# The BeanSprout Software Family

Hin-Tak Leung

January 20, 2012

## 1 Introduction

- Bonsai and CarrotCNV are genotyping report and CNV analysis plug-in's for GenomeStudio, to export data from GenomeStudio to snpMatrix. They can write Rdata files without using R, among other capabilities.
- BeanSprout manipulates GenomeStudio project files without GenomeStudio. It doesn't do much more than GenomeStudio, other than that it runs on Linux and other non-windows systems. (although GenomeStudio itself works well enough on Linux now...)
- SpringOnion and CauliFlower was written to perform genotyping calling with GenTrain1/2 and CNV analysis with cnvPartition and PennCNV/QuantiSNP, when the source data is no longer available in GenomeStudio project form, or to handle substantially larger amount (10x at least) of source data than GenomeStudio can handle. They operates on textual exports, and in principle, can work on non-Illumina data.

Bonsai and CarrotCNV, by their plug-in nature, runs wherever Illumina GenomeStudio runs. The non-native build of BeanSprout, SpringOnion, CauliFlower runs on platforms where Novell Mono runs, which includes Linux, Mac OS X, Solaris, *BSD, as well as Microsoft windows. The native standalone executables require GCC as well as GNU Binutils (specifically the GNU assembler) to build, so are limited to 3 platforms: Linux, Mac OS X and Microsoft Windows (with Mingw and not MS Visual studio, and specifically excluding Solaris, since GCC on Solaris uses the SUN assembler, AFAIK).

### 1.1 Building and Installation

In mid 2010, Bonsai and CarrotCNV became windows installers. The installers can auto-detect where GenomeStudio is and put them in the right place; so there is no longer a need for a separate manual. There is only one user option, the directory to deposit outputs and that's pretty much self-explanatory - see screen-shots in the Appendix.

Also in mid 2010, BeanSprout, SpringOnion and CauliFlower became build-able as standalone native executables (primarily for Linux). So they pretty much *just work* like any other command-line tool on Linux. The standalone executables are maybe 10% slower than non-native ones (probably due to AOT/JIT performance). They only have the usual Linux system library dependencies and does not requires DLL's from GenomeStudio; However, BeanSprout and CauliFlower take parameters for cnvPartition in the form of an accompanying config file `BeanSprout.config` or `CauliFlowr.config` . See the next section.

## 2 Copy variant discovery with cnvPartition

The copy variant discovery functionality in both BeanSprout and CauliFlower are controlled by a config file:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
   <appSettings>
      <add key="IncludeSexChromosomes" value="true"/>
      <add key="IncludeMitochondria" value="false"/>
      <add key="AdjustYLRR" value="false"/>
      <add key="GcWaveAdjustLRR" value="false"/>
      <add key="GcWaveAdjustmentWindow" value="115000"/>
```

```
      <add key="SmoothLRR" value="false"/>
      <add key="SmoothingMovingAveragePeriod" value="2"/>
      <add key="AveragePloidyAdjustLRR" value="false"/>
      <add key="AveragePloidyAdjustLRRThreshold" value="2.5"/>
      <add key="ConfidenceThreshold" value="35"/>
      <add key="GapSizeThreshold" value="1000000"/>
      <add key="MinProbeCount" value="3"/>
      <add key="LogDiagnosticInfo" value="false"/>
      <add key="DetectExtendedHomozygosity" value="true"/>
      <add key="CopyNeutralLOHOnly" value="true"/>
      <add key="ChiSquareThreshold" value="23.5"/>
      <add key="MinHomozygoteCount" value="50"/>
      <add key="MinHomozygousRegionSize" value="10000000"/>
   </appSettings>
</configuration>
```

Due to some programming idiosyncrasy of Illumina staff, The config file is looked up from the start-up directory of the command, and also relative to and dependent on the existence of the file of executable/dll being there. This sounds a bit complicated, but what it means is that:

- The only valid way to run BeanSprout and CauliFlower is in the form of `./command` and not via `$PATH` or full path:

```
$ ./BeanSprout ...
...
$ ./CauliFlower ...
```

- For native builds, these files must exist relative to the current directory:

```
./BeanSprout
./BeanSprout.config
```

```
./CauliFlower
./CauliFlower.config
```

  or, for non-native builds,

```
./cnvPartition.dll
./CNVAlgorithm/cnvPartition/cnvPartition.dll
./CNVAlgorithm/cnvPartition/cnvPartition.dll.config
```

# 3 Usage

## 3.1 BeanSprout

Running with `--help` or `-h` shows the help message.

