Genevestigator Export Formats

Revision: 2.3 (07/10/2020)

This document describes in detail and with examples the output format of the different files that can be exported by the tools contained in the `DataExportTool` package of Genevestigator. The actual tools and commands that produce those files are explained in the accompanying document, *Genevestigator Export Tools*.

In this document we will use the following conventions:

- `[x.y]` represents the value of attribute y (e.g., caption) in entity x (e.g., platform),
- `...` indicates that an example of output was shortend.

1 exporttool.sh

1.1 Exported Files

The following set of files can be exported for a given platform, either by multiple calls to the `exporttool.sh` script with the respective command name, or by calling the `dataexport.sh` wrapper script once (the latter internally uses `exporttool.sh`). The output file or directory as well as the target platform must be supplied to the script (see *Genevestigator Export Tools* document). The platform is uniquely identified by its short name, i.e. `[platform.short_caption]`:

Filename	Description
`experiments_[platform.short_caption].csv`	experiment descriptions
`experimenttree_[platform.short_caption].csv`	experiment-replica-chip relations
`anatomytree_[platform.short_caption].csv`	anatomy ontology and annotations
`celltypetree_[platform.short_caption].csv`	cell type ontology and annotations
`celllinetree_[platform.short_caption].csv`	cell line ontology and annotations
`neoplasmtree_[platform.short_caption].csv`	neoplasm ontology and annotations
`developmenttree_[platform.short_caption].csv`	development ontology and annotations
`propertytree_[platform.short_caption].csv`	state variable ontology and annotations
`stimulustree_[platform.short_caption].csv`	stimulus ontology, annotations and comparisons
`mutationtree_[platform.short_caption].csv`	mutation ontology, annotations and comparisons
`expressionmatrix_[platform.short_caption].csv`	expression data as a matrix

Table 1: Files that are exported by `dataexport.sh`

Additionally, a few more files that are not exported by `dataexport.sh` can be exported through the `exporttool.sh` script with the appropriate command and options (see Table 2). The output format of all files in Table 1 and Table 2 will be described in the following subsections.

Filename	Description
`platforms.csv`	platforms descriptions
`orthologpairs_[platform1]_[platform2].csv`	ortholog pairs for a platform combination
`diffexpression_[nr]_[experiment].csv`	differential expression for comparison `nr` in `experiment`

Table 2: Additional files that can be exported with `exporttool.sh`.

1.2 Platform Descriptions

All files listed in Table 1 reference the platform using its short name, `[platform.short_caption]`. The latter can be matched with the long description of the platform and other parameters thanks to the information provided in the platform description file.

The file contains the following attributes:

Fieldname	Description
`platform.short_caption`	Short name of the platform
`platform.caption`	Long name of the platform
`organism.short_caption`	Short name of the organism of this platform
`technology.caption`	Name of the technology of this platform
`datatype.caption`	Name of a datatype supported in this platform

Table 3: Field values exported for platforms.

1.2.1 File Name

`platforms.csv`

1.2.2 File Format

The file format uses a CSV format as defined in [2] with two header rows and the delimiter comma (,) to separate the fields of a row.

1. Header row indicating the type of the export

   ```
   "@@@@@platforms"
   ```

2. Header row defining the fields of each data row

   ```
   "platform.short_caption","platform.caption","organism.short_caption","organism.caption",
   "technology.caption","datatype.captions"
   ```

3. N data rows, each containing

   ```
   "[platform.short_caption]","[platform.caption]","[organism.short_caption]",
   "[organism.caption]","[technology.caption]","[datatype.caption],[datatype.caption],..."
   ```

1.2.3 Example

The export of the platforms files would result in the following file content:

```
"@@@@@platforms"
"platform.short_caption","platform.caption","organism...","...","technology.caption","datatype.captions"
"AT_AFFY_AG","Affymetrix Arabidopsis Genome Array","AT","Arabidopsis thaliana","MicroArray","RMA,RMA_LOG"
"AT_AFFY_ATH1","Affymetrix Arabidopsis ATH1 Genome Array","AT","Arabidopsis...","MicroArray","RMA,RMA_LOG"
"AT_mRNASeq_TAIR10_GL","mRNA-Seq Gene Level...","AT","Arabidopsis...","NGS","EXPECTED_COUNT,TPM,TPM_LOG"
...
```

1.3 Experiment Descriptions

This section specifies the attributes exported for experiments of a given platform. The following experiment attributes will be exported:

Fieldname	Description
`experiment.nbr`	Name of the experiment (unique identifier, defined by curation)
`experiment.caption`	Title of the experiment
`repository.caption`	Name of the experiment repository
`experiment.link`	URL of the experiment in the repository (can be empty/unknown)
`experiment.experimenter`	Author(s) of the experiment

Table 4: Field values exported for experiments.

