

BioJava Version 2

Matthew Pocock,
School of Computer Sciences,
Univerisity of Newcastle Upon Tyne

New Bioinformatics Toys
to give you an Easier Life

Why Re-write BioJava?



- Bioinformatics is 10% infrastructure and 90% munging
- BioJava 1 is good for that 10%, but not so good for the 90%
 - Memory-efficient & Robust
 - Verbose & Steep learning-curve
- Most limits on the functionality of BioJava 1 are due to the design, not the implementation
 - Sequence/Feature Object Model, Genomics-Centric
- Java 5.0 has loads of goodies e.g. `@Annotation` and generics that make developing and using libraries easier



Attractive things about BJV2

- Simple APIs for Simple Jobs
 - Utility facades for all day-to-day tasks
 - Fine-grained APIs for power users
- Highly modular - SPI framework to make things 'just work'
 - Grab just the jars you need, learn just those APIs
 - Drop in extra jars to your classpath, and they 'glue in'
 - File-Format Guessing, Database Drivers, etc.

More attractive things about BJV2

- Documentation is central
 - Full and –lite Javadocs
 - Cook-book examples written as we go
 - Web-site built by ant
- Strong support for Data Integration and Federation
- Automatically expose any bjev2 features as services
 - Web services, CORBA, Taverna processors, ...

Example of Feature Data Projection

```
interface Munged {  
    int min; int max;  
    Strand strand;  
    FeatureType type;  
}
```

*the data-structure we
want to work with*

load each sequence in a file

```
for(Sequence seq : SeqIO.readSequence(seqFile)) {  
    List<Munged> munged = Projections.project(  
        Munged.class, Feature.class, seq.getFeatures(),  
        "min", "min(locations anchors min)",  
        "max", "max(locations anchors max)",  
        "strand", "locations anchors strand",  
        "type", "type");
```

*Project from Feature
properties to Munged
properties*

```
Follows.sort(munged, "min");  
for(Munged m: munged)  
    System.out.println(  
        m.getMin() + "\t" + m.getMax() + "\t" +  
        m.getStrand() + "\t" + m.getType());
```

sort and print

Life So Far

A decorative graphic at the top of the slide consists of six circles. The first two circles are positioned behind the text 'Life So Far'. The first circle is solid light purple, and the second is an outline. To the right of the text, there are three more circles: a solid light purple circle, an outline circle, and another solid light purple circle.

- Started coding at Easter
 - primarily me coding
 - ~360 files, ~30k lines
- In Subversion
 - <http://www.derkholm.net/svn/repos/bjv2/trunk>
- Web-Site
 - <http://bjv2.derkholm.net/>
- Currently IGPL
 - up for discussion

Thanks To...



- Thomas Down (svn hosting)
- Everyone on [irc.freenode.net #biojava](https://irc.freenode.net/#biojava)
- All the guys on [biojava-l](#) and [biojava-dev](#)
- Open-Bio
- Anil Wipat, and the rest at School of Computing Sciences, Newcastle