

```
package edu.wsu.gridstat.subscriber.interfaces;
```

```
/**  
 * <p>Title: HolderBooleanInterface</p>  
 * <p>Description: </p>  
 * <p>Copyright: Copyright (c) 2002</p>  
 * <p>Company: Washington State University</p>  
 * @author Kjell Harald Gjermundrod  
 * @version 1.0  
 */
```

```
public interface HolderBooleanInterface extends HolderBaseInterface
```

```
{  
 /**  
 * The <code>getValue</code> method is used to return the current value of  
 * this variable.  
 * <BR>  
 * @return Returns the current value of this variable.  
 */  
 public boolean getValue();  
  
 /**  
 * The <code>setValue</code> method is used to set the value of this variable.  
 * <BR>  
 * @param value The new value of the variable.  
 * @param sentTime The time when this value was sent/created.  
 * @param receivedTime The time that we received this value.  
 * @return Return <code>>true</code> if value set, <code>>false</code> otherwise  
 */  
 public boolean setValue(boolean value, long sentTime, long receivedTime);  
}
```

```
package edu.wsu.gridstat.subscriber.interfaces;
```

```
/**  
 * <p>Title: HolderUserDefinedInterface</p>  
 * <p>Description: </p>  
 * <p>Copyright: Copyright (c) 2002</p>  
 * <p>Company: Washington State University</p>  
 * @author Kjell Harald Gjermundrod  
 * @version 1.0  
 */
```

```
// Standard java packages.  
import java.nio.ByteBuffer;
```

```
public abstract interface HolderUserDefinedInterface extends HolderBaseInterface
```

```
{  
 /**  
 * The <code>setValue</code> method is used to set the value of this variable.  
 * <BR>  
 * @param packet The buffers where the event is encoded.  
 * @param startIdx The index of which of the buffers that this event start.  
 * @param len The number of buffers that belong to this event.  
 * @param sentTime The time when this value was sent/created.  
 * @param receivedTime The time that we received this value.  
 * @param extrapolated If this value is extrapolated or not.  
 * @return Return <code>>true</code> if value set, <code>>false</code> otherwise  
 */  
 public boolean setValue(ByteBuffer[] packet, int startIdx, int len, long sentTime, long receivedTime, boolean extrapolated);  
  
 /**  
 * The <code>setValue</code> method is used to set the value of this variable.  
 * <BR>  
 * @param value The value of the event received.  
 * @param sentTime The time when this value was sent/created.  
 * @param receivedTime The time that we received this value.  
 * @param extrapolated If this value is extrapolated or not.  
 * @return Return <code>>true</code> if value set, <code>>false</code> otherwise  
 */  
 public boolean setValue(byte[] value, long sentTime, long receivedTime, boolean extrapolated);  
  
 /**  
 * The <code>getLength</code> method is used to return the length, number of bytes" of the datatype of the variable.  
 * <BR>  
 * @return Return the number of bytes of the datatype.  
 */  
 public int getLength();  
  
 /**  
 * The <code>getRealType</code> method is used to return the real data type of this variable.  
 * <BR>  
 * @return Returns the real type of this variable.  
 */  
 public int getRealType();  
}
```