

XPath

Directive that evaluates xpath against org.w3c.dom.Node by Vladimir Kirichenko <vladimir.kirichenko#gmail.com>

```
package org.apache.velocity.runtime.directive;

import org.apache.velocity.context.InternalContextAdapter;
import org.apache.velocity.exception.MethodInvocationException;
import org.apache.velocity.exception.ParseErrorException;
import org.apache.velocity.exception.ResourceNotFoundException;
import org.apache.velocity.runtime.directive.Directive;
import org.apache.velocity.runtime.parser.node.Node;
import org.apache.xpath.XPathAPI;

import javax.xml.transform.TransformerException;
import java.io.IOException;
import java.io.Writer;

public class XPath extends Directive {
    public String getName() {
        return "xpath";
    }

    public int getType() {
        return LINE;
    }

    public boolean render( InternalContextAdapter context, Writer writer, Node node ) throws IOException,
ResourceNotFoundException, ParseErrorException, MethodInvocationException {
        String var = node.jjtGetChild( 0 ).getFirstToken().image.substring( 1 );
        org.w3c.dom.Node document = ( org.w3c.dom.Node ) node.jjtGetChild( 1 ).value( context );
        String xpath = String.valueOf( node.jjtGetChild( 2 ).value( context ) );

        try {
            context.put( var, XPathAPI.eval( document, xpath ) );
        } catch( TransformerException e ) {
            throw new IOException( "cannot evaluate xpath: " + e.getMessage() );
        }
        return true;
    }
}
```

Example:

```
#xpath($result $doc '/my/x/path')
```