```
BeanSprout 5.3 by Hin-Tak Leung
Found Genotyping Module version: 1.9.4.24613
Usage: BeanSprout -in <project_dir> -out <out_dir>

Mandatory options:
       --in[-in][-i]    <project_dir>
       --out[-out][-o]  <out_dir>

Optional:
       -v2:              select GenTrain Version 2 (default 1)
```

```
       --cutoff[-c]:     <GenCall cutoff> (default 0.15)
       --precision[-p]:  output precision of Xnorm/Ynorm (default 3)

To disable some outputs:
       --skip-samples
       --skip-snps
       --skip-raws
       --skip-norms
       --skip-cnv

Exclusive options: (do this and nothing else)
       --egt[-egt][-e] <egtfile>

       --help[-h]           this message

Devel options:
       --write-new[-wn][-w] write new versions of *.bin files
       --write-snp-map      write a SNP_Map.txt file
       --chrom-egt          write chromomsome-separated egt files
```

Note not all the columns in the sample table are correct or populated — some of them are generated on-the-fly. A portion of the sample table, and almost the entirety of the genotype table, is generated on-the-fly when BeadStudio starts up. AFAIK, all the columns in the SNP stable are populated correctly.

A successful run looks like this:

```
$ mono BeanSprout.exe  -in ~/MyProject\ 100Samples\ 20080605/ -o out
BeanSprout 3.0 by Hin-Tak Leung, built against GT 3.2.33.27727
Found Genotyping Module version: 3.2.33.27727
GenTrain module: Call Version=6.3.0, Cluster Version = 6.3.1
Num of Samples: 100 with 45 columns
Skipping column 45, "HumanHap550v3_A", containing sub-columns in Sample Table
Number of SNPs: 561466 with 34 columns
Skipping column 34, "HumanHap550v3_A.bpm" containing sub-columns in SNP Table
```

Where the `out` directory contains three files `SNP_Table.txt`, `Sample_Table.txt` and `Raw_Table.txt`. The `SNP_Table.txt` file should be about 160MB for the 550K chips. The other two vary with the size of the project. The `out` directory also contains 20+ files with names `norm.chrom.*.gz` for the normalized X/Y values, in the same format as Bonsai Report plug-in, CelQuantileNorm, and readable by snpMatrix and Chiamo.

Optionally, the file `new.sd.bin` and `new.ld.bin` are written. They should be slightly larger than and no smaller than the original `sd.bin` anf `ld.bin` in the project `Data` directory and a compatible newer replacement of it, with some extra fields when loaded.

**Note:** Beansprout uses `Sample Name` in Carrot files but `Sample ID` in cnvPartition result. When the user names the two inconsistently, one might get inconsistencies.

**Note:** Illumina still continuously updates the project file format, so the last build of BeanSprout is useless as soon as a new version of GenomeStudio is out. Bonsai, CarrotCNV, SpringOnion and CauliFlower are not affected by the same problem, since none of them read project files directly.

## 3.2 SpringOnion

Using the -h option gives a usage summary:

```
SpringOnion 1.2 by Hin-Tak Leung
Found Genotyping Module version: 1.9.4.24613
Usage: SpringOnion -i <signalfile> -o <Robjectname> -l <logfile> -e <egtfile>

Mandatory options:
       -i    <gzed signalfile>
       -o    <R_object_name>
```

