

UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering



Script2Ant - a tool for cross-platform scripting between different operating systems

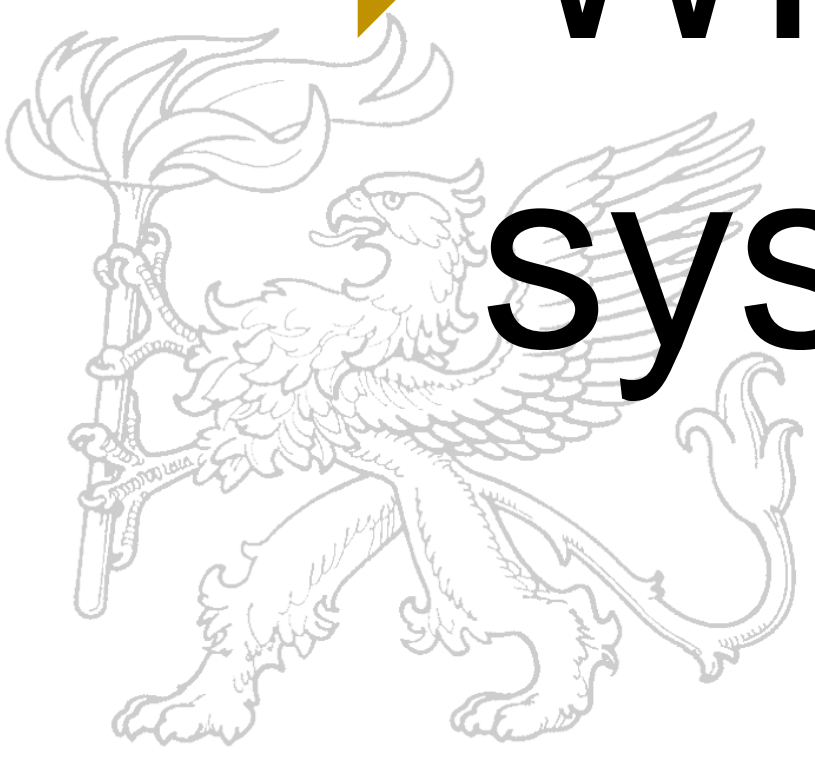


György Hegedűs
Lajos Jenő Fülöp
Rudolf Ferenc



Legacy scripts

- ▶ How to use legacy operating system scripts on other platforms?
- ▶ Earlier solutions
 - Cygwin
 - Wine
- ▶ When we should use legacy operating system scripts directly on other platforms?



Motivation

- ▶ Test scripts
 - Production binaries will not run in Wine or Cygwin
 - Chance for false results
- ▶ Install and other scripts
 - *“Dear Customer, you have to install Wine to install my application...”*
 - *“I don’t want to install Wine...”*



