

# The Pviz User Guide

Renan Sauteraud\*

July 18, 2012

## Contents

<b>1</b>	<b>Example of plot</b>	<b>2</b>
<b>2</b>	<b>Using the extended coordinates system</b>	<b>2</b>
2.1	Data formating . . . . .	2
2.2	Plotting with both coordinates system . . . . .	3

---

\*rsautera@fhcrc.org

## 1 Example of plot

```
> library(HIV.db)
> library(Pviz)
> ##### Get reference scale from alignment
> alignObj<-readAlign()
> refScale<-alignObj[[1]]
> refSeq<-alignObj[[2]]
> ##### Load database
> HIV_db<-loadFeatures(ref="env")
> envBase<-getFeature(HIV_db)
> envStart=getHXB2Coordinates(envBase)[1,][1]
> envEnd=getHXB2Coordinates(envBase)[1,][2]
> ##### Getting Features from database for annotation
> proteins<-getFeature(HIV_db,category="protein",start=envStart,end=envEnd,frame=getFrame(
> antis<-getEpitope(envBase,name=c("VRC01"))
> helix<-getChildren(envBase,category=c("helix"))

> ## ProteinAxisTrack using the extended coordinates system
> rpxet<-ProteinAxisTrack(littleTicks=TRUE)
> ## ProteinAxisTrack with coordinates relative to the reference
> rpref<-ProteinAxisTrack(refScale=refScale, adNC=TRUE)

> sTrack<-SequenceTrack(refSeq)

> data(pepMicroarrayEx)
> p1Track<-ProbeTrack(pepMicroarrayEx$probeSeq, pepMicroarrayEx$probeFreq, pepMicroarrayEx$
+               , protein="gp120", name="sequence(B)")

> a2Track<-ATrack(id=proteins@values@unlistData@listData[["name"]],start=start(proteins),e
> a3Track<-ATrack(id=helix@values@unlistData@listData[["name"]],start=start(helix),end=end
> a6Track<-ATrack(id=antis@values@unlistData@listData[["name"]],start=start(antis),end=end

> data(pepExprEx)
> library(IRanges)
> d6Track<-DTrack(range=IRanges(start=pepExprEx$dPos,width=1),groups=rownames(pepExprEx$dE
> plotTracks(trackList=c(rpext,rpref,sTrack,a2Track,a3Track,a6Track,p1Track,d6Track), from
```

## 2 Using the extended coordinates system

The extended coordinate system is based on a multiple alignment. It is a scale from 0 to the length of the alignment (i.e: reference sequence + gaps).

The refScale is a scale used to translate coordinates between the extended system and the normal system (which is based uniquely on the reference sequence length).

### 2.1 Data formatting

In the following examples, hxb2 is the reference sequence and is aligned with different subtypes.

First, get the refScale:

```
> alignObj<-readAlign(filename=system.file("extdata/alignment.fasta", package="Pviz"))
> refScale<-alignObj[[1]]
> refSeq<-alignObj[[2]]
```

Now, the refScale can be used to translate coordinates into the extended system.

Example of loading HIV\_db using extended coordinates.

```
> library(HIV.db)
> HIV_db<-loadFeatures(ref="env", refScale=refScale)
> envBase<-getFeature(HIV_db)
```

The positions in envBase are the ones observed in the alignment.

To translate all the positions in pep\_hxb2, use convertPep().

```
> data(pep_hxb2)
> nrd<-convertPep(rd=pep_hxb2)
```

nrd is pep\_hxb2 with updated ranges, aligned and trimmed columns.

To convert coordinates into the extended system, use coord2ext() with the refScale defined earlier:

```
> start<-coord2ext(c(200,450), ref=refScale)
> start
```

```
[1] 223 474
```

## 2.2 Plotting with both coordinates system

All objects used to create tracks should be in extended coordinate system.

```
> sTrack<-SequenceTrack(refSeq)
> pax1<-ProteinAxisTrack()
> pax2<-ProteinAxisTrack(refScale=refScale, col.gap="blue")
> proteins<-getFeature(HIV_db,category="protein",start=envStart,end=envEnd,frame=getFrame())
> aTrack<-ATrack(id=proteins@values@unlistData@listData$name,start=start(proteins),end=end(proteins))
> aTrack<-ATrack(id=helix@values@unlistData@listData$name,start=start(helix),end=end(helix))
> plotTracks(c(pax1,pax2,sTrack,aTrack), from=1, to=160)
```

The first axis track displays the extended scale, while the second displays the reference coordinates, it also shows the gaps in the reference sequence with respect of the alignment.