```
       -l    <log file>
       -e    <egt file>

Optional:
       -v2:    select GenTrain Version 2 (default 1)
       -c:    <GenCall cutoff> (default 0.15)
```

The input file is in the WTCCC signal format (same as Chiamo), and *must be* gz'ed. (plain uncompressed text files do not work).

Genotypes is outputted as R Data, with name "Genotypes" + "object name" + .Rda extension in the current directory. The log file is plain text, and the egt file is, well, Illumina's Cluster Definition file format.

**Note:** There is a small incompatibility with GenomeStudio regarding the EGT files: WTCCC signal format files written by Bonsai populate the vendor ID first column by the Illumina ID (e.g. `rs10013819-119_B_R_IFB1151893759:0`) of the SNP. This is not quite the rs number: it is rs number plus a few other pieces of information like forwards/backward strain and genome build, etc. SpringOnion copies those to the EGT files, but EGT files written by GenomeStudio uses rs numbers to index the clusters rather than the Illumina ID. This incompatbility is strictly-speaking "better" so will not be fixed. CauliFlower can read either form of the EGT files (and has the `--skip-longnameworkaround` option for this reason).

### 3.3 CauliFlower

```
CauliFlower 2.6 by Hin-Tak Leung
Found Genotyping Module version: 1.9.4.24613
Usage (read old result): CauliFlower -r CNVresult
    (old result sorted): CauliFlower -s CNVresult
       (read meta-info): CauliFlower -m CNVresult

Usage (new analysis):  CauliFlower -i <signalfile> -l <ld.bin> -e <egt-file> -o <outdir>

Mandatory options:
       -i/--in/-in    <gzed signalfile>
       -l/--ld.bin    <ld.bin>
       -e/--egt/-egt  <egt file>
       -o/--out/-out  <out dir>

Optional:
       --version:     output version info
       -v2:           select GenTrain Version 2 (default 1)
       -c/--chrom     <chromosome>
       --noCarrot    do not write Carrot files
       -w             <cachewindow> (default 500)

