

A Data Viewer for R

Paul Murrell

The University of Auckland

July 30 2009

Overview

- Motivation:** STATS 220
- Problem statement:** Students do not understand what they cannot see.
- What doesn't work:** `View()`
- A solution:** The **`rdataviewer`** package and the `tcltkViewer()` function.
- What else?:** Novel navigation interface, zooming, extensible for other data sources.

STATS 220 Data Technologies

- HTML (and CSS), XML (and DTDs), SQL (and databases), and **R** (and regular expressions)
- Online text book that nobody reads
- Computer lab each week (worth 0.5%) + three Assignments
 - 5 labs + one assignment on R
 - Emphasis on creating and modifying data structures
 - Attempt to use real data

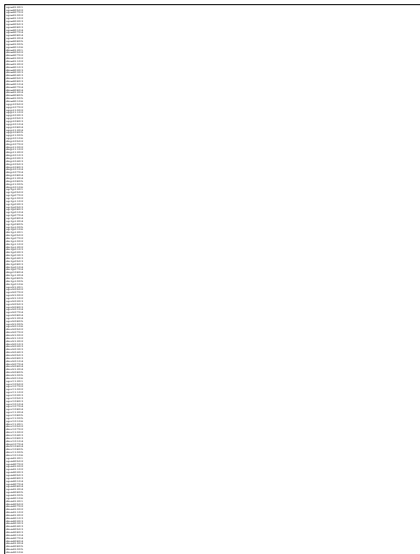
Example Lab Question

Read the file `lab10.txt` into R as a character vector.

You should end up with a symbol `habitats` that prints like this (this shows just the first 10 values; there are 192 values in total):

```
> head(habitats, 10)
 [1] "upwd1201" "upwd0502" "upwd0702"
 [4] "upwd1002" "upwd1102" "upwd0203"
 [7] "upwd0503" "upwd0803" "upwd0104"
[10] "upwd0704"
```

The file lab10.txt



```
1 # A tibble: 1 x 1  
2   x  
3   1
```

Example Student Answer

```
habitats <- head(readLines("lab10.txt"), 10)
```

To be fair, it is difficult to “see” all of an R data structure, unless the data structure is very small.

The main reason that I (occasionally) use a spreadsheet is just to be able to see the raw data in fixed-width columns.

The View() function

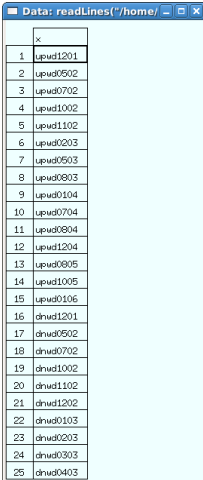
View(habitats)

<PgUp>

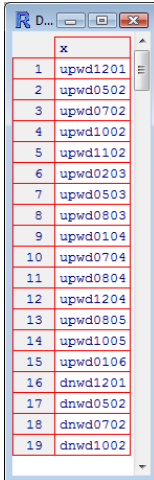
<PgDn>

<Home>

<End>



	x
1	upwd1201
2	upwd0502
3	upwd0702
4	upwd1002
5	upwd1102
6	upwd0203
7	upwd0503
8	upwd0803
9	upwd0104
10	upwd0704
11	upwd0804
12	upwd1204
13	upwd0805
14	upwd1005
15	upwd0106
16	dnwd1201
17	dnwd0502
18	dnwd0702
19	dnwd1002
20	dnwd1102
21	dnwd1202
22	dnwd0103
23	dnwd0203
24	dnwd0303
25	dnwd0403

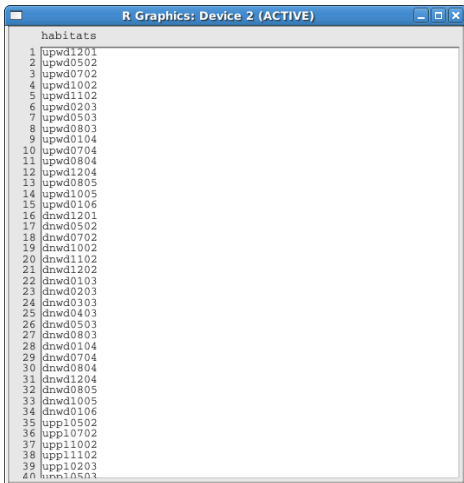
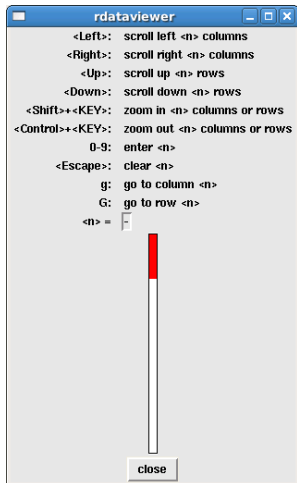


	x
1	upwd1201
2	upwd0502
3	upwd0702
4	upwd1002
5	upwd1102
6	upwd0203
7	upwd0503
8	upwd0803
9	upwd0104
10	upwd0704
11	upwd0804
12	upwd1204
13	upwd0805
14	upwd1005
15	upwd0106
16	dnwd1201
17	dnwd0502
18	dnwd0702
19	dnwd1002

