[Xstream之常用方式与常用注解](http://blog.csdn.net/gufeng672/article/details/24907747)

2014-05-03 10:36 1430人阅读 [评论](http://blog.csdn.net/gufeng672/article/details/24907747#comments)(2) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/gufeng672/article/details/24907747#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

WEB开发（4） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

参考资料   
1 xStream框架完美实现Java对象和xml文档JSON、XML相互转换   
<http://www.cnblogs.com/hoojo/archive/2011/04/22/2025197.html>   
2 xStream完美转换XML、JSON   
<http://archive.cnblogs.com/a/2025197/>   
官网:http://xstream.codehaus.org/download.html   
相关jar包可在:   
XStream之初识   
<http://liuzidong.iteye.com/blog/1059453>下载   
示例代码

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. Blog teamBlog = **new** Blog(**new** Author("Guilherme Silveira"));
2. teamBlog.add(**new** Entry("first","My first blog entry."));
3. eamBlog.add(**new** Entry("tutorial","Today we have developed a nice alias tutorial. Tell your friends! NOW!"));
4. XStream xstream = **new** XStream();
5. System.out.println(xstream.toXML(teamBlog));

**[java]**    
打印结果为:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. <xtream.Blog>
2. <writer>
3. <name>Guilherme Silveira</name>
4. </writer>
5. <entries>
6. <xtream.Entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </xtream.Entry>
10. <xtream.Entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </xtream.Entry>
14. </entries>
15. </xtream.Blog>

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. <xtream.Blog>
2. <writer>
3. <name>Guilherme Silveira</name>
4. </writer>
5. <entries>
6. <xtream.Entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </xtream.Entry>
10. <xtream.Entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </xtream.Entry>
14. </entries>
15. </xtream.Blog>

以上节点就包含了包名,如包名很长,就很难看了,怎样才能重新命名这个节点呀！   
以下的1,2设置效果一样

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. 1 //xstream.aliasPackage("", "xtream");
2. 2 xstream.alias("blog", Blog.**class**);
3. 3 xstream.alias("entry", Entry.**class**);

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. 1 //xstream.aliasPackage("", "xtream");
2. 2 xstream.alias("blog", Blog.**class**);
3. 3 xstream.alias("entry", Entry.**class**);

通过这样设置就完成了节点名称的设置.如下:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. <blog>
2. <writer>
3. <name>Guilherme Silveira</name>
4. </writer>
5. <entries>
6. <entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </entry>
10. <entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </entry>
14. </entries>
15. </blog>

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. <blog>
2. <writer>
3. <name>Guilherme Silveira</name>
4. </writer>
5. <entries>
6. <entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </entry>
10. <entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </entry>
14. </entries>
15. </blog>

修改子节点属性名称   
将writer属性名称修改为:autor

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.aliasField("author", Blog.**class**, "writer");

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.aliasField("author", Blog.**class**, "writer");

输出如下:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. <blog>
2. <author>
3. <name>Guilherme Silveira</name>
4. </author>
5. <entries>
6. <entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </entry>
10. <entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </entry>
14. </entries>
15. </blog>

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. <blog>
2. <author>
3. <name>Guilherme Silveira</name>
4. </author>
5. <entries>
6. <entry>
7. <title>first</title>
8. <description>My first blog entry.</description>
9. </entry>
10. <entry>
11. <title>tutorial</title>
12. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
13. </entry>
14. </entries>
15. </blog>

删除集合节点名称的设置

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.addImplicitCollection(Blog.**class**, "entries");

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.addImplicitCollection(Blog.**class**, "entries");

输出如下:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. <blog>
2. <author>
3. <name>Guilherme Silveira</name>
4. </author>
5. <entry>
6. <title>first</title>
7. <description>My first blog entry.</description>
8. </entry>
9. <entry>
10. <title>tutorial</title>
11. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
12. </entry>
13. </blog>

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. <blog>
2. <author>
3. <name>Guilherme Silveira</name>
4. </author>
5. <entry>
6. <title>first</title>
7. <description>My first blog entry.</description>
8. </entry>
9. <entry>
10. <title>tutorial</title>
11. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
12. </entry>
13. </blog>

将Blog类的元素设置为:它的节点属性   
//使用这个属性名作为节点上的元素

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.useAttributeFor(Blog.**class**, "writer");

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.useAttributeFor(Blog.**class**, "writer");

//重新命名

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.aliasField("author", Blog.**class**, "writer");

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.aliasField("author", Blog.**class**, "writer");

//注册转换器

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.registerConverter(**new** AuthorConverter());

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.registerConverter(**new** AuthorConverter());

转换器:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. **import** com.thoughtworks.xstream.converters.SingleValueConverter;
2. **class** AuthorConverter **implements** SingleValueConverter {
3. //显示节点属性值
4. **public** String toString(Object obj) {
5. **return** ((Author) obj).getName();
6. }
8. **public** Object fromString(String name) {
9. **return** **new** Author(name);
10. }
12. **public** **boolean** canConvert(Class type) {
13. **return** type.equals(Author.**class**);
14. }
15. }

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. **import** com.thoughtworks.xstream.converters.SingleValueConverter;
2. **class** AuthorConverter **implements** SingleValueConverter {
3. //显示节点属性值
4. **public** String toString(Object obj) {
5. **return** ((Author) obj).getName();
6. }
8. **public** Object fromString(String name) {
9. **return** **new** Author(name);
10. }
12. **public** **boolean** canConvert(Class type) {
13. **return** type.equals(Author.**class**);
14. }
15. }