Devel:
       --skip-longnameworkaround
```

The 3 mandatory inputs are:

**signal file:** gz'ed, the usual WTCCC signal format

**ld.bin:** one of Data/ld.bin files from a relevant GenomeStudio project - it does not need to bear any relationship with the source data, just need to be of the same product type. (550k v3, versus 330k etc). This is used for information like GC content, seq and chromosome position of SNPs, etc.

**egt file:** a cluster definition file. Preferably of SpringOnion's output against the signal file.

In "read old result" mode, it can read its own output (obviously), but also those from any other GenomeStudio's CNV analysis plug-in's. (tyically found as `CNV/<algorithm>*.bin` inside a GenomeStudio project directory)

The optional inputs:

**chromosome:** mainly the difference is just autosome and sex chromosome/mitochondria; for autosome it is just used for file naming.

**cache window:** large for faster file access (also larger memory consumption). Input values higher than sample count has no effect. Adjust value for very large number of samples or low-spec machine.

If a 2nd sample is found to have the sample name as an earlier one, its name is append with ".1" before writing out to the CNV discovery output, as well as the Carrot output. **This behavior is intentional. This behavior is also incompatible with cnvParition under BeanSprout/GenomeStudio and CarrotCNV under GenomeStudio: cnvPartition under BeanSprout