The `tcltkViewer()` function

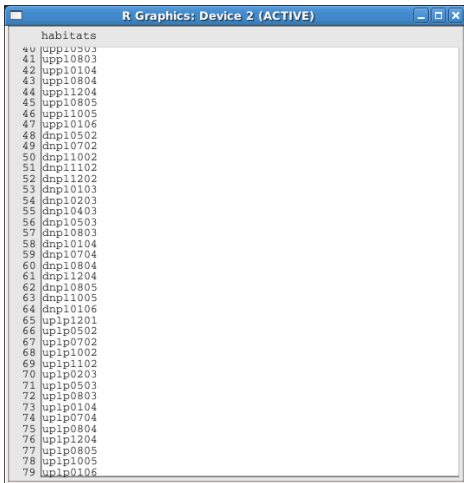
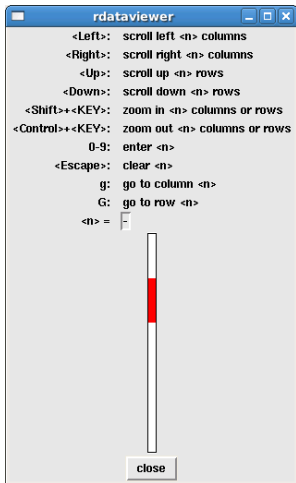
```
habitats <- readLines("lab10.txt")  
  
data <- viewerData(habitats)  
  
vdvp <- viewerDeviceVp(data)  
  
v <- simpleViewer(data, dev=vdvp)  
  
tcltkViewer(v)
```


The tcltkViewer() function



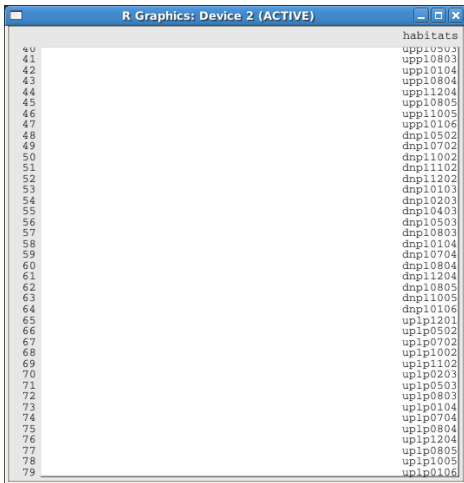
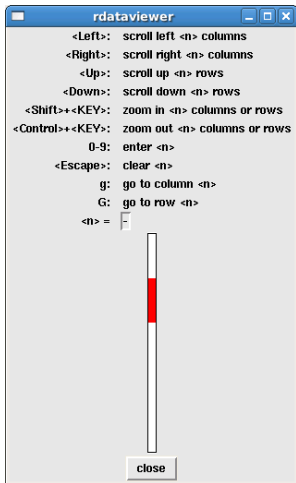
Standard-ish navigation (PgDn)

Now in "bottom-to-top" mode.



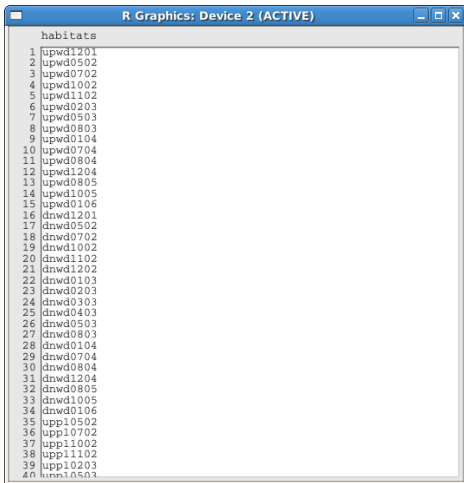
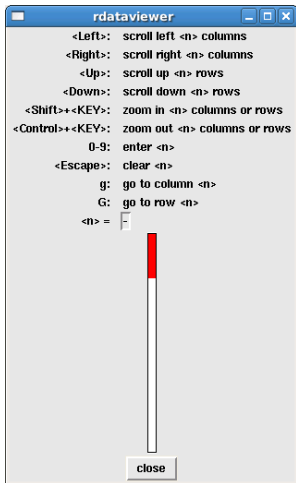
Standard-ish navigation (Right)

Now in "right-to-left" mode.

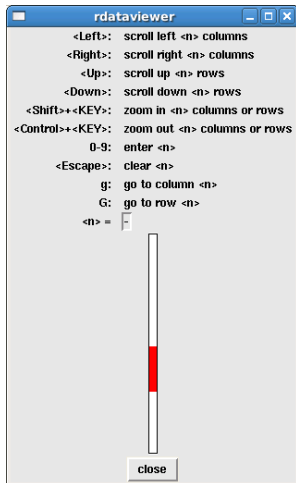


Standard-ish navigation (Ctrl-Home)

Now back in "top-to-bottom" and "left-to-right" mode.



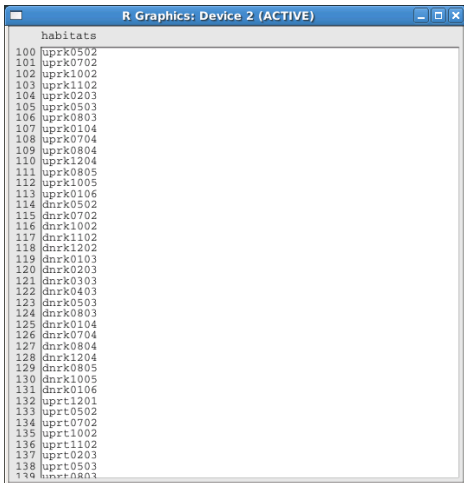
vi-ish navigation (100 G)



The `rdataviewer` window displays a list of navigation and editing commands:

- `<Left>`: scroll left `<n>` columns
- `<Right>`: scroll right `<n>` columns
- `<Up>`: scroll up `<n>` rows
- `<Down>`: scroll down `<n>` rows
- `<Shift>+<KEY>`: zoom in `<n>` columns or rows
- `<Control>+<KEY>`: zoom out `<n>` columns or rows
- `0-9`: enter `<n>`
- `<Escape>`: clear `<n>`
- `g`: go to column `<n>`
- `G`: go to row `<n>`
- `<n> =`

A vertical scrollbar is visible below the text, with a red highlight indicating the current position. A `close` button is located at the bottom of the window.