1.3.1 File Name

`experiments_[platform.short_caption].csv`

1.3.2 File Format

The file format uses a CVS format as defined in [2] with two header rows and the delimiter comma (,) to separate the fields of a row.

1. Header row indicating the type of the export

   ```
   "@@@@@experiments"
   ```

2. Header row defining the fields of each data row

   ```
   "experiment.nbr","experiment.caption","repository.caption","experiment.link","experiment.experimenter"
   ```

3. N data rows, each containing

   ```
   "[experiment.nbr]","[experiment.caption]","[repository.caption]","[experiment.link]",
   "[experiment.experimenter]"
   ```

1.3.3 Example

For example, exporting the experiment attributes of a *Mus musculus* array, `MM_AFFY_430_2`, results in the following file content (output shortened):

```
"@@@@@experiments"
"experiment.nbr","experiment.caption","repository.caption","experiment.link","experiment.experimenter"
"MM-00001","GSE1479: C57BL/6 Benchmark Set for Early...","GEO","http:...","Schinke M et al. / Izumo S"
"MM-00002","GSE1871: Lung samples treated with...","GEO","http:...","Jacobson J et al. / Garcia JG"
"MM-00004","GSE2372: Aortae of 32 weeks old apoE mice","GEO","http:...","Habenicht AJ"
"MM-00006","GSE2463: Targets of FGFR2-IIIb signalling in the hair follicle","GEO","http:...","Schlake T"
...
```

1.4 Experiment-Replica-Sample Relations

This section specifies the format of the exported relation between experiments, replicas, and samples in a given platform. The following attributes will be exported:

Fieldname	Description
`experiment.nbr`	Unique name of the experiment
`replica.caption`	Name of this replica according to internal curation rules
`chip.caption`	Original name of a sample as defined by the authors/repository
`chip.key`	Unique key identifying a sample: `[experiment.nbr];[chip.caption]`

Table 5: Field values exported for the experiment-replica-sample tree.

1.4.1 File Name

`experimenttree_[platform.short_caption].csv`

1.4.2 File Format

The file format represents a tree of the "contained-in" relation between an experiment, its replicas, and the chips belonging to each replica:

1. Header row indicating the type of the export:

   `"@@@@@experiment"`

2. One row (not indented) with the experiment name:

   `"[experiment.nbr]"`

3. One row, indented by one comma, containing the name of a replica belonging to the immediately preceding experiment:

   `",[replica1.caption]"`

4. N rows, indented by two commas, each identifying a sample belonging to the previously defined replica:

   `",,[chip.key],[chip.caption]"`

5. One row, indented by one comma, containing the next replica belonging to the same experiment:

   `",[replica2.caption]"`

...

6. One row (not indented) with the next experiment name:

   `"[experiment.nbr]"`

...

1.4.3 Example

The export of the relation experiment-replica-samples of an *Arabidopsis thaliana* array, `AT_AFFY_ATH1`, results in the file content (excerpt):

```
"@@@@@experiment"
"AT-00087"
,"Dev. base._wt_cot_7d"
,,"AT-00087;ATGE_1_A","ATGE_1_A"
,,"AT-00087;ATGE_1_B","ATGE_1_B"
,,"AT-00087;ATGE_1_C","ATGE_1_C"
,"Dev. base._wt_hyp_7d"
,,"AT-00087;ATGE_2_A","ATGE_2_A"
,,"AT-00087;ATGE_2_B","ATGE_2_B"
,,"AT-00087;ATGE_2_C","ATGE_2_C"
...
AT-00088"
,"Dev. base. 2_wt_inf_shoa_14d"
,,"AT-00088;ATGE_8_A","ATGE_8_A"
,,"AT-00088;ATGE_8_B","ATGE_8_B"
,,"AT-00088;ATGE_8_C","ATGE_8_C"
,"Dev. base. 2_wt_roo_17d"
,,"AT-00088;ATGE_9_A","ATGE_9_A"
,,"AT-00088;ATGE_9_B","ATGE_9_B"
,,"AT-00088;ATGE_9_C","ATGE_9_C"
...
```

1.5 Experiment Ontologies

This section specifies the exported experiment-level ontology tree and its relation to the annotated experiments. The entries for this tree represent the ontology categories. The three experiment ontologies *Research Area*, *Global Study Type*, and *Study Design* are currently available. The following attributes will be exported:

Fieldname	Description
`ontology.class`	The name of the ontology class. One among: `applicationarea`, `globalstudytype`, `studydesign`
`factor.caption`	The name of this ontology category
`factor.notes`	Notes/description for this category
`experiment.nbr`	Unique name of the experiment

Table 6: Field values exported for an experiment-level ontology tree.