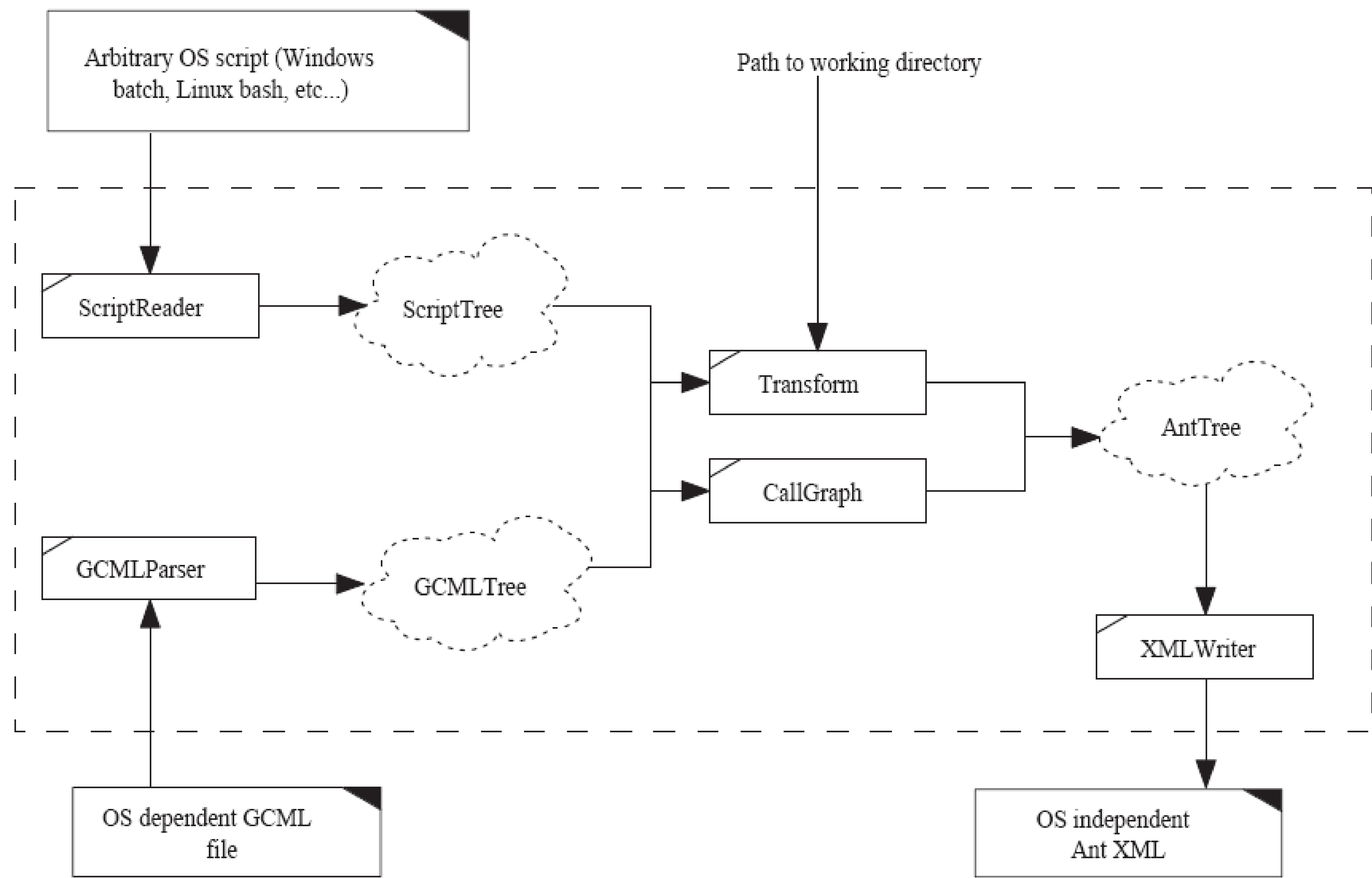
Script2Ant

- ▶ Need to convert scripts between different OS
- ▶ Script2Ant is the first step
 - platform independent solution
 - converting scripts into platform independent form





Architecture



GCML

- ▶ *General Command Mapping Language*
- ▶ XML-format
- ▶ It assigns an Ant task to each language-dependent command





GCML DTD

```
<!ELEMENT language (command*, methoddef*)>
<!ELEMENT methoddef (sign*)>
<!ELEMENT sign EMPTY>
<!ELEMENT command (parameter*)>
<!ELEMENT parameter (attribute*)>
<!ELEMENT attribute (attribute*)>

<!ATTLIST language
  id      ID      #REQUIRED
  name    CDATA  #REQUIRED>

<!ATTLIST methoddef
  id      ID      #REQUIRED>

<!ATTLIST sign
  type    ( open | close ) "open"
  sign    CDATA  #REQUIRED
  need    ( true | false ) "true">

<!ATTLIST command
  id      ID      #REQUIRED
  name    CDATA  #REQUIRED
  antname CDATA  #REQUIRED
  regexp  CDATA  #IMPLIED>

<!ATTLIST parameter
  optional      ( true | false ) "true"
  optionalName  CDATA  #IMPLIED>

<!ATTLIST attribute
  type    ( attribute | tag ) "attribute"
  antname CDATA  #REQUIRED
  value   ( this | next | prev | null | all ) "this"
  fixvalue CDATA  #IMPLIED
  ifContains CDATA  #IMPLIED
  ifNotContains CDATA  #IMPLIED>
```