The `R Graphics: Device 2 (ACTIVE)` window displays a list of habitats, numbered 100 to 139:

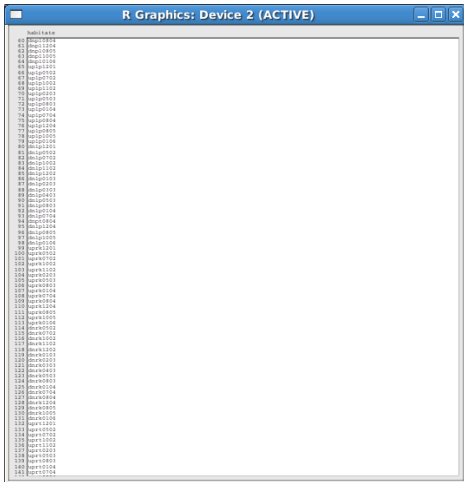
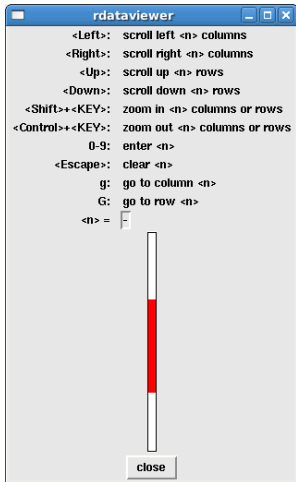
```
habitats
100 uprk0502
101 uprk0702
102 uprk1002
103 uprk1102
104 uprk0203
105 uprk0503
106 uprk0803
107 uprk0104
108 uprk0704
109 uprk0804
110 uprk1204
111 uprk0805
112 uprk1005
113 uprk0106
114 dnrk0502
115 dnrk0702
116 dnrk1002
117 dnrk1102
118 dnrk1202
119 dnrk0103
120 dnrk0203
121 dnrk0303
122 dnrk0403
123 dnrk0503
124 dnrk0803
125 dnrk0104
126 dnrk0704
127 dnrk0804
128 dnrk1204
129 dnrk0805
130 dnrk1005
131 dnrk0106
132 uprt1201
133 uprt0502
134 uprt0702
135 uprt1002
136 uprt1102
137 uprt0203
138 uprt0503
139 uprt0803
```

rdataviewer navigation

- Navigation is row- and column-based.
- There is an up-down mode and a left-right mode.
- Navigation commands can be preceded by a number (e.g., scroll down 3 rows).

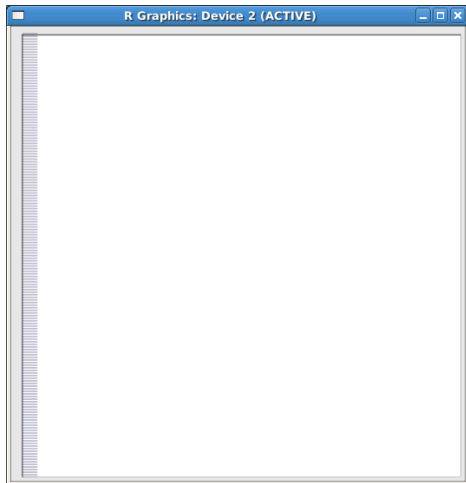
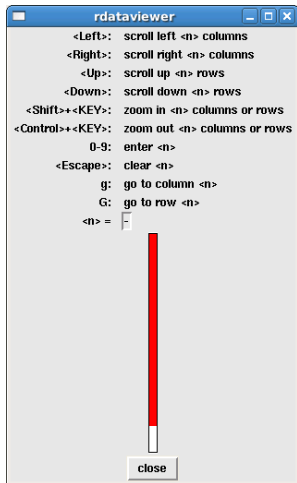
Zooming out (Ctrl-PgUp)

View twice as many rows.



Zooming out (Ctrl-PgUp)

Below a threshold, lines are drawn instead of text.



Zooming in (Shift-PgUp)

View half as many rows.

rdataviewer

- <Left>: scroll left <n> columns
- <Right>: scroll right <n> columns
- <Up>: scroll up <n> rows
- <Down>: scroll down <n> rows
- <Shift>+<KEY>: zoom in <n> columns or rows
- <Control>+<KEY>: zoom out <n> columns or rows
- 0-9: enter <n>
- <Escape>: clear <n>
- g: go to column <n>
- G: go to row <n>
- <n> =

close

R Graphics: Device 2 (ACTIVE)

```

substate
86 86mLp0204
87 86mLp0203
88 86mLp0403
89 86mLp0403
90 86mLp0203
91 86mLp0803
92 86mLp0704
93 86mLp0804
94 86mLp0804
95 86mLp1204
96 86mLp0106
97 86mLp0106
98 86mLp0106
99 86mLp0106
100 86mLp0502
101 86mLp0702
102 86mLp1003
103 86mLp1102
104 86mLp0203
105 86mLp0803
106 86mLp0104
107 86mLp0704
108 86mLp0704
109 86mLp0104
110 86mLp0104
111 86mLp0805
112 86mLp1005
113 86mLp0108
114 86mLp0502
115 86mLp0702
116 86mLp0502
117 86mLp1102
118 86mLp1202
119 86mLp0103
120 86mLp0103
121 86mLp0103
122 86mLp0403
123 86mLp0103
124 86mLp0803
125 86mLp0704
126 86mLp0704
127 86mLp0804
128 86mLp1204
129 86mLp0805
130 86mLp1005
131 86mLp0104
132 86mLp1101
133 86mLp1101
134 86mLp0502
135 86mLp1002
136 86mLp1002
137 86mLp1002
138 86mLp0203
139 86mLp0803
140 86mLp0104
141 86mLp0704
142 86mLp0804
143 86mLp1204
144 86mLp0805
145 86mLp0106
146 86mLp0106
147 86mLp1201
148 86mLp0502
149 86mLp0702
150 86mLp1003
151 86mLp0403
152 86mLp0803
153 86mLp0104
154 86mLp0704
155 86mLp0804
156 86mLp0805
157 86mLp1005
158 86mLp1008
159 86mLp0101
160 86mLp0702
161 86mLp0502
162 86mLp0103
163 86mLp0103
164 86mLp0803
165 86mLp0703
166 86mLp0803
167 86mLp0104
168 86mLp0704
169 86mLp0104
  
```

rdataviewer zooming

- Ctrl-`<command>` zooms out.
- Shift-`<command>` zooms in.
- Zooming is row- and column-based.
- There is still an up-down mode and a left-right mode.
- Zooming commands can be preceded by a number (e.g., view 3 extra rows).