1.5.1 File Name

```
applicationarea_[platform.short_caption].csv
globalstudytype_[platform.short_caption].csv
studydesign_[platform.short_caption].csv
```

1.5.2 File Format

The file format represents a tree of the relation between experiment ontology categories and the experiments:

1. Header row indicating the type of the export:

   ```
   "@@@@@[ontology.class]"
   ```

2. One row with the definition of a root category (and optional value for notes):

   `"[factor.caption]","[factor.notes]"`

3. One row with the definition of a child category of the preceding category (and optional value for notes). The number of commas by which it is indented represent the hierarchy level of this category (in this example, two below root level):

   `,,"[factor.caption]","[factor.notes]"`

4. N rows, each identifying an experiment annotated with the immediately preceding category. The `[experiment.nbr]` values are indented with an extra comma with respect to the hierarchy level of the annotated category:

   `,,,"[experiment.nbr]"`

5. And so on, until the complete tree structure is detailed ...

NOTE: Ontology categories that are not annotated to any experiment are never printed. Therefore the only type of leaf nodes in the exported tree are of type `[experiment.nbr]`. This is actually enough to distinguish between lines containing categories vs. experiment numbers.

1.5.3 Example

The export of the experiment ontology *Research Area* without notes results in the file (excerpt):

```
"@@@@@applicationarea"
"Cardiology and angiology"
,"Basic research (cardiology and angiology)"
,,"MM-00001"
,,"MM-00148"
...
,"Arrhythmia"
,,"Atrial fibrilation"
,,,"MM-00348"
,,,"MM-00427"
,,,"HS-00120"
,,"Tachycardia"
,,,"RN-00095"
...
```

1.6 Anatomy, Cell Type, Cell Line, Neoplasm, Development, Property

This section specifies the exported ontology tree and the annotated samples for a given platform and sample-level ontology. The following attributes of this relation will be exported:

Fieldname	Description
`ontology.class`	Name of the ontology class. One among: `anatomy`, `celltype`, `cellline`, `neoplasm`, `development`, `property`
`factor.caption`	Name of this category
`factor.notes`	If present, notes/description for this category
`experiment.nbr`	Unique name of an experiment
`chip.caption`	Original name of a sample as defined by the authors/repository
`chip.key`	Unique key identifying a sample: `[experiment.nbr];[chip.caption]`

Table 7: Field values exported for ontology-annotation trees.

1.6.1 File Names

`anatomytree_[platform.short_caption].csv`
`celltypetree_[platform.short_caption].csv`
`celllinetree_[platform.short_caption].csv`
`neoplasmtree_[platform.short_caption].csv`
`developmenttree_[platform.short_caption].csv`
`propertytree_[platform.short_caption].csv`

1.6.2 File Format

The file format represents a tree of the relation between ontology categories and the chips:

1. Header row indicating the type of the export:

   `"@@@@@[ontology.class]"`

2. One row with the definition of a root category in the ontology (and optional value for notes):

   `"[factor.caption]","[factor.notes]"`

3. One row with the definition of a child category of the preceding ontology category (and optional value for notes). The number of commas by which it is indented represent the hierarchy level of this category (in this example, two below root level):

   `,,"[factor.caption]","[factor.notes]"`

4. N rows, each identifying a sample which is annotated with the immediately preceding ontology category. The `[chip.key]` values are indented with an extra comma with respect to the hierarchy level of the annotated category:

   `,,,"[chip.key]"`

5. And so on, until the complete tree structure is detailed ...

NOTE: Ontology categories that are not annotated to any sample are never printed. Therefore the only type of leaf nodes in the exported tree are of type `[chip.key]`. This is actually enough to distinguish between lines containing categories vs. sample identifiers.