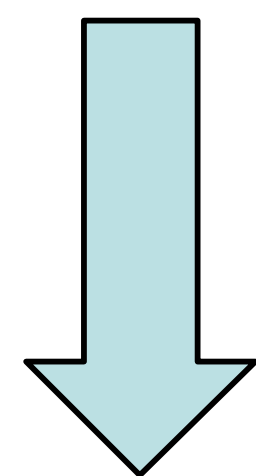
GCML example

```
<command name="copy" antname="copy" id="0020">  
  <parameter optional="false">  
    <attribute type="attribute" antname="file" value="this" ifContains=".">  
      <attribute antname="overwrite" value="fixed" fixvalue="true"/>  
    </attribute>  
    <attribute type="tag" antname="fileset dir" value="this" ifNotContains=".">  
  </parameter>  
<parameter optional="false">  
  <attribute type="attribute" antname="tofile" value="this" ifContains="."/>  
  <attribute type="attribute" antname="todir" value="this" ifNotContains="."/>  
</parameter>
```

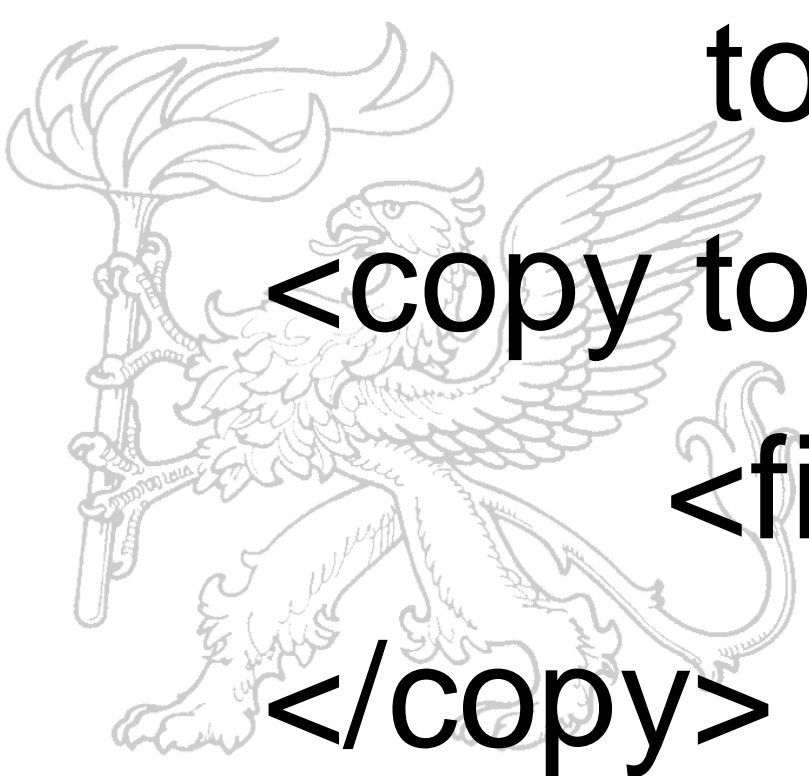


Conversion example

```
copy "D:\Work\bin\Release\Windows\Script2Ant.cfg" Test\bin
copy "D:\Work\bin\Release\Windows\Script2Ant.cfg" Test\bin\Script2Ant
copy D:\Work\bin\Release\Windows Test\bin
```



```
<copy file="${WorkDir}\bin\Release\Windows\Script2Ant.cfg" overwrite="true"
  todir="Test\bin"/>
<copy file="${WorkDir}\bin\Release\Windows\Script2Ant.cfg" overwrite="true"
  tofile="Test\bin\Script2Ant"/>
<copy todir="Test\bin">
  <fileset dir="${WorkDir}\bin\Release\Windows"/>
</copy>
```