or GenomeStudio lets the CNV discovery result of a 2nd/later entry overwrite the 1st entry of the same sample name. CarrotCNV under GenomeStudio write duplicate sample names without modification. Duplicate entries inside Carrot are distinguishable by order, so most usage of CarrotCNV should try to read the 2nd of a duplicate entry to keep in sync with GenomeStudio's CNV discovery behavior.**

**The behavior of 2nd entry overwriting earlier entry of the same name is generic to all CNV discovery algorithm running under GenomeStudio.**

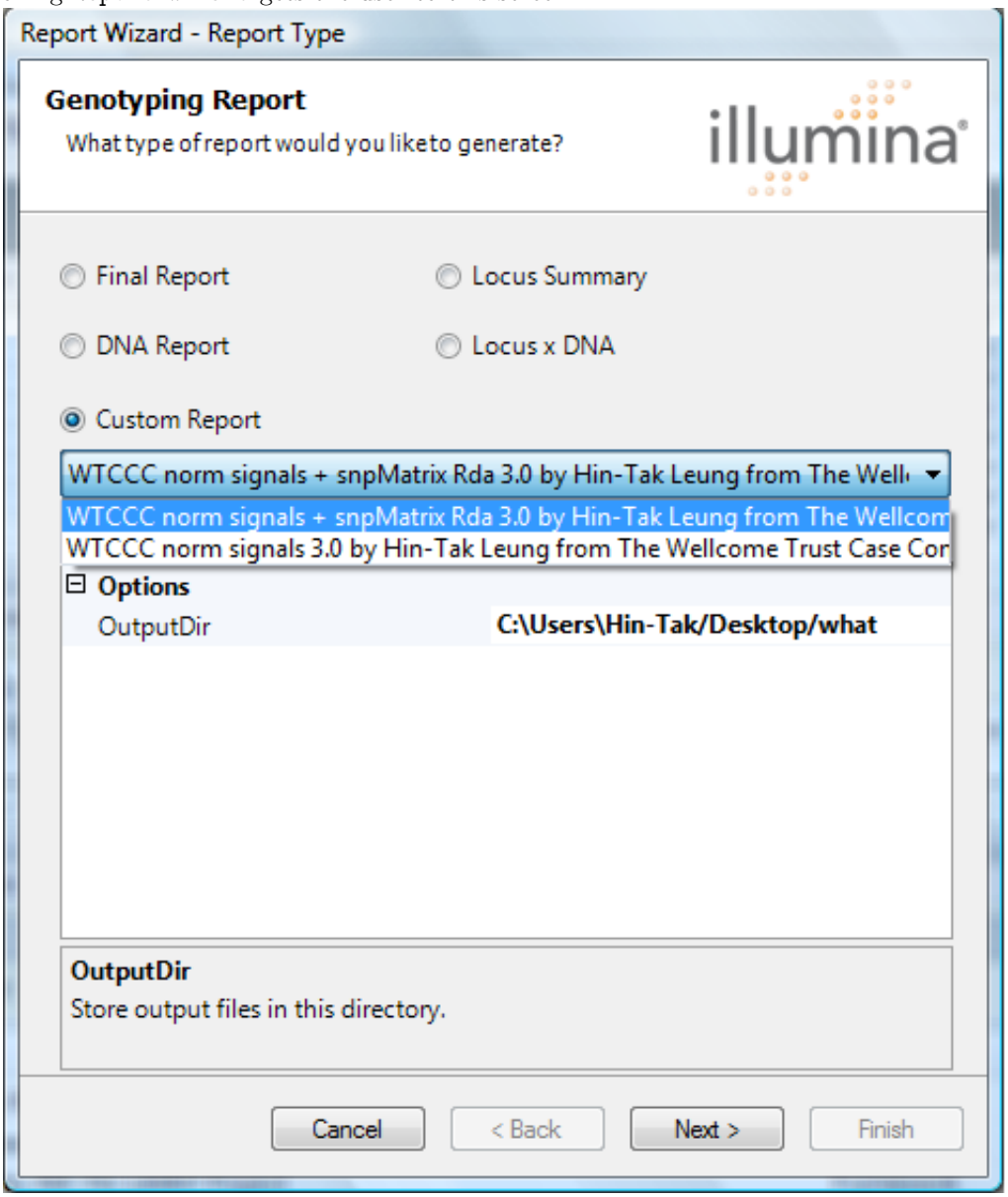
The `--skip-longnameworkaround` option is to disable `SpringOnion` compatibility code. See the SpringOnion section on EGT files.

**Note: Sex chromosome and Mitochondria analysis has not yet been worked on.**

# A Screenshots

## A.1 Bonsai

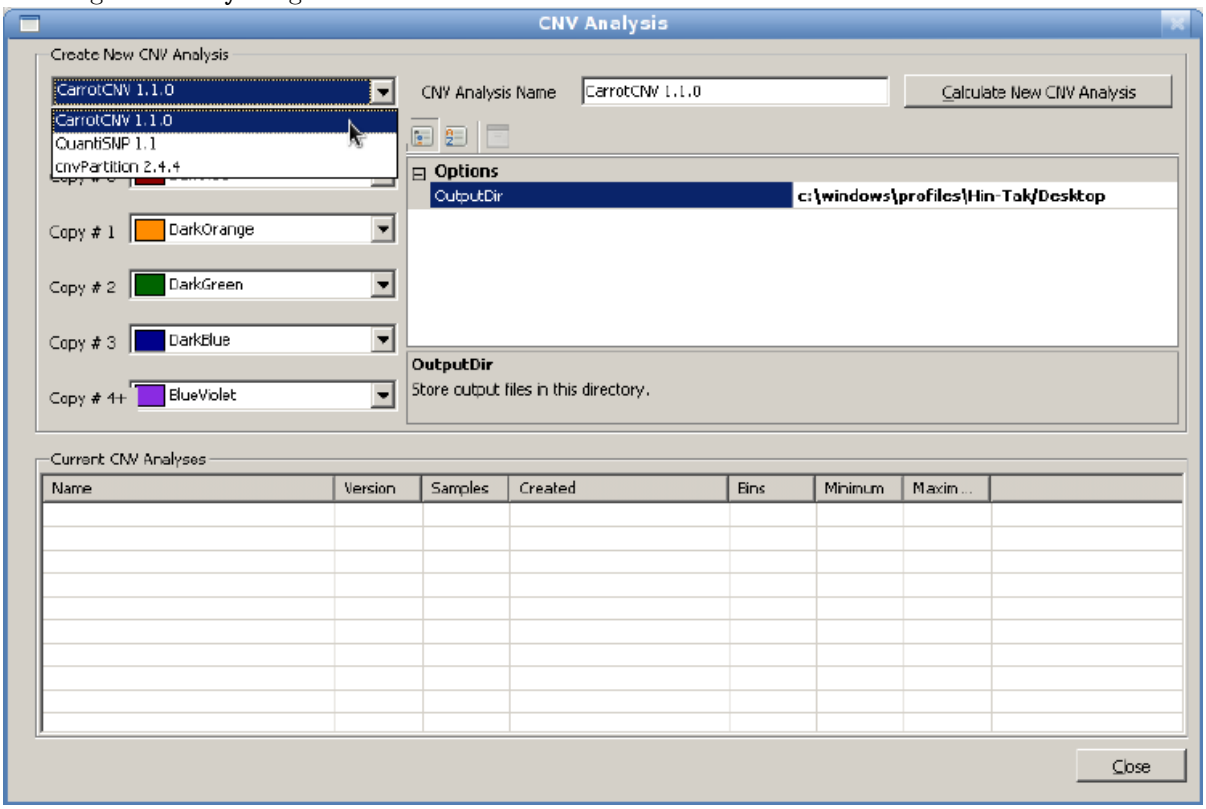
Invoking `Report Wizard` gets the user to this screen:



## A.2 CarrotCNV

**See also the sections of BeanSprout and CauliFlower for incompatibilities.** One source is inconsistency from users: `Sample ID` and `Sample Name`. CarrotCNV uses `Sample ID` while Beansprout uses `Sample Name`; CauliFlower may use either depends on how the WTCCC signal files are written.

Invoking `CNV Analysis` gets the user to this screen:




# B Differences between native and non-native builds

Non-native builds requires GenomeStudio DLLs, as well as these files located relative to the executables:

```
Modules/BSGT/dat.bin
cnvPartition.dll
CNVAlgorithm/cnvPartition/cnvPartition.dll
CNVAlgorithm/cnvPartition/cnvPartition.dll.config
```

Also Non-native builds requires setting the environment variable `MONO_IOMAP=case` to work around some file system case-sensitivity inconsistencies within the Genotyping module.

Note Case-sensitivity on unix platforms means some of the DLL files need to be renamed from *.DLL to *.dll (as above).

# C Minimal requirements for Running SpringOnion and CauliFlowser on Windows XP

The information in this section is correct as of May 2011 against the GenomeStudio 2010.3 infrastructure (i.e. just before 2011.1 comes out).

On windows XP (Professional, version 2002, Service Pack 3):

```
$version
Microsoft Windows XP [Version 5.1.2600]
```

These are necessary (i.e. the minimal) and sufficient for running SpringOnion:

```
./BeadStudioCommon.dll
./ClusterAlgInterface.dll
./GenotypingBSM.dll
./ILCA.dll
./Modules/BSGT/dat.bin
```

These are sufficient to run CauliFlower (some of these may not be needed):

```
./BeadStudioCommon.dll
./ClusterAlgInterface.dll
./CNVAlgorithm/cnvPartition/cnvPartition.dll
./CNVAlgorithm/cnvPartition/cnvPartition.dll.config
./cnvPartition.dll
./ColumnPlugin.dll
./FrameworkInterfaceLibrary.DLL
./GenotypingBSM.dll
./IGVInterface.dll
./ILCA.dll
./Table.dll
./UIUtils.DLL
```

**Note:** Also note that CauliFlower requires explicitly an `ld.bin` for input, but next to the `ld.bin` there is an implicit input requirement of an accompanying `seqdata.bin`. This is normally satisfied in the GenomeStudio project layout.

# D The most common problems

## D.1 cnvPartition config

If you see this error message below, you have read the section above on the only valid way to run BeanSprout and CauliFlower is in the form of `./command` , right?

```
CauliFlower 2.0 by Hin-Tak Leung
Found Genotyping Module version: 1.7.4.25862
GenTrain Version 2 selected

Unhandled Exception: System.Reflection.TargetInvocationException: Exception has been thrown by the
Parameter name: exePath
  --- End of inner exception stack trace ---
  at System.Configuration.ConfigurationManager.OpenExeConfigurationInternal (ConfigurationUserLevel
  at System.Configuration.ConfigurationManager.OpenExeConfiguration (System.String exePath) [0x0000
  at cnvPartition.cnvPartitionProperties..ctor () [0x00000] in <filename unknown>:0
  at cnvPartition.TomsNewAlgorithm..ctor () [0x00000] in <filename unknown>:0
  at (wrapper managed-to-native) System.Reflection.MonoCMethod:InternalInvoke (object,object[],Syst
  at System.Reflection.MonoCMethod.Invoke (System.Object obj, BindingFlags invokeAttr, System.Refle
  --- End of inner exception stack trace ---
  at System.Reflection.MonoCMethod.Invoke (System.Object obj, BindingFlags invokeAttr, System.Refle
  at System.Reflection.MonoCMethod.Invoke (BindingFlags invokeAttr, System.Reflection.Binder binder
  at System.Reflection.ConstructorInfo.Invoke (System.Object[] parameters) [0x00000] in <filename u
  at System.Activator.CreateInstance (System.Type type, Boolean nonPublic) [0x00000] in <filename u
  at System.Activator.CreateInstance (System.Type type) [0x00000] in <filename unknown>:0
  at System.Reflection.Assembly.CreateInstance (System.String typeName, Boolean ignoreCase) [0x0000
  at System.Reflection.Assembly.CreateInstance (System.String typeName) [0x00000] in <filename unkn
  at WTCCC.CauliFlower.Main (System.String[] args) [0x00000] in <filename unknown>:0
```

## D.2 MONO IOMAP

If you think you have done everything correctly and it complains "files not found", you forgot to set the `MONO_IOMAP=case` environment variable (note the backslash in `Data\seqdata.bin`) (This should only

happen to non-native build and should not be seen from mid-2010 onwards):

```
BeanSprout 3.0 by Hin-Tak Leung, built against GT 3.2.33.27727
Found Genotyping Module version: 3.2.33.27727
Failed to load dat.bin  Please make sure the file exists and is in the
                                same directory as the GenTrain executable
GenTrain module: Call Version=6.3.0, Cluster Version = 6.3.1
Num of Samples: 24 with 45 columns