1.6.3 Example

The export of the anatomy-chip-relation of an *Arabidopsis thaliana* array, `AT_AFFY_ATH1`, results in the file (excerpt, without category notes):

```
"@@@@@anatomy"
"callus"
,"AT-00265;GSM227609"
,"AT-00265;GSM227610"
,"AT-00078;GSM142591"
,"AT-00078;GSM142592"
,"AT-00078;GSM142593"
...
"cell culture / primary cell"
,"AT-00005;T0_APH"
,"AT-00005;T2_APH"
,"AT-00005;T4_APH"
,"AT-00005;T6_APH"
,"AT-00005;T8_APH"
...
```

1.7 Stimulus and Mutation

This section specifies the exported relation between Stimulus (resp. Mutation), the treatment and control comparison sets and its chips for a given platform. The following attributes will be exported:

Fieldname	Description
`ontology.class`	Name of the ontology class: `stimulus` or `mutation`
`factor.caption`	Name of the stimulus/mutation category
`factor.notes`	Notes/description for this stimulus/mutation category
`experiment.nbr`	Unique name of an experiment
`chip.caption`	Original name of a sample as defined by the authors/repository
`chip.key`	Unique key identifying a sample: `[experiment.nbr];[chip.caption]`

Table 8: Field values exported for stimulus/mutation ontology tree.

1.7.1 File Names

`stimulustree_[platform.short_caption].csv`
`mutationtree_[platform.short_caption].csv`

1.7.2 File Format

The file format represents a tree of the relation between stimulus (mutation), its optional contained stimulus (mutation), the chips annotated to the stimulus (mutation) and the comparison set in which the stimulus (mutation) participates defined as a treatment and control group of chips:

1. Header row indicating the type of the export:

   ```
   "@@@@@[ontology.class]"
   ```

2. One row with the definition of the category (and optional value for notes) :

   ```
   "[factor.caption]","[factor.notes]"
   ```

3. N rows, each identifying a sample annotated with the previously defined category. These rows are indented by an additional comma with respect to the hierarchy level of the category.

   ```
   ,"[chip.key]"
   ```

4. One placeholder row indicating the start of a treatment and control group description. This row is indented by the a number of commas equivalent to the current hierarchy level.

   ```
   ,"COMPARISONSET"
   ```

5. One row with the comma-separated list of categories for the treatment group:

   ```
   ,,"[factor.caption]"
   ```

6. N rows, each identifying a sample belonging to the previously defined treatment group:

   ```
   ,,,"[chip.key]"
   ```

7. One row with the comma-separated list of categories for the control group:

   ```
   ,,"[factor.caption]"
   ```

8. N rows, each identifying a sample belonging to the previously defined control group:

   ,,,"[chip.key]"

9. One row describing a category contained in the previously defined category (and optional value for notes), if any. As for the other ontologies, these are indented by a number of commas equivalent to the current hierarchy level.

...

NOTE: Each COMPARISONSET entry with the same treatment and control group is printed only once in order to avoid redundancy. During the depth-first traversal of the tree, the COMPARISONSET entry is written to the first category participating in the control or treatment group. When the tree traversal encounters another stimulus category belonging to the same COMPARISONSET entry the output is suppressed.

```
"@@@@@[ontology.class]"
"factor1.caption","factor1.notes"
,"chip1.key"
,"chip2.key"
...
,"chipN.key"
,"COMPARISONSET"
,,"factor2.caption" of treatment replica set
,,,"chip1.key"
,,,"chip2.key"
...
,,,"chipN.key"
,,"factor3.caption" of control replica set
,,,"chip1.key"
,,,"chip2.key"
...
,,,"chipN.key"
...
"factor6.caption","factor6.notes"
,"chip1.key"
,"chip2.key"
...
,"chipN.key"
,"factor7.caption","factor7.notes"
,,"chip1.key"
,,"chip2.key"
...
,,"chipN.key"
,,"COMPARISONSET"
,,,"factor8.caption" of treatment replica set
,,,,"chip1.key"
,,,,"chip2.key"
...
,,,,"chipN.key"
,,,"factor9.caption" of control replica set
,,,,"chip1.key"
,,,,"chip2.key"
...
,,,,"chipN.key"
```

