

Harrix Data 1.1

А. Б. Сергиенко

24 января 2014 г.

Аннотация

Harrix Data 1.1 — формат файлов вида ***.hdata** для представления данных для обработки и отображения в графиках в программах Harrix.

Содержание

1	Вводная информация	2
2	Краткое описание формата данных	2
3	Пример файла *.hdata	2
4	Структура файла	3
5	Функции, которые обрабатывают данный формат файлов	10

1 Вводная информация

Описание данного формата файлов располагается по адресу <https://github.com/Harrix/HarrixFileFormats>.

С автором можно связаться по адресу sergienkoanton@mail.ru или <http://vk.com/harrix>. Сайт автора, где публикуются последние новости: <http://blog.harrix.org>, а проекты располагаются по адресу <http://harrix.org>.

2 Краткое описание формата данных

Файл формата **Harrix Data 1.1** имеет расширение вида ***.hdata**.

Файл представляет собой обычный текстовый файл, где информация располагается по строкам. Вначале файла идет служебная информация, а потом идут непосредственно данные.

В качестве разделителя для числовых данных используется точка, а не запятая.

3 Пример файла *.hdata

Обратите внимание, что если Вы будете копировать примеры данных между строчками **BeginData** и **EndData** из данного документа, то между числами вместо знака табуляции будет знак пробела. Это неправильно! Так что копируйте из файлов, которые находятся на сайте <https://github.com/Harrix/HarrixFileFormats> в папке **Examples** в папке **Harrix Data 1.1** данного формата файлов или меняйте пробел на знак табуляции.

Код 1. Пример файла *.hdata

```
HarrixFileFormat = Harrix Data 1.1
Site = https://github.com/Harrix/HarrixFileFormats
Type = PointsAndLine
Title = Тестовый график
AxisX = x
AxisY = y
Parameters = ShowLine, ShowPoints, ShowArea, ShowSpecPoints
BeginNamesOfCharts
Количество деревьев
Количество домов
EndNamesOfCharts
BeginData
-0.807560891 -0.94366779 -3.14 -0.001592653
0.00127521 -0.128120962 -3.04 -0.101417986
-2.489437639 -0.611951554 -2.94 -0.200229985
0.445603028 0.599178044 -2.84 -0.297041351
1.586889318 1.085454093 -2.74 -0.390884779
0.746497053 0.566555897 -2.64 -0.480822615
2.568177686 0.617068194 -2.54 -0.56595623
EndData
```

4 Структура файла

Вначале в файле в первых двух строчках идет служебная информация, которая описывает формат файла.

- **Harrix File Format.** В первой строчке содержится название формата файла:

Код 2. Первая строчка файла *.hdata

```
HarrixFileFormat = Harrix Data 1.1
```

Она должна содержаться в каждом файле и не подлежит изменению.

- **Site.** Во второй строчке содержится информация о сайте, на котором содержится данное описание формата файла:

Код 3. Вторая строчка файла *.hdata

```
Site = https://github.com/Harrix/HarrixFileFormats
```

Она должна содержаться в каждом файле и не подлежит изменению.

Далее идут строчки описывающие данные, которые содержатся в файле. Данные строчки могут отсутствовать.

- **Type.** В следующей строчке написан формат данных, содержащихся в данном файле.

Код 4. Строчка с описанием типа данных в файле

```
Type = TwoIndependentChartsOfPointsAndLine
```

Данный параметр может принимать следующие значения:

- **Line;**
- **TwoLines;**
- **TwoIndependentLines;**
- **SeveralIndependentLines;**
- **SeveralLines;**
- **PointsAndLine;**
- **Bar;**
- **3DPoints;**
- **NPoints.**

Рассмотрим все возможные варианты более подробно.