Case study

- ▶ conversion of a Regression Test
- ▶ Columbus, LIM2Metrics
- ▶ Input is a binary file (ASG)
- ▶ Output is a CSV file (metrics)
- ▶ Regression test
- Compare CSV files with previous stable version



Main method

```
echo "Lim2Metrics Regression Test System"  
set TestDir=Test  
set ReferenceDir=%TestDir%\Reference  
set ResultDir=%TestDir%\Result  
set BinDir=%TestDir%\bin  
set BUILDDIR=Build  
set INPUTDIR=Input  
set LOGDIR=log  
set LOGFILE=%LOGDIR%\Lim2Metrics.log  
set ProgramsLOG=..\..\LOGS  
echo [ Lim2Metrics Regression Test System ] > %LOGFILE%  
echo " " >> %LOGFILE%  
sleep 5  
call :makeenvironment  
call :copyresources  
call :test  
call :diffingresults  
call :clean  
exit
```

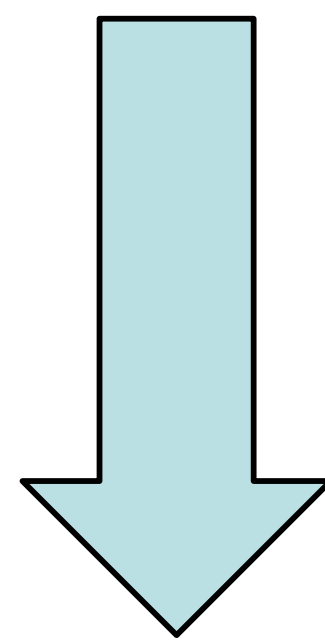


Converted main method

```
<target name="main">
  <echo message="Lim2Metrics Regression Test System"/>
  <property name="TestDir" value="Test"/>
  <property name="ReferenceDir" value="${TestDir}\Reference"/>
  <property name="ResultDir" value="${TestDir}\Result"/>
  <property name="BinDir" value="${TestDir}\bin"/>
  <property name="BUILDDIR" value="Build"/>
  <property name="INPUTDIR" value="Input"/>
  <property name="LOGDIR" value="log"/>
  <property name="LOGFILE" value="${LOGDIR}\Lim2Metrics.log"/>
  <property name="ProgramsLOG" value="..\..\LOGS"/>
  <echo file="${LOGFILE}" message="[ Lim2Metrics Regression Test System ]"/>
  <echo file="${LOGFILE}" append="true" message=" "/>
  <sleep milliseconds="5"/>
  <antcall target=":makeenvironment"/>
  <antcall target=":copyresources"/>
  <antcall target=":test"/>
  <antcall target=":diffingresults"/>
  <antcall target=":clean"/>
</target>
```

Conversion of clean method

```
:clean  
echo " Copy Lim2Metrics.log" >> %LOGFILE%  
copy "%LOGDIR%\Lim2Metrics.log" %ProgramsLOG%  
echo " Delete license.txt" >> %LOGFILE%  
del /F /S /Q "%BinDir%\license.txt"  
sleep 5  
goto:eof
```



```
<target name=":clean">  
  <echo file="{LOGFILE}" append="true" message=" Copy Lim2Metrics.log"/>  
  <copy file="{LOGDIR}\Lim2Metrics.log" overwrite="true"  
    todir="{ProgramsLOG}"/>  
  <echo file="{LOGFILE}" append="true" message=" Delete license.txt"/>  
  <delete quiet="true" file="{BinDir}\license.txt"/>  
  <sleep seconds="5"/>  
</target>
```