Viewing the exo-planets data set

```
> allPlanetData <- read.csv("exoplanets.csv")
```

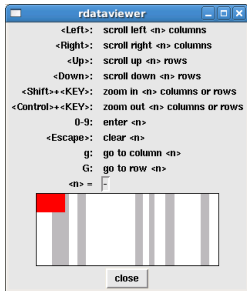
```
> head(allPlanetData)
```

	Planet	Star	Constellation	Year.of.Discovery	Discovered.By	Orbital.Radius
1	PSR 1257 a	PSR 1257		1991		
2	PSR 1257 b	PSR 1257		1991		
3	PSR 1257 c	PSR 1257		1994		
4	PSR 1257 d	PSR 1257		1994		
5	51 Pegasi b	51 Pegasi	Pegasus	1995		
6	Upsilon Andromedae b	Upsilon Andromedae	Andromedae	1996		
1	Alexander Wolszczan and Dale Frail, National Radio Astronomy Observatory					0.19 AU
2	Alexander Wolszczan and Dale Frail of the National Radio Observatory in New Mexico					0.36 AU
3						0.46 AU
4						40 AU
5	Michel Mayor and Didier Queloz, Geneva Observatory					0.05 AU
6	Geoffrey Marcy and R. Paul Butler, San Francisco State University					0.05 AU
	Orbit.Period	Planet.Type	Planet.Mass	Orbit.Eccentricity		
1	25.262s	Pulsar	0 (Earth = 1)	-1.000		
2	66.5s	Pulsar	0 (Earth = 1)	0.100		
3	98.2s	Pulsar	0 (Earth = 1)	0.020		
4	62050s	Pulsar	0 (Earth = 1)	-1.000		
5	4.23s	Hot Jupiter	0.47 (Jupiter = 1)	0.000		
6	4.62s	Hot Jupiter	0.71 (Jupiter = 1)	0.034		
		Method.of.Detection	Star.Coords...RA.	Star.Coords...DEC.		
1		Timing	13 00 01	+12 40 00		
2		Timing	13 00 01	+12 40 00		
3		Timing	13 00 01	+12 40 00		
4		Timing	13 00 01	+12 40 00		
5	Radial Velocity (or Doppler Spectroscopy)		22 57 27.14	+20 46 04.5		
6	Radial Velocity (or Doppler Spectroscopy)		01 36 48.527	+41 24 38.71		

Viewing the exo-planets data set

```
data <- viewerDataFrame(read.csv("exoplanets.csv"))  
  
vdvp <- viewerDeviceVp(data)  
  
v <- simpleViewer(data, dev=vdvp)  
  
tcltkViewer(v)
```

Viewing the exo-planets data set

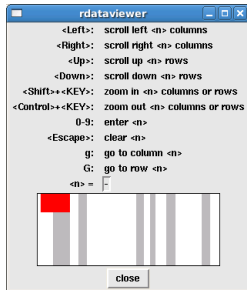


The screenshot shows the 'R Graphics: Device 2 (ACTIVE)' window displaying a text-based table of exoplanet data. The table has 40 rows and 3 columns. The first column is labeled 'Planet'.

	Planet	
1	PSR 1257 a	
2	PSR 1257 b	
3	PSR 1257 c	
4	PSR 1257 d	
5	51 Pegasi b	5
6	Upsilon Andromedae b	Upsilon An
7	55 Cancri b	5
8	47 Ursae Majoris b	47 Ursae
9	tau Boo	ta
10	70 Virginis b	70
11	rho CrB	rho Coronae
12	16 Cygni b	
13	HD 217107 b	H
14	HD 210277 b	H
15	HD 187123 b	H
16	Gliese 876 b	Gl
17	HD 195019	H
18	HD 168443 b	H
19	HD 168443 c	H
20	14 Herculis b	14
21	HD 209458 b	H
22	HD 192263 b	H
23	HD 37124 b	
24	HD 130322 b	H
25	HD 177830 b	H
26	HD 134987 b	H
27	HR 810 b	
28	Upsilon Andromedae c	Upsilon An
29	Upsilon Andromedae d	Upsilon An
30	HD 222582 b	H
31	HD 10697 b	
32	HD 83443 b	
33	HD 168746 b	H
34	HD 46375 b	
35	HD 108147 b	H
36	HD 75289 b	
37	BD -10 3166 b	BD
38	HD 6434 b	
39	Epsilon Eridani b	Epsilon
40	HD 18529 b	

Standard-ish navigation (Right)

Now the "right-to-left" mode might make more sense.

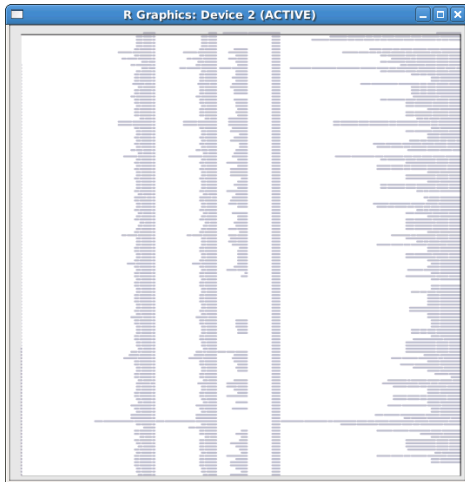
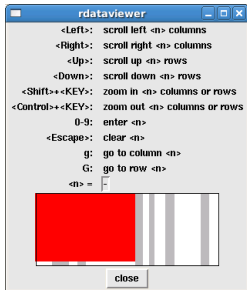


The screenshot shows a window titled "R Graphics: Device 2 (ACTIVE)" displaying a table of star data. The table has two columns: "Planet" and "Star".