- **Line** — файл содержит информацию о массиве точек с координатами x и y. То есть на графике выведется одна линия по точкам. Нет требования для того, чтобы массив точек был отсортирован. Первый столбец содержит координаты точек по оси x. Второй столбец содержит координаты точек по оси y. В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться только одна строчка, которая содержит название линии:

Код 5. Для Line

```
BeginNamesOfCharts  
Название линии  
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 6. Для Line

```
BeginData  
3 0.111111  
4 0.0666667  
5 0.04  
6 0.0285714  
7 0.0204082  
8 0.015873  
9 0.0123457  
10 0.010101  
EndData
```

- **TwoLines** — файл содержит информацию о двух линиях с одинаковыми значениями по оси Ох. То есть на графике выведется две линии по точкам. Нет требования для того, чтобы массив точек был отсортирован. Первый столбец содержит координаты точек по оси х. Второй столбец содержит координаты точек по оси у первой линии. Третий столбец содержит координаты точек по оси у второй линии. Обратите внимание, что в этом случае число точек в обеих линиях одинаковое.

В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться две строчки, которая содержит название линии:

Код 7. Для TwoLines

```
BeginNamesOfCharts  
Количество домов  
Количество участков  
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 8. Для TwoLines

```
BeginData  
3 5.111111 9  
7 7.0666667 -1  
5 4.04 12  
6 2.0285714 5  
7.1 8.0204082 8  
8 0.015873 4  
9 6.0123457 9  
5.5 4.010101 1  
11 15.00826446 7  
EndData
```

- **TwoIndependentLines** — файл содержит информацию о двух линиях с разными значениями по оси Ох. То есть на графике выведется две линии по точкам. Нет требования для того, чтобы массив точек был отсортирован. Первый

столбец содержит координаты точек по оси x первой линии. Второй столбец содержит координаты точек по оси x второй линии. Третий столбец содержит координаты точек по оси y первой линии. Четвертый столбец содержит координаты точек по оси y второй линии. Количество элементов в каждом столбце должно быть одинаково. Но количество точек в каждой линии может различаться (например, число точек в первой линии — 6 штук, а для второй — 8). Поэтому (для рассматриваемого случая) в первом и втором столбцах, отвечающих за первую линию в «пустых ячейках» ставится знак минуса (в примере ниже это продемонстрировано). Если в первой линии элементов больше, то, соответственно, в третьем и четвертом столбцах будут «пустые ячейки» с минусами.

В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться две строчки, которая содержит название линии:

Код 9. Для TwoIndependentLines

```
BeginNamesOfCharts
Количество домов
Количество участков
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 10. Для TwoIndependentLines

```
BeginData
-3.5  5  4  8
5.3   4  6  1
4.2   7  1  1.9
2.6   2  2  7
1.4  -4  3  5
5.5   7  4  4
-   -  8  6
-   -  9  7
EndData
```

- **SeveralLines** — файл содержат информацию о нескольких линиях (n штук) с одинаковыми значениями по оси Ox . То есть на графике выведется несколько линии по точкам. Нет требования для того, чтобы массив точек был отсортирован. Первый столбец содержит координаты точек по оси x . Остальные столбцы содержат координаты точек по оси y каждой линии. Обратите внимание, что в этом случае число точек во всех линиях одинаковое.

В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться n строчек, которые содержат названия линий:

Код 11. Для SeveralLines

```
BeginNamesOfCharts
Количество домов
Количество участков
Количество лужаек
Количество тазиков
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно) ниже. Количество столбцов n .

Код 12. Для SeveralLines

```
BeginData
0.029462501 0.14561098 1.393046208 2.241360938 3.1353736
0.402566025 0.045810082 1.526551265 2.06864077 3.4732082
0.061311083 0.095163714 1.445856657 2.831574039 3.5118008
0.765765255 0.028180801 1.95558673 2.22015302 3.3458553
0.415193096 0.2694622 1.721272806 2.696043386 3.9191817
0.416133908 0.874141366 1.609156731 2.915444486 3.3856970
0.408535896 0.548411006 1.749252223 2.812573789 3.0996358
0.269031516 0.808771887 1.601134047 2.877347363 3.5698883
0.374612333 0.50509752 1.77570622 2.604983261 3.3258520
EndData
```

- **SeveralIndependentLines** — файл содержат информацию о нескольких линиях (n штук) с разными значениями по оси Ох. То есть на графике выведется две линии по точкам. Нет требования для того, чтобы массив точек был отсортирован. Столбцы идут попарно: x и y координаты точек каждой линии. Количество элементов в каждом столбце должно быть одинаково. Но количество точек в каждой линии может различаться (например, число точек в первой линии — 6 штук, а для второй — 8). Поэтому на месте «пустых ячеек» ставится знак минуса (в примере ниже это продемонстрировано).