1.7.3 Example

The export of the stimulus-chip-relation of an *Arabidopsis thaliana* array, `AT_AFFY_ATH1` results in the file (excerpt without notes):

```
"@@@@@stimulus"
"Biotic"
,"B. cinerea"
,,"AT-00147;GSM133025"
,,"AT-00147;GSM133026"
,,"AT-00147;GSM133027"
,,"AT-00147;GSM133028"
,,"AT-00147;GSM133029"
,,"AT-00147;GSM133030"
,,"COMPARISONSET"
,,,"B. cinerea"
,,,,"AT-00147;GSM133028"
,,,,"AT-00147;GSM133029"
,,,,"AT-00147;GSM133030"
,,,,"AT-00147;GSM133025"
,,,,"AT-00147;GSM133026"
,,,,"AT-00147;GSM133027"
,,,"non-infected rosette leaf samples"
,,,,"AT-00147;GSM133034"
,,,,"AT-00147;GSM133035"
,,,,"AT-00147;GSM133036"
,,,,"AT-00147;GSM133031"
,,,,"AT-00147;GSM133032"
,,,,"AT-00147;GSM133033"
,"P. syringae pv. tomato study 4 (DC3000 avrRpm1)"
,,"AT-00238;GSM157375"
,,"AT-00238;GSM157378"
,,"AT-00238;GSM157381"
,,"COMPARISONSET"
,,,"P. syringae pv. tomato study 4 (DC3000 avrRpm1)"
,,,,"AT-00238;GSM157375"
,,,,"AT-00238;GSM157378"
,,,,"AT-00238;GSM157381"
,,,"P. syringae pv. tomato study 4 (DC3000)"
,,,,"AT-00238;GSM157373"
,,,,"AT-00238;GSM157376"
,,,,"AT-00238;GSM157379"
,"P. syringae pv. tomato study 4 (DC3000 hrpA-)"
,,"AT-00238;GSM157374"
,,"AT-00238;GSM157377"
,,"AT-00238;GSM157380"
...
```

1.8 Expression Matrix

This section specifies the files of exported expression values (RMA-normalized signal values for MicroArrays, TPM or expected counts for mRNA-Seq) for a given platform. The following fields will be exported:

Fieldname	Description
`datatype.caption`	The requested type of data. Depending on the technology, can be one of: `RMA`, `RMA_LOG`, `TPM`, `TPM_LOG`, `EXPECTED_COUNT`
`measure.caption`	Unique name identifying a measure within a platform
`experiment.nbr`	Unique name of an experiment
`chip.caption`	Original name of a sample as defined by the authors/repository
`chip.key`	Unique key identifying a sample: `[experiment.nbr];[chip.caption]`

Table 9: Field values exported for expression matrix.

1.8.1 File Name

`expressionmatrix_[platform.short_caption].csv`

1.8.2 File Format

The file format is a matrix containing the sample identifiers in the first row and the measure captions in the first column. The upper left corner contains the datatype of the expression values. The comma is used as a delimiter. Each expression value is formatted according to the Java method `Float.toString(float f)`, see [1].

```
[datatype.caption],[chip1.key],[chip2.key],[chip3.key], ...
[measure1.caption],#.#########,#.#########,#.#########, ...
[measure2.caption],#.#########,#.#########,#.#########, ...
[measure3.caption],#.#########,#.#########,#.#########, ...
...
```

1.8.3 Example

The export of the gene-level expression values of the mRNA-Seq platform for *Arabidopsis thaliana*, `AT_mRNASeq_ARABI_GL`, results in the file (excerpt):

```
"TPM","AT-00699;SRR515335","AT-00699;SRR515336","AT-00699;SRR515337","AT-00699;SRR515338",...
"AT1G01010","6.831336","5.1100297","4.6547737","5.411358",...
"AT1G01020","19.085173","18.716497","18.911911","17.526339",...
"AT1G01030","6.436051","6.9451795","5.311087","6.461323",...
"AT1G01040","12.0511265","11.710485","11.278486","12.226034",...
...
```

1.9 Ortholog Pairs

This section specifies the exported ortholog information. The following attributes will be exported:

Fieldname	Description
organism.short_caption	Two-letter code of the organism
gene.caption	Name of the gene
alias.caption	Name of the alias(es) of the gene in different naming schemes
platform.short_caption	Short name of the platform of the measure
measure.caption	Unique name identifying a measure within a platform
numTissues	Number of tissues
sequenceScore	Sequence score (currently PAM)
expressionScore	Expression score based on tissues and measure of the specifed gene

Table 10: Field values exported in the ortholog file.

1.9.1 File Name

orthologs.csv

1.9.2 File Format

The file format is a CSV file containing on each row a pair of orthologus genes from two different organisms, including the number of tissues and the measure used to compute the expression score, as well as the sequence score (PAM).

