Plast, this step took ~1 hours for 17 organisms provided as demo.

-MAX_JOBS {POSITIVE_DIGIT}
To parallelize the processes and make this step run faster, you can use the option -MAX_JOBS. The total time for this step will decrease proportionally with the number of jobs launched in parallel. With -MAX_JOBS 17, this step took ~10 minutes for the 17 organisms provided as demo:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -MAX_JOBS 17
```

The -MAX_JOBS value must be at most the number of available processors. To know the number of internal processors available, type in the command:

```
grep -c processor /proc/cpuinfo
```

You can configure your VM to have as much internal processors as the host machine. If the number of available processors is limited on your host machine, a solution to greatly increase the number of available processors is to use a cluster (external processors, see option -CMD_CLUSTER below).

-CMD_CLUSTER {**COMMAND_SUBMIT_CLUSTER*}
See the section "*Using the IFB cloud cluster*" for more details. If the CMD_CLUSTER argument is omitted (default), then the command is executed by the bash locally on the internal processors.

~~ *Other options for this script:*

- To change the default threshold parameters for homologs and syntenies, see the section "*Changing the threshold parameters for the programs blast (used to find gene similarity) and align (used to compute the synteny)*".

-FORCE_SPLIT_ACCNUM_INTO_SEPARATE_ENTRIES {ON, OFF} # default OFF
If you want each element (chromosomes, plasmids, etc.) of an organism to be treated as its own entry instead of being grouped by organism/assembly, turn this option ON:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -FORCE_SPLIT_ACCNUM_INTO_SEPARATE_ENTRIES ON
```

1.5.5 See the results

That is it, launch Insyght (see section "Launch Insyght") to see the results!

2 Other information

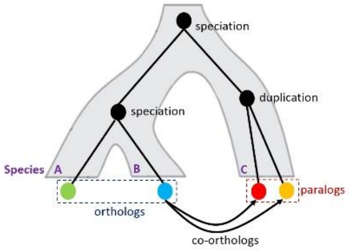
2.1 Contact and report error

- Email us: insyght[AT]inra.fr
- Report a bug: Go to https://migale.jouy.inra.fr/redmine/projects/insyght and then click on "Issues" to see the list of already reported bugs and features. You can browse the list anonymously but you will need to register / login to submit a new one. Once you register / login (click on "sign in / register" on the top right hand corner), a tab "New issue" appears. If you want to report a bug, please describe the organism / element / gene / action that led to it and the web browser you are using. You can also attach documents such as screenshots.

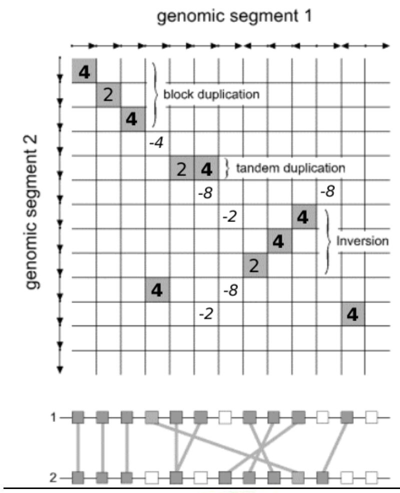
2.2 Overview of the pipeline

I/ The automated pipeline carries out the following tasks:

1. Process the genome files (.embl, .gbk, ou .gbff format) to extract the primary data (genomic annotations).



2. Find homologs (BLASTp).



3. Find syntenies.

4. Insert the data into the relational database.

II/ The Insyght web application is used to visualise the data and carry out the data mining.

2.3 How to manually connect to the PostgreSQL database

In a terminal connected to the VM, type in or paste the following command:

```
psql -h localhost -p 5432 -U origami_admin -d origami_prod
```

You are now connected to your instance of the origami database. If you wish, you can use plain sql commands to query the data as you like.

~~ *Remark: type in* **\q** *to exit the psql mode.*

2.4 Reclaim disk space of deleted rows and re-calculate the indexes of the database

To reclaim disk space following important changes to the database or if over time the web application feels sluggish, it is best to clean up the database underlining storage and indexing mechanisms. Type in the following command in a terminal:

```
vacuumdb --full --analyze -h localhost -p 5432 -U origami_admin origami_prod
```

~~ Remark: safely ignore the warnings on the terminal

2.5 Other useful command line tips

- Check free disk space: **df -h .**

- Check database size: connect to the database and then type: **\l+**

- Count inserted data into the database:

```
perl -I $SCRIPTS_DIR $SCRIPTS_DIR/count_inserted_data.pl
```

- For processes that are running for a long time, detach the process from the terminal by using nohup (https://en.wikipedia.org/wiki/Nohup) or screen (https://www.gnu.org/software/screen/manual/screen.html).