	Planet	Star
1	PSR 1257 a	PSR 1257
2	PSR 1257 b	PSR 1257
3	PSR 1257 c	PSR 1257
4	PSR 1257 d	PSR 1257
5	51 Pegasi b	51 Pegasi
6	Upsilon Andromedae b	Upsilon Andromedae
7	55 Cancri b	55 Cancri
8	47 Ursae Majoris b	47 Ursae Majoris
9	tau Boo	tau Bootis
10	70 Virginis b	70 Virginis
11	rho CrB	rho Coronae Borealis
12	16 Cygni b	16 Cygni
13	HD 217107 b	HD 217107
14	HD 210277 b	HD 210277
15	HD 187123 b	HD 187123
16	Gliese 876 b	Gliese 876
17	HD 195019	HD 195019
18	HD 168443 b	HD 168443
19	HD 168443 c	HD 168443
20	14 Herculis b	14 Herculis
21	HD 209458 b	HD 209458
22	HD 192263 b	HD 192263
23	HD 37124 b	HD 37124
24	HD 130322 b	HD 130322
25	HD 177830 b	HD 177830
26	HD 134987 b	HD 134987
27	HR 810 b	HR 810
28	Upsilon Andromedae c	Upsilon Andromedae
29	Upsilon Andromedae d	Upsilon Andromedae
30	HD 222582 b	HD 222582
31	HD 10697 b	HD 10697
32	HD 83443 b	HD 83443
33	HD 168746 b	HD 168746
34	HD 46375 b	HD 46375
35	HD 108147 b	HD 108147
36	HD 75289 b	HD 75289
37	BD -10 3166 b	BD -10 3166
38	HD 6434 b	HD 6434
39	Epsilon Eridani b	Epsilon Eridani
40	HD 38529 b	HD 38529

Zooming out (3 Ctrl-Right)

View 3 more columns.



The design of `rdataviewer`

- A data source to view.

```
data <- viewerDataFrame(read.csv("exoplanets.csv"))
```

- A device to draw the view on.

```
vdvp <- viewerDeviceVp(data)
```

- An object to represent the current view.

```
v <- simpleViewer(data, dev=vdvp)
```

- A GUI to allow interactive modification of the view.

```
tcltkViewer(v)
```

Each component is represented by an S4 virtual class so alternative implementations are possible.

The ViewerData class

Examples we have already seen:

- A character vector (`ViewerDataVector`)

```
data <- viewerData(habitats)
```

- A data frame (`ViewerDataFrame`)

```
data <- viewerDataFrame(read.csv("exoplanets.csv"))
```

The ViewerData class

Generics that require implementing:

- How many rows and columns?
`dimensions(data)`
- How wide is each column?
`colWidths(data, which)`
- What are the column names?
`colNames(data, cols)`
- Provide a text representation of given rows and columns
`getText(data rows, cols)`

The ViewerDataFrame class

Methods for viewing data frames:

- How many rows and columns?
`dim(data@df)`
- How wide is each column?
`max(nchar(capture.output(print(data@df[i])))`
- What are the column names?
`colnames(data@df[, cols])`
- Provide a text representation of given rows and columns
`capture.output(print(data[rows, cols]))`

The ViewerDataText class

Methods for viewing external text files:

Scan file to build an index of pointers to line beginnings.

- How many rows and columns?
Number of line beginnings (only 1 column).
- How wide is each column?
Longest difference between line beginnings.
- What are the column names?
Filename.
- Provide a text representation of given rows and columns
Seek to the relevant line beginning in the file then readLines().

The Meter Readings data set

- 300 MB XML file (4,772,013 lines)

```
<LOADPROFILE RETAILER="MEEN" TIMESTAMPSTART="01/10/2007 00:00:00" TIMESTAMPEND="30/10/2008 23:59:59">
  <METER ICP="XXXXXXXX" SERIALNUMBER="XXXXXXXX" TARIFF="IN5" MRU="XXXXXXXX" UOM="kWh">
    <READING TIMESTAMP="2007-10-01 00:00:00" RAWREADING="0.723" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 00:30:00" RAWREADING="0.683" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 01:00:00" RAWREADING="0.747" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 01:30:00" RAWREADING="0.654" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 02:00:00" RAWREADING="0.707" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 02:30:00" RAWREADING="0.853" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 03:00:00" RAWREADING="0.667" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 03:30:00" RAWREADING="0.474" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 04:00:00" RAWREADING="0.182" QUALITYFLAG="7"/>
    <READING TIMESTAMP="2007-10-01 04:30:00" RAWREADING="0.058" QUALITYFLAG="7"/>
    ...
  </METER>
</LOADPROFILE>
```

- vi took 3.5 minutes just to open and close the file.
- Emacs cannot open the file (buffer size exceeded).

The Meter Readings data set

```
data <- viewerDataText("MEENDL30102008_001-blind.XML",  
                      index=TRUE)  
  
vdvp <- viewerDeviceVp(data)  
  
v <- simpleViewer(data, dev=vdvp)  
  
tcltkViewer(v)
```