1. Row. Comment line indicating the type of the export:

```
 # organism1,gene1,alias1,platform1,measure1,numTissues1,
organism2,gene2,alias2,platform2,measure2,numTissues2,
sequenceScore,expressionScore
```

N. N rows. Definition of the orthologous gene pair :

```
 "[organism1.short_caption.caption]","[gene1.caption]","[alias1.caption]",
"[platform1.short_caption]","[measure1.caption]","[numTissues1]",
"[organism2.short_caption.caption]","[gene2.caption]","[alias2.caption]",
"[platform2.short_caption]","[measure2.caption]","[numTissues2]",
"[sequenceScore]","[expressionScore]"
```

1.9.3 Example

The export of orthologous genes between *Arabidopsis thaliana* (AT) and *Glycine max* (GM) results in the following output (excerpt):

```
# organism1,gene1,alias1,platform1,measure1,numTissues1,organism2,gene2,alias2,platform2,...
AT,AT1G01050,PPA1,AT_AFFY_ATH1,261579_at,105,GM,Glyma.07G048300,Glyma.07G048300,GM_AFFY_...
AT,AT1G01060,LHY,AT_AFFY_ATH1,261569_at,105,GM,Glyma.07G048500,Glyma.07G048500,GM_AFFY_...
AT,AT1G01060,LHY,AT_AFFY_ATH1,261569_at,105,GM,Glyma.16G017400,Glyma.16G017400,GM_AFFY_...
AT,AT1G01080,,AT_AFFY_ATH1,261577_at,105,GM,Glyma.07G049500,Glyma.07G049500,GM_AFFY_...
AT,AT1G01100,RPP1A,AT_AFFY_ATH1,261578_at,105,GM,Glyma.04G229000,Glyma.04G229000,GM_AFFY_...
AT,AT1G01110,IQD18,AT_AFFY_ATH1,261580_at,105,GM,Glyma.16G019700,Glyma.16G019700,GM_AFFY_...
AT,AT1G01120,KCS1,AT_AFFY_ATH1,261570_at,105,GM,Glyma.03G260300,Glyma.03G260300,GM_AFFY_...
```

1.10 Differential Expression

This section specifies the file format used to export differentially expressed genes given a comparison. The following attributes are used:

Fieldname	Description
`experiment.nbr`	Unique key identifying the experiment in the database
`comparison.nr`	Index of the comparison within the experiment
`treatment.caption`	Caption of the samples in the treatment group of a comparison
`chip.name`	Annotation-based name of the sample
`control.caption`	Caption of the samples in the control group of a comparison
`gene.caption`	Name of the gene
`measure.caption`	Name of the measure
`namingscheme.caption`	Name of the selected naming scheme (optional)
`alias.caption`	Name of the gene alias in the selected naming scheme (optional)
`alias.description`	Description of the gene alias (if present)

Table 11: Field values referenced in a differential expression export.

1.10.1 File Name

`diffexpression_[comparison.nr]_[experiment.nbr].csv`

1.10.2 File Format

The file format is a CSV file containing a few rows of header information and then a list of differentially expressed genes ordered by increasing p-value.

1. One comment line indicating the experiment:

   ```
   # experiment number: [experiment.nbr]
   ```

2. One comment line defining the comparison:

   ```
   # comparison name: [treatment.caption] / [control.caption]
   ```

3. N comment lines indicating all samples in the treatment group (one per line):

   ```
   # treatment: [chip.name]
   ```

4. M comment lines indicating all samples in the control group (one per line):

   ```
   # control: [chip.name]
   ```

5. Header row defining the exported data for differentially expressed genes (the `[namingscheme.caption]` column is optional, see command description and options):

   ```
   "treatmentMeanExpression","controlMeanExpression","Log-ratio","p-value","minFDR",
    "[namingscheme.caption]","Gene","Measure","Description"
   ```

6. P rows with the exported data values (one gene per row). The `"[alias.caption]"` column is optional and the `[alias.description]` may be empty. Floating point values are formatted according to the Java method `Float.toString(float f)`, see [1]:

   ```
   "#.#######","#.#######","#.#######","#.#######","#.#######","[alias.caption]",
    "[gene.caption]","[measure.caption]","[alias.description]"
   ```