显示结果:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. <blog author="Guilherme Silveira">
2. <entry>
3. <title>first</title>
4. <description>My first blog entry.</description>
5. </entry>
6. <entry>
7. <title>tutorial</title>
8. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
9. </entry>
10. </blog>

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. <blog author="Guilherme Silveira">
2. <entry>
3. <title>first</title>
4. <description>My first blog entry.</description>
5. </entry>
6. <entry>
7. <title>tutorial</title>
8. <description>Today we have developed a nice alias tutorial. Tell your friends! NOW!</description>
9. </entry>
10. </blog>

注解的使用方式,使用之前必须加上注解类才有作用:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. XStream xstream = **new** XStream();
2. xstream.processAnnotations(Blog.**class**);
3. xstream.processAnnotations(Entry.**class**);

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. XStream xstream = **new** XStream();
2. xstream.processAnnotations(Blog.**class**);
3. xstream.processAnnotations(Entry.**class**);

1  @XStreamAlias注解可在类与属性上使用设置名称,相当于: xstream.alias("blog", Blog.class);

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. @XStreamAlias("blog")
2. **public** **class** Blog {
4. @XStreamAlias("author")
5. **private** Author writer;
7. .....
8. }

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. @XStreamAlias("blog")
2. **public** **class** Blog {
4. @XStreamAlias("author")
5. **private** Author writer;
7. .....
8. }

当然Entry类也要设置同样的类属性:@XStreamAlias("entity")   
2 @XStreamImplicit去集合节点名:相当于 xstream.addImplicitCollection(Blog.class, "entries");   
3 @XStreamConverter(AuthorConverter.class),参见以上的转换器类相当于:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. xstream.useAttributeFor(Blog.**class**, "writer");
2. //重新命名
3. xstream.aliasField("author", Blog.**class**, "writer");
4. //注册转换器
5. xstream.registerConverter(**new** AuthorConverter());

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. xstream.useAttributeFor(Blog.**class**, "writer");
2. //重新命名
3. xstream.aliasField("author", Blog.**class**, "writer");
4. //注册转换器
5. xstream.registerConverter(**new** AuthorConverter());

要使用这个注解AuthorConverter必须符合二个条件   
a 必须要有个默认的无参构造函数

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. **public** AuthorConverter() {
3. }

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. **public** AuthorConverter() {
3. }

b 类声明必须为:public

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. **public** **class** AuthorConverter **implements** SingleValueConverter {
2. ...
3. }

**[java]** [view plain](http://blog.csdn.net/gufeng672/article/details/24907747) [copy](http://blog.csdn.net/gufeng672/article/details/24907747)

[在CODE上查看代码片](https://code.csdn.net/snippets/324188)

1. **public** **class** AuthorConverter **implements** SingleValueConverter {
2. ...
3. }

不用注解这二点都可以不强制要求!   
完整代码如下:

Java代码 [复制代码](http://liuzidong.iteye.com/blog/1059861) [[收藏代码http://liuzidong.iteye.com/images/spinner.gif](http://blog.csdn.net/gufeng672/article/details/24907747)](http://blog.csdn.net/gufeng672/article/details/24907747)

1. **import** com.thoughtworks.xstream.XStream;
2. **import** com.thoughtworks.xstream.annotations.XStreamAlias;
3. **import** com.thoughtworks.xstream.annotations.XStreamAsAttribute;
4. **import** com.thoughtworks.xstream.annotations.XStreamConverter;
5. **import** com.thoughtworks.xstream.annotations.XStreamImplicit;
6. @XStreamAlias("blog")
7. **public** **class** Blog {
9. @XStreamAsAttribute
10. @XStreamAlias("author")
11. @XStreamConverter(AuthorConverter.**class**)
12. **private** Author writer;
14. @XStreamImplicit
15. **private** List entries = **new** ArrayList();
17. **public** Blog(Author writer) {
18. **this**.writer = writer;
19. }
21. **public** **void** add(Entry entry) {
22. entries.add(entry);
23. }
25. **public** List getContent() {
26. **return** entries;
27. }
29. **public** **static** **void** main(String[] args) {
31. Blog teamBlog = **new** Blog(**new** Author("Guilherme Silveira"));
32. teamBlog.add(**new** Entry("first", "My first blog entry."));
33. teamBlog
34. .add(**new** Entry("tutorial",
35. "Today we have developed a nice alias tutorial. Tell your friends! NOW!"));
36. XStream xstream = **new** XStream();
37. xstream.processAnnotations(Blog.**class**);
38. xstream.processAnnotations(Entry.**class**);
39. // 重新命名节点名
40. // xstream.aliasPackage("", "xtream");
41. /\*
42. \* xstream.alias("blog", Blog.class); xstream.alias("entry",
43. \* Entry.class); //重新命名属性名 // xstream.aliasField("author", Blog.class,
44. \* "writer"); //去节点 xstream.addImplicitCollection(Blog.class,
45. \* "entries"); // xstream.useAttributeFor(Blog.class, "writer"); //
46. \* xstream.aliasField("author", Blog.class, "writer"); //
47. \* xstream.addImplicitCollection(Blog.class, "entries");
48. \* //使用这个属性名作为节点上的元素 xstream.useAttributeFor(Blog.class, "writer");
49. \* //重新命名 xstream.aliasField("author", Blog.class, "writer"); //注册转换器
50. \* xstream.registerConverter(new AuthorConverter());
51. \*/
52. System.out.println(xstream.toXML(teamBlog));
53. }
55. }