The Meter Readings data set

rdataviewer

- <Left>: scroll left <n> columns
- <Right>: scroll right <n> columns
- <Up>: scroll up <n> rows
- <Down>: scroll down <n> rows
- <Shift>+<KEY>: zoom in <n> columns or rows
- <Control>+<KEY>: zoom out <n> columns or rows
- 0-9: enter <n>
- <Escape>: clear <n>
- g: go to column <n>
- G: go to row <n>
- <n> =

close

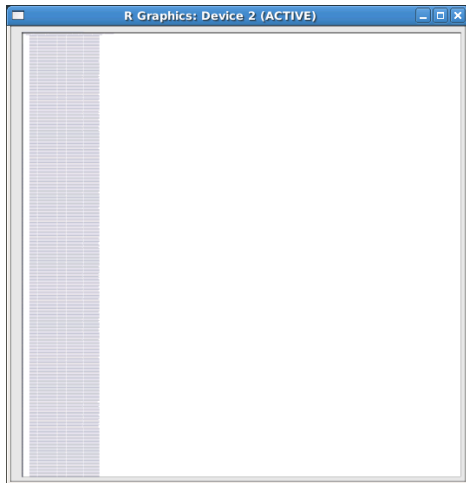
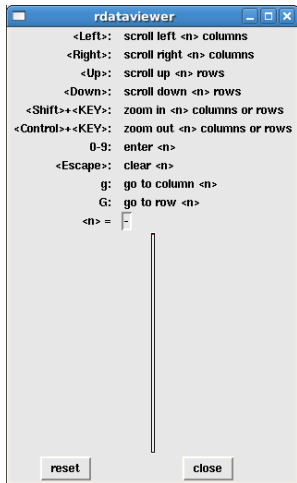
R Graphics: Device 2 (ACTIVE)

```

/home/fos/pmur002/Talks/Dept2009/Metrix/MEENDL30102008_0
1 <LOADPROFILE RETAILER="MEEN" TIMESTAMPSTART="01/10/2007
2 <METER ICP="XXXXXXXX" SERIALNUMBER="XXXXXXXX" TARIFF
3 <READING TIMESTAMP="2007-10-01 00:00:00" RAWREAD
4 <READING TIMESTAMP="2007-10-01 00:30:00" RAWREAD
5 <READING TIMESTAMP="2007-10-01 01:00:00" RAWREAD
6 <READING TIMESTAMP="2007-10-01 01:30:00" RAWREAD
7 <READING TIMESTAMP="2007-10-01 02:00:00" RAWREAD
8 <READING TIMESTAMP="2007-10-01 02:30:00" RAWREAD
9 <READING TIMESTAMP="2007-10-01 03:00:00" RAWREAD
10 <READING TIMESTAMP="2007-10-01 03:30:00" RAWREAD
11 <READING TIMESTAMP="2007-10-01 04:00:00" RAWREAD
12 <READING TIMESTAMP="2007-10-01 04:30:00" RAWREAD
13 <READING TIMESTAMP="2007-10-01 05:00:00" RAWREAD
14 <READING TIMESTAMP="2007-10-01 05:30:00" RAWREAD
15 <READING TIMESTAMP="2007-10-01 06:00:00" RAWREAD
16 <READING TIMESTAMP="2007-10-01 06:30:00" RAWREAD
17 <READING TIMESTAMP="2007-10-01 07:00:00" RAWREAD
18 <READING TIMESTAMP="2007-10-01 07:30:00" RAWREAD
19 <READING TIMESTAMP="2007-10-01 08:00:00" RAWREAD
20 <READING TIMESTAMP="2007-10-01 08:30:00" RAWREAD
21 <READING TIMESTAMP="2007-10-01 09:00:00" RAWREAD
22 <READING TIMESTAMP="2007-10-01 09:30:00" RAWREAD
23 <READING TIMESTAMP="2007-10-01 10:00:00" RAWREAD
24 <READING TIMESTAMP="2007-10-01 10:30:00" RAWREAD
25 <READING TIMESTAMP="2007-10-01 11:00:00" RAWREAD
26 <READING TIMESTAMP="2007-10-01 11:30:00" RAWREAD
27 <READING TIMESTAMP="2007-10-01 12:00:00" RAWREAD
28 <READING TIMESTAMP="2007-10-01 12:30:00" RAWREAD
29 <READING TIMESTAMP="2007-10-01 13:00:00" RAWREAD
30 <READING TIMESTAMP="2007-10-01 13:30:00" RAWREAD
31 <READING TIMESTAMP="2007-10-01 14:00:00" RAWREAD
32 <READING TIMESTAMP="2007-10-01 14:30:00" RAWREAD
33 <READING TIMESTAMP="2007-10-01 15:00:00" RAWREAD
34 <READING TIMESTAMP="2007-10-01 15:30:00" RAWREAD
35 <READING TIMESTAMP="2007-10-01 16:00:00" RAWREAD
36 <READING TIMESTAMP="2007-10-01 16:30:00" RAWREAD
37 <READING TIMESTAMP="2007-10-01 17:00:00" RAWREAD
38 <READING TIMESTAMP="2007-10-01 17:30:00" RAWREAD
39 <READING TIMESTAMP="2007-10-01 18:00:00" RAWREAD
40 <READING TIMESTAMP="2007-10-01 18:30:00" RAWREAD
  
```

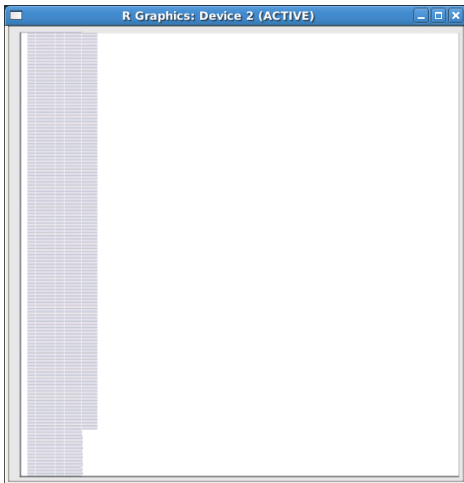
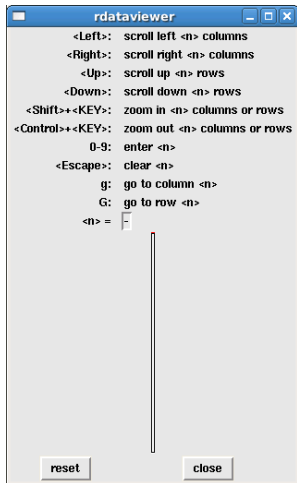

Zooming (Ctrl-PgUp Ctrl-PgUp Ctrl-PgUp)