1.10.3 Example

Exporting the differentially expressed genes for a comparison in a human experiment may produce an output like this (excerpt):

```
# experiment number: HS-00739
# comparison name: prostate cancer study 8 (tissue) / adjacent prostate tissue
# treatment: prim_pros neo (mal)_m_4-1
# treatment: prim_pros neo (mal)_m_4-2
# treatment: prim_pros neo (mal)_m_4-3
# treatment: prim_pros neo (mal)_m_4-4
# treatment: prim_pros neo (mal)_m_4-5
# control: con_prost_m_1-1
# control: con_prost_m_1-2
# control: con_prost_m_1-3
# control: con_prost_m_1-4
# control: con_prost_m_1-5
"treatmentMeanExpression","controlMeanExpression","Log-ratio","p-value","minFDR","Gene",...
13.134493,7.85658740,5.2779064,4.55793837E-9,1.0685174E-4,"ENSG00000144355","242138_at",""
11.500292,8.17097949,3.3293132,1.98972574E-8,2.3322570E-4,"ENSG00000115844","207147_at",""
11.956574,13.9206571,-1.9640827,2.8679061E-7,0.0022410789,"ENSG00000135929","203979_at",""
...
```

2 experimentexport.sh

The `experimentexport.sh` script exports data in tabular format. The four available commands (`annotation`, `comparison`, `signal`, `geoseries`) and their options are described in the *Genevestigator Export Tools* document. In the following four sections we will define the respective output formats.

2.1 Annotation

This section specifies the annotation file format. The following attributes are used:

Fieldname	Description
`experiment.nbr`	Unique key identifying the experiment in the database
`experiment.caption`	Title of the experiment
`experiment.experimenter`	Author(s) of the experiment
`experiment.link`	URL of the experiment in the original repository
`repository.caption`	Name of the experiment repository
`chip.caption`	Unique key identifying the chip within the experiment
`replica.caption`	Annotation-based name of the replica
`ontology.class`	The name of an ontology class (see previous sections)
`factor.caption`	The name of an ontology category
`factor.path`	List of categories (separated by "`>`") from node to root
`factor.notes`	Notes/description for the category

Table 12: Field values of an annotation file.

2.1.1 File Format

This tabular annotation export consists of a CSV file containing the following:

1. Header row indicating the content of each column. The leftmost eight columns are fixed, while the remaining depend on the actual experiment and sample annotations.

   ```
   "Chip","Sample Name","Replica Caption","Experiment Number","Experiment Caption","Experimenter",
   "Experiment Link","Experiment Repository", "[ontology.class]","[ontology.class] - path from root",
   "[ontology.class] - notes","[ontology.class]",...
   ```

2. N rows, one per sample, each containing sample- and experiment-level data plus the annotations from the ontologies listed in the header. If a certain ontology is not used for annotation of a specific sample, the corresponding column field is empty.

   ```
   [chip.caption],[chip.name],[replica.caption],[experiment.nbr],[experiment.caption],
   [experiment.experimenter],[experiment.link],[repository.caption],[factor.caption],[factor.path],
   [factor.notes],[factor.caption],...
   ```

2.1.2 Example

Exporting the annotation for an experiment on *Arabidopsis thaliana* results in the following output (excerpt):

```
"Chip",...,"Experiment Repository","ANATOMY","ANATOMY - path from root","ANATOMY - notes","MUTATION",...
"ATGE_1_A",...,"AtGenExpress","cotyledon","cotyledon > seedling",,"Col-0","Col-0 > Col","Arabidopsis...
"ATGE_1_B",...,"AtGenExpress","cotyledon","cotyledon > seedling",,"Col-0","Col-0 > Col","Arabidopsis...
...
"ATGE_2_A",...,"AtGenExpress","hypocotyl","hypocotyl > seedling",,"Col-0","Col-0 > Col","Arabidopsis...
...
```

2.2 Comparison

This section specifies the comparison file format. The following attributes are used:

Fieldname	Description
`experiment.nbr`	A unique key identifying the experiment in the database
`chip.name`	Annotation-based name of the sample
`chip.caption`	Original sample name/unique identifier within experiment
`treatment.caption`	Caption of the treatment group of a comparison
`control.caption`	Caption of the control group of a comparison
`(t,c,)`	Sample membership within a comparison (column): `t` = treatment, `c` = control, (empty) = does not belong to comparison

Table 13: Field values of a comparison file.

2.2.1 File Format

The file format is a CSV file containing for an experiment on each row the sample at its membership in a comparison of this experiment.