В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться n строчек, которые содержат названия линий:

Код 13. Для SeveralIndependentLines

```
BeginNamesOfCharts
Количество домов
Количество участков
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 14. Для SeveralIndependentLines

```
BeginData
-3.5 5 4 8
5.3 4 6 1
4.2 7 1 1.9
2.6 2 2 7
1.4 -4 3 5
5.5 7 4 4
- - 8 6
- - 9 7
EndData
```

- **PointsAndLine** — файл содержат информацию о двух линиях с разными значениями по оси Ох. Фактически это тоже самое, что и **TwoIndependentLines**. Но на графике первая линия выведется в виде точек, а вторая в виде линии. Это удобно для отображения регрессий. Нет требования для того, чтобы массив точек был отсортирован. Первый столбец содержит координаты точек по оси x первой линии. Второй столбец содержит координаты точек по оси x второй линии. Третий столбец содержит координаты точек по оси y первой линии. Четвертый столбец содержит координаты точек по оси y второй линии. Количество элементов в каждом столбце должно быть одинаково. Но количество точек

в каждой линии может различаться (например, число точек в первой линии — 6 штук, а для второй — 8). Поэтому (для рассматриваемого случая) в первом и втором столбцах, отвечающих за первую линию в «пустых ячейках» ставится знак минуса (в примере ниже это продемонстрировано). Если в первой линии элементов больше, то, соответственно, в третьем и четвертом столбцах будут «пустые ячейки» с минусами.

В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться две строчки, которая содержит название линии:

Код 15. Для PointsAndLine

```
BeginNamesOfCharts
Количество домов
Количество участков
EndNamesOfCharts
```

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 16. Для PointsAndLine

```
BeginData
-3.5  5  4  8
5.3   4  6  1
4.2   7  1  1.9
2.6   2  2  7
1.4  -4  3  5
5.5   7  4  4
- -   8  6
- -   9  7
EndData
```

- **Bar** — файл содержат информацию о некоторой гистограмме. То есть на графике выведется столбчатый график. Содержится только один столбец с данными по оси Oy. Каждая цифра будет означать высоту соответствующего столбца. В блоке между строчками **BeginNamesOfCharts** и **EndNamesOfCharts** должно содержаться столько строк, сколько данных находится в данных. Это будут названия столбцов:

Код 17. Для Bar

```
BeginNamesOfCharts
цинк
серебро
бром
шоколад
EndNamesOfCharts
```

Описываемый ниже параметр **AxisX** в данном типе графиков отсутствует. Из возможных дополнительных параметров может присутствовать только **MinZero**.

Пример данных, которые содержатся в между строчками **BeginData** и **EndData** (с этими строчками включительно):

Код 18. Для Bar

```
BeginData
5
```

```
6
1
3
EndData
```

- **3DPoints** — файл содержат информацию о точках в трехмерном пространстве. Каждая строка это одна точка. Первый столбец дает координаты по оси Ох, второй - Оу, третий - Oz.

В блоке между строками **BeginNamesOfCharts** и **EndNamesOfCharts** ничего не содержится:

Код 19. Для 3DPoints

```
BeginNamesOfCharts
EndNamesOfCharts
```

Дополнительных параметров нет.

Пример данных, которые содержатся в между строками **BeginData** и **EndData** (с этими строками включительно):

Код 20. Для 3DPoints

```
BeginData
0.029462501 0.14561098 1.393046208
0.402566025 0.045810082 2.526551265
0.061311083 0.095163714 1.445856657
0.765765255 0.028180801 2.95558673
0.415193096 0.2694622 1.721272806
0.416133908 0.874141366 1.609156731
EndData
```

- **NPoints** — файл содержат информацию о многомерных точках.

В блоке между строками **BeginNamesOfCharts** и **EndNamesOfCharts** должна содержаться n строчек, которые содержат названия координат точек:

Код 21. Для NPoints

```
BeginNamesOfCharts
Количество домов
Количество участков
Количество лужаек
Количество тазиков
Вероятность выбора
EndNamesOfCharts
```

Пример данных, которые содержатся в между строками **BeginData** и **EndData** (с этими строками включительно) ниже. Количество столбцов $n + 1$.

Код 22. Для NPoints