Regular pattern discernable (stable values)



Zooming (PgDn x7)

Change in pattern discernable



Navigation (200000 G)

Instant navigation to middle of file.

rdataviewer
_ □ ×

<Left>: scroll left <n> columns
 <Right>: scroll right <n> columns
 <Up>: scroll up <n> rows
 <Down>: scroll down <n> rows
 <Shift>+<KEY>: zoom in <n> columns or rows
 <Control>+<KEY>: zoom out <n> columns or rows
 0-9: enter <n>
 <Escape>: clear <n>
 g: go to column <n>
 G: go to row <n>
 <n> =

reset
close

R Graphics: Device 2 (ACTIVE)
_ □ ×

```

//msc:/Fov/gnu002/7913w/Temp000/View1/MSB000000000000_001_01146_000
19999977 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999978 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999979 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999980 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999981 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999982 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999983 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999984 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999985 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999986 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999987 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999988 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999989 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999990 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999991 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999992 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999993 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999994 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999995 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999996 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999997 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999998 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
19999999 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*
20000000 *SHELLS VTIMEOUT**2007-12-16 17:00:00* BARRACUDA%0_000*

```

The ViewerDataMySQL class

Methods for viewing a (MySQL) databases query:

```
result <- dbGetQuery("SELECT * FROM (<query>) LIMIT 1")
```

- How many rows and columns?

```
SELECT COUNT(*) FROM (<query>);
```

- How wide is each column?

```
dbColumnInfo(result)$len
```

- What are the column names?

```
dbColumnInfo(result)$name
```

- Provide a text representation of given rows and columns

```
SELECT columns FROM (<query>) LIMIT rows;
```

The Ensembl data set

- MySQL database `homo_sapiens_core_46_36h` containing human genome.
- `seq_region` table contains 54695 “sequence regions”.

seq_region_id	name	coord_system_id	length
143909	AADC01095577.1.1.41877	4	41877
143910	AADD01112371.1.1.20303	4	20303
143911	AADD01115518.1.1.7924	4	7924
143912	AADD01116787.1.1.8237	4	8237
143913	AADD01116788.1.1.31207	4	31207
143914	AADD01116830.1.1.32292	4	32292
143915	AADD01117999.1.1.9773	4	9773
143916	AADD01118406.1.1.26766	4	26766
143917	AADD01118410.1.1.20289	4	20289
143918	AADD01172788.1.1.9996	4	9996
...

```
data <- viewerDataMySQL("select * from seq_region",
                        "homo_sapiens_core_46_36h",
                        "anonymous",
                        host="ensemldb.ensembl.org")
```

The Ensembl data set

rdataviewer

<Left>: scroll left <n> columns
 <Right>: scroll right <n> columns
 <Up>: scroll up <n> rows
 <Down>: scroll down <n> rows
 <Shift>+<KEY>: zoom in <n> columns or rows
 <Control>+<KEY>: zoom out <n> columns or rows
 0-9: enter <n>
 <Escape>: clear <n>
 g: go to column <n>
 G: go to row <n>
 <n> =

reset close

R Graphics: Device 2 (ACTIVE)

	seq_region_id	name	co
1	143909	AADC01095577.1.1.41877	
2	143910	AADD0112371.1.1.20303	
3	143911	AADD01115518.1.1.7924	
4	143912	AADD01116787.1.1.8237	
5	143913	AADD01116788.1.1.31207	
6	143914	AADD01116830.1.1.32292	
7	143915	AADD01117999.1.1.9773	
8	143916	AADD01118406.1.1.26766	
9	143917	AADD01118410.1.1.20289	
10	143918	AADD01172788.1.1.1.9996	
11	143919	AADD01172789.1.1.1.10593	
12	143920	AADD01172902.1.1.1.16221	
13	143921	AADD01209098.1.1.1.15883	
14	143922	AB001517.1.1.1.43051	
15	143923	AB001523.1.1.1.122638	
16	143924	AB015752.1.1.1.116160	
17	143925	AB017651.2.1.1.131668	
18	143926	AB019437.1.1.1.200000	
19	143927	AB019438.1.1.1.200000	
20	143928	AB019439.1.1.1.200000	
21	143929	AB019440.1.1.1.200000	
22	143930	AB019441.1.1.1.157090	
23	143931	AB020862.1.1.1.100000	
24	143932	AB020863.1.1.1.156909	
25	143933	AB126081.1.1.1.27543	
26	143934	AC000003.1.1.1.122228	
27	143935	AC000015.2.1.1.66704	
28	143936	AC000026.3.1.1.126312	
29	143937	AC000029.17.1.1.142979	
30	143938	AC000031.6.1.1.38703	
31	143939	AC000032.7.1.1.30140	
32	143940	AC000034.3.1.1.43738	
33	143941	AC000035.2.1.1.38429	
34	143942	AC000036.5.1.1.43847	
35	143943	AC000041.2.1.1.45436	
36	143944	AC000050.22.1.1.42167	
37	143945	AC000055.1.1.1.93578	
38	143946	AC000056.1.1.1.84838	
39	143947	AC000057.1.1.1.78631	
40	143948	AC000058.1.1.1.46419	

Zoom out (2 Ctrl-Right)

View all columns at once.