Unhandled Exception: System.Reflection.TargetInvocationException:
Exception has been thrown by the target of an invocation.
                                ---> System.IO.FileNotFoundException:
/home/Hin-Tak/my-project/Data\seqdata.bin does not exist
File name: '/home/Hin-Tak/my-project/Data\seqdata.bin'
```

# E Genotyping Module versions

| **GT version** | Built Date | Shipped with |
|---|---|---|
| 3.0.22 | Fri Feb 16 02:31:55 2007 | |
| 3.0.27.14356 | Wed Mar 28 17:00:21 2007 | `BeadStudioV3.0.19_Hot_Fix_updater_2007-03-26.zip` |
| 3.1.12.18034 | Tue Jul 3 19:02:58 2007 | `BeadStudioV3.1.0 2007-07-02.zip` |
| 3.1.14.16583 | | `BeadStudioV3.1_Hot_Fix_Upgrader_2007-09-25.zip` |
| 3.1.14.16583 | | `BeadStudioV3.1_Hot_Fix_Full_Installer_2007-09-25.zip` |
| 3.1.14.16583 | Tue Sep 25 18:14:40 2007 | `BeadStudioV3.1_GX_3.2_2007-10-12.zip` |
| 3.2.23.31579 | Sun Jan 13 01:33:30 2008 | `BeadStudioV3.2Installer.zip` |
| 3.2.32.16565 | Fri Feb 29 17:13:00 2008 | `BeadStudioV3.2.exe` |
| 3.2.33.27727 | Thu Apr 24 00:25:01 2008 | `BeadStudioV3.1_GX_3.3.exe` |
| 3.3.4.30944 | Fri Aug 8 02:11:29 2008 | `BeadStudio_July_2008.exe` |
| 3.3.7.30795 | Fri Oct 3 02:06:30 2008 | `BeadStudio_July_2008_Hotfix_ProteinAnalysis.exe` |

GenomeStudio's GT versions starts again from 1.x:

| **GT Assembly version** | Built Date | shipped with |
|---|---|---|
| 1.0.8.26634 | Mon Oct 13 23:47:49 2008 | GenomeStudio V2008.1plus2 (1.0.2.26594) |
| 1.0.10.20745 | Fri Oct 31 19:31:31 2008 | GenomeStudio V2008.1 (1.0.2.20706) |
| 1.1.9.0 | Tue May 19 19:54:16 2009 | GenomeStudio V2009.1 (1.1.0.19598) |
| 1.5.16.0 | Thu Dec 10 00:34:20 2009 | GenomeStudio V2009.2 (2009.2.0.29803) |
| 1.6.3.0 | Mon Mar 8 18:13:56 2010 | GenomeStudio V2010.1 (2010.1.0.18378) |
| 1.7.0.12158 | Thu Jul 29 15:45:17 2010 | Beeline Genotyping.dll |
| 1.7.4.12198 | Thu Jul 29 15:46:36 2010 | Beeline |
| 1.7.4.25862 | Mon Aug 16 23:22:04 2010 | GenomeStudio V2010.2 (2010.2.0.25786) |
| 1.8.4.30183 | Sat Dec 11 00:46:07 2010 | GenomeStudio V2010.3 (2010.3.0.30128) |
| 1.9.4.24613 | Thu May 19 22:40:27 2011 | GenomeStudio V2011.1 (2011.1.0.24550) |

Starting with GenomeStudio V2010.2, it appears that the assembly version, can be different from the file version and product version, which is reported to be 1.7.11 . (GenomeStudio V2010.3/V2011.1 reports file version and product version to be 1.7.10.0)

## E.1 Checking Genotyping Module versions

There is a version checking tool, `GTversionCheck`, and it runs on linux:

```
$ mono ./GTversionCheck.exe BeanSprout_30_3_2_33.exe
Module BeanSprout, Version=3.0.3089.33573, Culture=neutral, PublicKeyToken=null
$ mono ./GTversionCheck.exe GTversionCheck.exe
Module GTversionCheck, Version=1.0.3091.7615, Culture=neutral, PublicKeyToken=null
```

and on Windows:

```
C:\> GTversionCheck.exe ``C:\Program Files\Illumina\BeadStudio 2.0\Modules\BSGT\GenotypingBSM.dll''
Module GenotypingBSM, Version=3.2.23.31579, Culture=neutral, PublicKeyToken=null
```

# F Caveate on numerical accuracy

On a test project of 24 samples (thus generating $24 \times 561466 \times 2 = 26950368$ normalized X/Y data), there are 324 differences in the last digit of the normalized data due to numerical rounding, between BeanSprout and Final Report from an earlier version of BeadStudio running on 64-bit Windows XP. This is a difference of 1 in 83180. This may be because of:

- differences between different versions of the genotyping module (GT 3.1.12 vs GT 3.2.33)
- differences between mono and Microsoft.NET
- differences between implementation of mathematical operations on linux and windows
- differences in precisions and/or implementations of floating point operations between two CPU processors (AMD Turion vs Intel Core Duo)

I am sure most scientists don't notice this kind of differences, so I am just writing it down here just in case somebody else notices it.