javanaise

Fast, robust and easy to use system for shared, distributed and duplicated Java classes
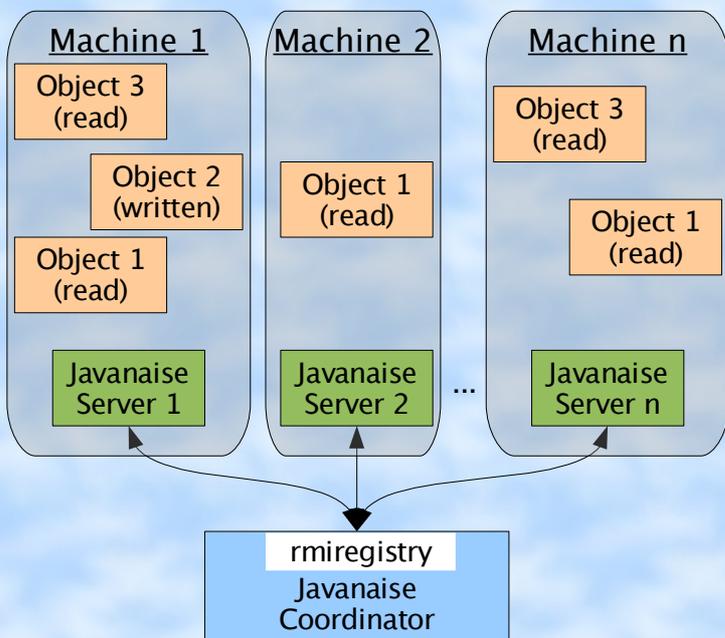
What can Javanaise do for me ?

Thanks to Javanaise, existing Java applications can create and use shared Java objects. Since it uses caching mechanisms, Javanaise is extremely efficient for applications that read objects more than they write.
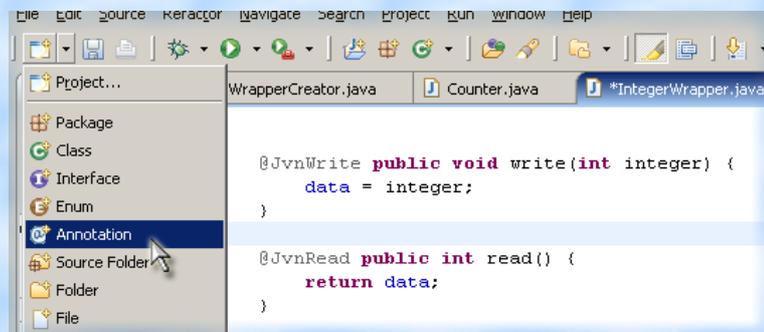
How does Javanaise work?

To obtain a reference to a Java object, you would call "new" and use the Java Virtual Machine. To obtain a reference to a Javanaise Object, programs use the Javanaise Server. The Javanaise Coordinator synchronizes all servers so all references to Javanaise Objects are always up to date.

Javanaise Servers and Javanaise Objects can be added to or removed from the system at any time, and the system has been extensively tested in heavy load conditions. Javanaise uses standard Java RMI calls to share any Serializable Java object.
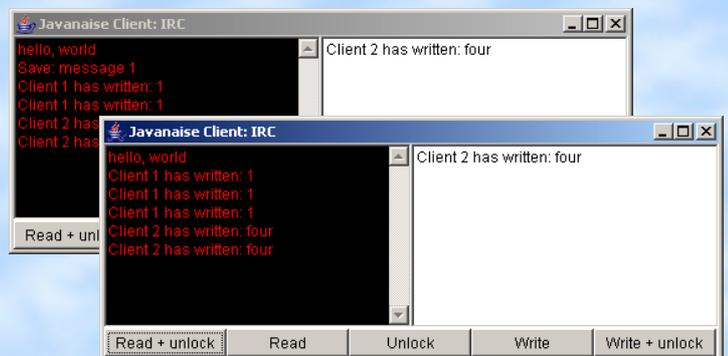
How do I use Javanaise?

For rapid and easy development, you simply put annotations at the beginning of the constructor and shared methods of the objects that you want to share. The Javanaise API will automatically connect to the Server and Coordinator and synchronize those objects.


```
@JvnWrite public void write(int integer) {
    data = integer;
}

@JvnRead public int read() {
    return data;
}
```

One who wishes to have an advanced control over the Javanaise API can also manually access the Javanaise Objects and tailor the way the those objects will be locked, unlocked and therefore synchronized.

I want Javanaise!

To get your copy of Javanaise, together with a detailed manual and usage examples, simply contact us:
http://ali.tokmen.com and
http://lionel.debroux.free.fr