rdataviewer

<Left>: scroll left <n> columns
 <Right>: scroll right <n> columns
 <Up>: scroll up <n> rows
 <Down>: scroll down <n> rows
 <Shift>+<KEY>: zoom in <n> columns or rows
 <Control>+<KEY>: zoom out <n> columns or rows
 0-9: enter <n>
 <Escape>: clear <n>
 g: go to column <n>
 G: go to row <n>
 <n> =

reset
close

R Graphics: Device 2 (ACTIVE)

i_region_id	name	coord_system_id	length
1	143909	AADC01095577.1.1.41877	4
2	143910	AADD0112371.1.1.20303	4
3	143911	ADD01115518.1.1.7924	4
4	143912	AADD01136787.1.1.8237	4
5	143913	AADD0116788.1.1.31207	4
6	143914	AADD0116830.1.1.32292	4
7	143915	ADD01117989.1.1.8773	4
8	143916	AADD0118406.1.1.26766	4
9	143917	AADD0118410.1.1.20289	4
10	143918	AADD01172788.1.1.9996	4
11	143919	AADD01172789.1.1.10593	4
12	143920	AADD01172902.1.1.16221	4
13	143921	AADD01209098.1.1.15883	4
14	143922	AB001517.1.1.43051	4
15	143923	AB001523.1.1.122638	4
16	143924	AB015752.1.1.116160	4
17	143925	AB017651.2.1.131668	4
18	143926	AB019437.1.1.200000	4
19	143927	AB019438.1.1.200000	4
20	143928	AB019439.1.1.200000	4
21	143929	AB019440.1.1.200000	4
22	143930	AB019441.1.1.157090	4
23	143931	AB020862.1.1.100000	4
24	143932	AB020863.1.1.156909	4
25	143933	AB126081.1.1.27543	4
26	143934	AC000003.1.1.122228	4
27	143935	AC000025.2.1.66704	4
28	143936	AC000026.3.1.126312	4
29	143937	AC000029.17.1.142979	4
30	143938	AC000031.6.1.38703	4
31	143939	AC000032.7.1.30140	4
32	143940	AC000034.3.1.43738	4
33	143941	AC000035.2.1.38429	4
34	143942	AC000036.5.1.43847	4
35	143943	AC000041.2.1.45436	4
36	143944	AC000050.22.1.42167	4
37	143945	AC000055.1.1.93578	4
38	143946	AC000056.1.1.84638	4
39	143947	AC000057.1.1.78631	4
40	143948	AC000058.1.1.46419	4
41	143949	AC000059.2.1.107298	4
42	143950	AC000061.1.1.82512	4
43	143951	AC000063.1.1.34478	4
44	143952	AC000065.1.1.82261	4
45	143953	AC000066.1.1.94960	4
46	143954	AC000067.2.1.40645	4
47	143955	AC000068.2.1.43934	4
48	143956	AC000070.2.1.38699	4
49	143957	AC000076.2.1.41091	4
50	143958	AC000077.2.1.35739	4
51	143959	AC000078.2.1.44426	4
52	143960	AC000079.1.1.39540	4
53	143961	AC000080.2.1.44090	4

Navigation (20000 G)

Navigate to middle of query (with only small delay).

rdataviewer

<Left>: scroll left <n> columns
 <Right>: scroll right <n> columns
 <Up>: scroll up <n> rows
 <Down>: scroll down <n> rows
 <Shift>+<KEY>: zoom in <n> columns or rows
 <Control>+<KEY>: zoom out <n> columns or rows
 0-9: enter <n>
 <Escape>: clear <n>
 g: go to column <n>
 G: go to row <n>
 <n> =

R Graphics: Device 2 (ACTIVE)

i_region_id	name	coord_system_id	length
20000	163909	299290.1.1.77183	4
20001	163910	299291.1.1.96608	4
20002	163951	299297.1.1.100997	4
20003	163912	299495.1.1.151213	4
20004	163913	299496.1.1.150290	4
20005	163914	299497.1.1.106148	4
20006	163915	299569.1.1.80730	4
20007	163916	299570.2.1.66395	4
20008	163917	299571.2.1.104002	4
20009	163918	299572.1.1.106571	4
20010	163919	299704.1.1.7963	4
20011	163920	299706.1.1.14543	4
20012	163921	299714.2.1.87646	4
20013	163922	299715.1.1.52086	4
20014	163923	299716.4.1.220895	4
20015	163924	299754.1.1.22029	4
20016	163925	299755.1.1.98274	4
20017	163926	299756.7.1.81961	4
20018	163927	299758.8.1.140788	4
20019	163928	299774.1.1.73239	4
20020	163929	299786.1.1.102543	4
20021	163930	299943.1.1.128230	4
20022	163931	AC007463.3.1.166892	4
20023	163932	AC007464.4.1.186187	4
20024	163933	AC007465.4.1.170305	4
20025	163934	AC007481.6.1.137032	4
20026	163935	AC007483.7.1.176174	4
20027	163936	AC007485.10.1.164733	4
20028	163937	AC007486.1.1.174725	4
20029	163938	AC007487.2.1.23179	4
20030	163939	AC007488.15.1.146443	4
20031	163940	AC007489.3.1.180401	4
20032	163941	AC007490.7.1.169207	4
20033	163942	AC007491.7.1.180888	4
20034	163943	AC007493.7.1.155461	4
20035	163944	AC007494.8.1.206262	4
20036	163945	AC007495.9.1.198575	4
20037	163946	AC007496.6.1.198417	4
20038	163947	AC007497.6.1.164269	4
20039	163948	AC007500.7.1.220562	4
20040	163949	AC007501.2.1.162617	4
20041	163950	AC007502.6.1.158696	4
20042	163951	AC007510.6.1.156420	4
20043	163952	AC007511.8.1.168822	4
20044	163953	AC007513.40.1.147118	4
20045	163954	AC007514.5.1.177720	4
20046	163955	AC007527.6.1.186692	4
20047	163956	AC007528.5.1.188451	4
20048	163957	AC007529.5.1.147239	4
20049	163958	AC007533.2.1.153053	4
20050	163959	AC007535.3.1.161339	4
20051	163960	AC007536.10.1.71819	4
20052	163961	AC007537.3.1.190858	4
----	----	----	----

Summary

- The novel navigation ideas don't really work; they would probably just scare off the students.
- Being able to zoom is a small improvement on most basic viewers.
- Seeing the “shape” of the entire data structure is quite nice.
- It's fun to be able to navigate quickly around a massive file.
- The modular structure of the code encourages further development.

Available from <http://www.stat.auckland.ac.nz/~paul/>