- Check what programs are running: **top**

- To access the log files of tomcat: **cd /var/lib/tomcat8/logs**

2.6 Changing the threshold parameters for the programs blast (used to find gene similarity) and align (used to compute the synteny)

Prior to running the step “Add your own data in the database”, you can configure the blast threshold parameter (e-value). Open the file BlastConfig.pm and change the threshold value to your liking (default is -thresh => '5e-2').

```
vim ${SCRIPTS_DIR}/BlastConfig.pm
```

Synteny regions are computed with a dynamic programming algorithm called “align” to determine the highest scoring paths among the chain of collinear homologs. By default, small gaps are allowed within the conserved synteny. You can customize those parameters during the step “Syntenies generator”. This script supports the following arguments:

-MAIN_ALIGN_OPTION_os: *defines the ortholog (BDBH) score; default value = 4; must be integer > 0.*

-MAIN_ALIGN_OPTION_hs: *defines the homolog (non-BDBH but significative blast result) score; default value = 2; must be integer > 0.*

-MAIN_ALIGN_OPTION_mp: *defines the mismatch penalty score within a synteny; default value = -4; must be integer < 0.*

-MAIN_ALIGN_OPTION_gc: *defines the gap creation score within a synteny; default value = -3; must be integer < 0. Must be different than the mismatch penalty score due to a current bug.*

-MAIN_ALIGN_OPTION_ge: *defines the gap extension score within a synteny; default value = -3; must be integer < 0. Must be identical to the gap creation score due to a current bug.*

-MAIN_ALIGN_OPTION_m: *defines the minimum size for a synteny (number of CDS); default value = 1; must be integer > 0.*

-MAIN_ALIGN_OPTION_c: *defines the cutoff score for a synteny; default value = 8; must be integer > 0.*
-MAIN_ALIGN_OPTION_o: *defines if the program must include all orthologs regardless of the cutoff score; default value = 1; must be integer 1 (Yes) or 0 (No).*
-MAIN_ALIGN_OPTION_pf: *defines the minimum protein fraction for Ortholog; default value = 0.5; must be double between 0 and 1.*

For example, here is a command that uses two of the optional arguments above:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -MAIN_ALIGN_OPTION_os 8 \
 -MAIN_ALIGN_OPTION_hs 4
```

~~ Remark: As you can add data incrementally, it is possible (although not advised) to insert data computed with 2 different sets of blast or align parameters. Comparing the results between data generated with different sets of parameters is strongly discouraged however. A warning that the database is deemed not stable will be raised.

2.7 Different ways to download a raw fasta or annotated genome file from the ncbi or ebi repositories

2.7.1 Download genome files from ncbi using script

The files downloaded by this script will be stored directly into the ${BANK_DIR} directory. To see the files already in ${BANK_DIR}:

```
ll ${BANK_DIR}
```

Run the script download_genome_files_from_ncbi.pl. For example, to download complete genomes files from Refseq for Enterococcus faecalis (taxon id 1351), Acidobacteriia (taxon id 204432), and Escherichia albertii (taxon id 208962), use the following command:

```
perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/download_genome_files_from_ncbi.pl \
 -NCBI_ASSEMBLY_SUMMARY_FILE DEFAULT_refseq_bacteria \
 -RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES 1351 \
 -RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES 204432 \
 -RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES 208962 \
 -RESTRICT_TO_COMPLETE_GENOME ON \
 -RESTRICT_TO_LATEST_ASSEMBLY ON
```

The option -NCBI_ASSEMBLY_SUMMARY_FILE can be a path to the assembly file of your own or one of the 4 default values:
DEFAULT_genbank_archaea, DEFAULT_refseq_archaea,
DEFAULT_genbank_bacteria, DEFAULT_refseq_bacteria.
You can mix archaea and bacteria assembly summary files if you want by repeating this option multiple times, but do not mix genbank and refseq assembly files that are redundant over genomes else their elements will be duplicated in the database also.

At least one -NCBI_ASSEMBLY_SUMMARY_FILE option or -UNIPROT_GENOME_TABLE_FILE option (see below) is mandatory in order to provide the script with assembly accession. If neither is provided, the script will not download anything. If you wish to use Uniprot, you can download a Uniprot genome table file from http://www.uniprot.org/proteomes/ by creating you list of genomes of interest, clicking on “Columns” to add all columns, and then clicking on “Download”.
If the option -RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES is omitted, all the assemblies in the assembly file will be downloaded. The option -RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES expects NCBI taxon ids (browse the NCBI Taxonomy database to get the taxon id for your taxonomic node of interest: https://www.ncbi.nlm.nih.gov/taxonomy).

The complete list of options for the script download_genome_files_from_ncbi.pl is as follow:
-NCBI_ASSEMBLY_SUMMARY_FILE {**PATH_TO_FILE**, DEFAULT_genbank_archaea, DEFAULT_refseq_archaea, DEFAULT_genbank_bacteria, DEFAULT_refseq_bacteria}
-UNIPROT_GENOME_TABLE_FILE {**PATH_TO_FILE**}
-RESTRICT_TO_TAXA_DOMAIN {archaea, bacteria}
-RESTRICT_TO_DATABANK {genbank, refseq}
-RESTRICT_TO_LATEST_ASSEMBLY {ON, OFF} #default ON
-RESTRICT_TO_COMPLETE_GENOME {ON, OFF} #default ON
-RESTRICT_TO_LIST_TAXON_IDS_AND_SUBNODES {NCBI taxon id}
-LIMIT_TO_X_RANDOM_SAMPLES {POSITIVE DIGIT}
-LIMIT_TO_X_FIRST_ORDERED_SAMPLES {POSITIVE DIGIT}
-LIST_AND_EXIT {ON, OFF} #default OFF
-SOFTLINK_LOCAL_PUBLIC_REPOSITORY_IF_AVAILABLE {**PATH_TO_DIR**}

To access the log file for this script:

```
less $LOG_DIR/download/download_genome_files_from_ncbi.pl
```

To check the list of files that were downloaded:

```
ll ${BANK_DIR}
```

2.7.2 Browse and download http content with a web browser

- ncbi http server : http://www.ncbi.nlm.nih.gov/guide/genomes-maps/
- ebi http server : http://www.ebi.ac.uk/genomes/bacteria.html or http://bacteria.ensembl.org/index.html

~~ *Remark: There is no web browser installed on the IFB VM but you can use the web browser on your host machine to download the files and then transfer them to the IFB VM using the command provided in the dashboard.*

~~ *Remark: Upon download on your host machine, the files will be stored in the default “Downloads” directory associated with the web browser.*

2.7.3 Browse and download ftp content with a web browser

- ncbi ftp server : ftp://ftp.ncbi.nih.gov/

~~ *Remark: To access the genbank content go to ftp://ftp.ncbi.nih.gov/genomes/genbank/. To access the refseq content go to ftp://ftp.ncbi.nih.gov/genomes/refseq/. The ncbi refseq repository is less exhaustive than genbank but contains only annotated genomes.*

~~ *Remark: To search a particular word within the web page, use ctrl-f. You can use this trick to find your genome of interest among the long list of organisms.*

~~ *Remark: Prefer the latest assembly for a given organism (directory latest_assembly_versions). The .fna files are the raw fasta file, the .gbff files are the annotated genome files (if any).*

- ebi ftp server : ftp://ftp.ensemblgenomes.org/pub/bacteria/current

2.8 Run prokka to quickly annotate raw fasta file

Prokka (http://www.vicbioinformatics.com/software.prokka.shtml) is a software tool for the rapid annotation of prokaryotic genomes developed by Victorian Bioinformatics Consortium. It is pre-installed on this virtual machine. The following commands delete any previous output and run prokka on the example file:

```
rm -f -r /root/mydisk/prokka_output/*
prokka --compliant --addgenes --mincontiglen 200 \
--outdir /root/mydisk/prokka_output \
--genus Acetohalobium --species arabaticum --strain DSM_5501 \
--centre JOUY --locustag AarD \
--force \
$SCRIPTS_DIR/../DEMO_FILES/GENOMIC_FASTA/GCF_000270085.1_ASM27008v1_genomic.fna
```

To annotate your own fasta file, just substitute the example filename above with the path to your file and adapt the different arguments of prokka (--genus, --species etc.) accordingly.

~~ *Remark: Information will be printed on the screen while the script run. If the script is successful, you should see a list of output files followed by a reference to the prokka paper printed on the last lines. To see the resulting files generated by prokka, type:*

```
ll /root/mydisk/prokka_output
```

The file that we are interested in is the .gbk file; to display it, type:

```
less /root/mydisk/prokka_output/*.gbk
```

~~ Remark on the other prokka arguments: type:

```
prokka -help
```

~~ Benchmark: This step took ~10 minutes for the fasta file provided as demo with 1 CPU in use.

2.9 Using the IFB cloud cluster to speed up processing

When you deploy the appliance in advanced mode, you can specify how many slaves you want. Choose as many slaves as you would like parallel jobs to be executed. When using multiple parallel processes (multicore VM or cluster ni this case), it approximately divides the time it takes to process the genomes by the number of jobs in parallel (option -MAX_JOBS, see below) plus an extra hour. For example, with -MAX_JOBS 20 (20 processes in parallel, either on local cpu or cluster) it takes ~7 hours to process 150 genomes instead of 108 hours with a single thread. You can use the arguments **-MAX_JOBS** (must be equivalent to the number of slaves you choose) and **-CMD_CLUSTER**. Here is an example commands:

```
screen perl -I $SCRIPTS_DIR/ $SCRIPTS_DIR/run_Insyght_pipeline.pl \
 -MAX_JOBS 20 -CMD_CLUSTER "qsub -m ea" -PROGRAM_TO_USE PLAST
```

~~ Remark: Make sure to customize the -MAX_JOBS option to the number of slaves in your cluster when you launch the appliance.