```
BeginData
0.029462501 0.14561098 1.393046208 2.241360938 3.1353736
0.402566025 0.045810082 1.526551265 2.06864077 3.4732082
0.061311083 0.095163714 1.445856657 2.831574039 3.5118008
0.765765255 0.028180801 1.95558673 2.22015302 3.3458553
0.415193096 0.2694622 1.721272806 2.696043386 3.9191817
0.416133908 0.874141366 1.609156731 2.915444486 3.3856970
0.408535896 0.548411006 1.749252223 2.812573789 3.0996358
0.269031516 0.808771887 1.601134047 2.877347363 3.5698883
0.374612333 0.50509752 1.77570622 2.604983261 3.3258520
EndData
```


- **Title.** В следующей строчке вводится название данного набора данных или графика.

Код 23. Название графика или набора данных

```
Title = Тестовый график
```

- **AxisX.** В следующей строчке вводится название оси Ох для отображения на графике. Отсутствует при Type=Bar.

Код 24. Название оси Ох

```
AxisX = x
```

- **AxisY.** В следующей строчке вводится название оси Оу для отображения на графике.

Код 25. Название оси Оу

```
AxisY = y
```

- **AxisZ.** В следующей строчке вводится название оси Oz для отображения на графике. Только для Type=3DPoints.

Код 26. Название оси Oz

```
AxisZ = z
```

- **Parameters.** В следующей строчке через запятую вводятся параметры, которые могут принимать значения:
 - **ShowLine** — графики будут отображаться в виде линий. Если параметра не будет, то будут отображаться только точки;
 - **ShowPoints** — графики будут отображаться в виде точек. Если будут включены оба параметра ShowLine и ShowPoints, то будет показываться и линия, и точки;
 - **ShowArea** — под графиком будет показываться закрашенная область;
 - **ShowSpecPoints** — показывать на графике специальные точки-метки: начальную точку, максимальные и минимальные значения;
 - **RedLine** — показывать на графике линию красного цвета (только для Type = Line);
 - **MinZero** — показывать на графике линию красного цвета (только для Type = Bar).

Параметры не обязательные и по умолчанию отсутствуют. Если нет ни параметра ShowLine, ни ShowPoints, то ShowLine поставится автоматически. В этой строчке параметры могут идти в произвольном порядке и могут быть не в полном составе.

Код 27. Строка параметров

```
Parameters = ShowLine, ShowPoints, ShowArea, ShowSpecPoints
```

Потом идет блок, который обрамляется строчками **BeginNamesOfCharts** и **EndNamesOfCharts**. В нем записываются наименования линий, согласно типу данных. Обратите внимание, что это не наименования каждого столбца. Для каждого типа данных выше расписано содержимое этого блока. Данный блок не обязателен.

Наконец, идет блок, который обрамляется строчками **BeginData** и **EndData**. В данном блоке идут непосредственно данные, и он обязателен. Данные располагаются в виде стандартного представления строк и столбцов. То есть каждая строка описывает одну точку или несколько точек (для некоторых типов данных). Каждый столбец содержит данные одного типа. Все столбцы разделяются знаком табуляции. Если в какой-то ячейке нет данных (например, данные содержат данные о двух линиях с разным количеством точек), то на месте пропуска ставится знак минуса «-».

5 Функции, которые обрабатывают данный формат файлов

В библиотеке <https://github.com/Harrix/HarrixQtLibraryForQWebView> на языке C++ имеются функции, которые обрабатывают данный формат файлов с среде Qt. К таким функциям относятся:

- QString **HQt_ReadHdataToHtmlChart** (QString filename) — функция возвращает строку с HTML кодом графика в результате считывания информации из *.hdata.

В библиотеке <https://github.com/Harrix/HarrixQtLibraryForLaTeX> на языке C++ имеются функции, которые обрабатывают данный формат файлов с среде Qt. К таким функциям относятся:

- QString **HQt_ReadHdataToLatexChart** (QString filename) — функция возвращает строку с \LaTeX кодом графика в результате считывания информации из *.hdata.