1. Header row defining the comparison name(s):

```
"experiment number","chip name","chip caption org","[treatment1.caption] / [control1.caption]"
  ,"[treatment2.caption] / [control2.caption]",...
```

2. N rows with the definition of the membership of a sample to the comparison(s) listed in the header:

```
"[experiment.nbr]","[chip.name]","[chip.caption]","(t|c|)","(t|c|)",...
```

NOTE: Any field containing the separator character "`,`" (comma) must be enclosed in double quotes.

2.2.2 Example

The export of all comparisons of a human experiment results in a file (excerpt):

```
"experiment number","chip name","chip caption org","prostate cancer study 8 (p. canc) / ...
HS-00739,Prostate cancer 9_stro._pro_CD90+ ptasc_m#2_rep_1,GSM447178,c,,,t
HS-00739,Prostate cancer 9_stro._pro_CD90+ ptasc_m#1_rep_1,GSM447176,c,,,t
HS-00739,Prostate cancer 9_stro._bla_CD13+ btasc_m_rep_1,GSM447174,,,,
HS-00739,Prostate cancer 9_prim._pro_p. can_m#4_rep_1,GSM447169,,t,,
...
```

2.3 Signal

Export the expression data of selected experiment(s) for a data type of the corresponding platform (either RMA or TPM). Optionally, export the name of gene/transcript aliases mapped to a measure in a chosen naming scheme (e.g., *Ensembl Gene*, *Gene Symbol*, ...)

2.3.1 File Format

The file format is a CSV file containing the following:

1. Header row indicating the content of each column (the third column is optional):

```
"datatype.caption","Gene|Transcript|Protein|Probeset","[namingscheme.caption]","[chip.key]","[c
```

Fieldname	Description
measure.caption	Unique name identifying the measure in the platform
ID	ID of the genetic element of the reference genome/transcriptome/proteome
experiment.nbr	Unique name of an experiment
chip.caption	Original name of a sample as defined by the authors/repository
chip.key	Unique key identifying a sample: [experiment.nbr] /// [chip.caption]
namingscheme.caption	Name of the selected naming scheme (optional)
alias.caption	Name of a gene alias (if a naming scheme is selected)

Table 14: Field values of a signal file.

2. N rows, one per measure, each containing the expression values for the samples indicated by the header. The upper left corner contains the datatype of the expression values. The second column is the ID of the reference genome/transcriptome/proteome to which the measure is mapped, if the measure maps to multiple IDs, the IDs are concatenated with ///. May be empty if no mapping exists. The third column is present only if a naming scheme was selected and may be empty, if no mapping exists for the current ID and the selected naming scheme, or may contain multiple aliases concatenated with /// if there are multiple mappings to the ID.

```
[measure1.caption],[ID1],[alias1.caption],#.######,#.######,#.######, ...
[measure2.caption],[ID2],[alias2.caption],#.######,#.######,#.######, ...
[measure3.caption],[ID3],[alias3.caption],#.######,#.######,#.######, ...
...
```

2.3.2 Example

The export of the gene-level expression values of an experiment in the *Arabidopsis thaliana* array, showing the ID of the reference genome and additionally the mappings for the *Gene Symbol* naming scheme results in the file (excerpt):

```
Measure,Gene,Gene Symbol,AT-00087 /// ATGE_1_A,AT-00087 /// ATGE_1_B,AT-00087 /// ATGE_1_C,...
244901_at,ATMG00640,ORF25,691.6898,726.6299,540.2,...
244902_at,ATMG00650,NAD4L,683.52014,859.03973,812.8398,...
...
244908_at,ATMG00720,ORF107D,249.25996,270.03,241.41998,...
244909_at,ATMG00740,ORF100A,447.9,451.77005,455.9401,...
244910_s_at,AT2G07686///ATMG00750,ORF119,180.18999,165.12999,147.95998,...
...
```

2.4 Geo Matrix Series format

This format combines in a single file both the annotation and the expression data for an experiment. We refer to the offical documentation on GEO for a description of this format:

```
https://www.ncbi.nlm.nih.gov/geo/info/overview.html
https://www.ncbi.nlm.nih.gov/geo/info/soft.html
```

NOTE: At variance with the annotation export (see Section 2.1), this type of export format can accommodate *one, and only one* experiment per file.

References

[1] S. Microsystems. Java float.tostring(float f), 2008. http://www.j2ee.me/javase/6/docs/api/java/lang/Float.html.

[2] Y. Shafranovich. Common format and mime type for comma-separated values (csv) files, 2005. http://www.rfc-editor.org/rfc/rfc4180.txt.