Performance

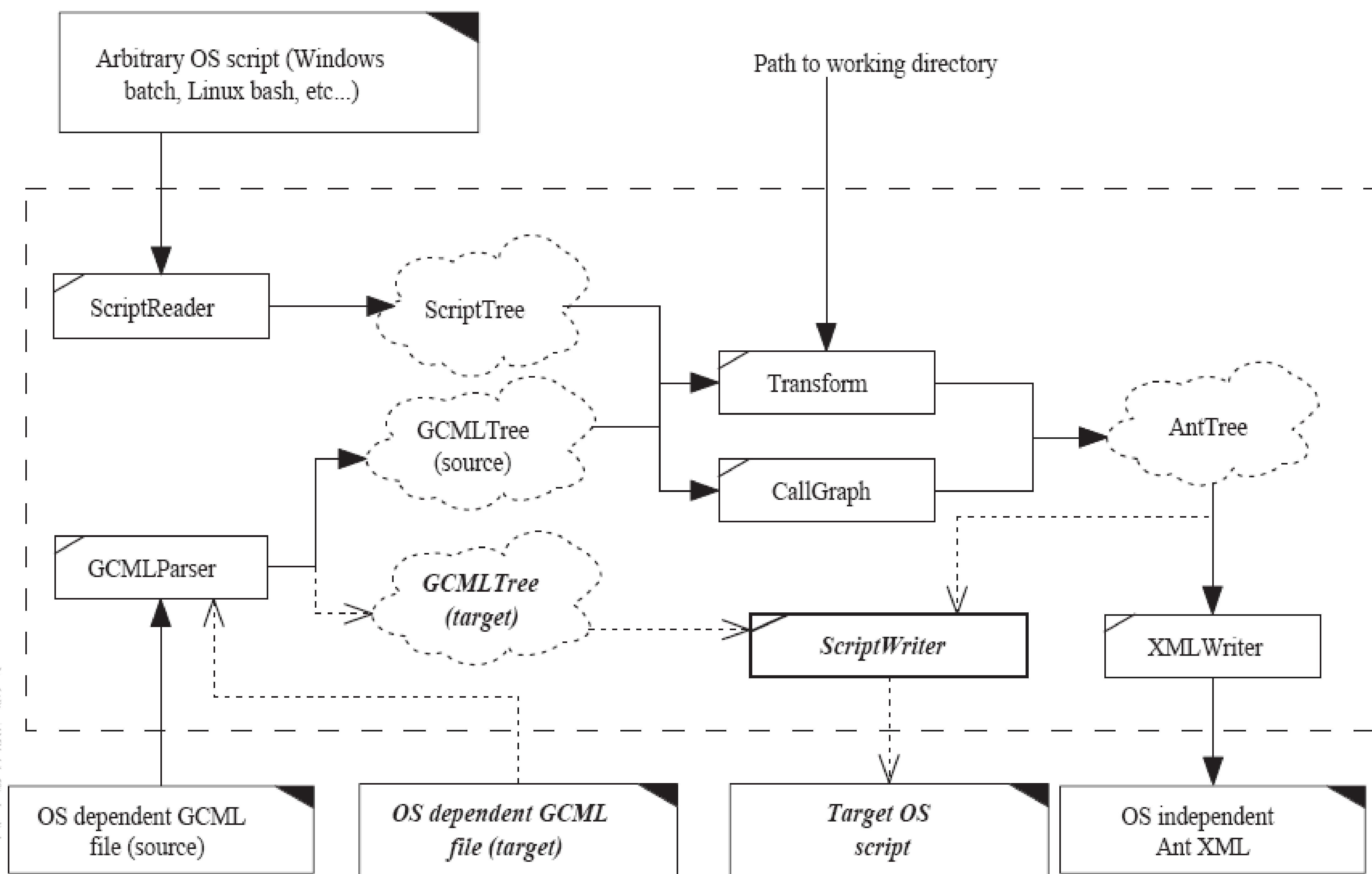
- ▶ Running certain commands in increasing number of iterations

Iterations	10		50		100		1000		10000	
Command	Unix	Ant	Unix	Ant	Unix	Ant	Unix	Ant	Unix	Ant
echo	0,01	0,03	0,01	0,04	0,01	0,13	0,01	0,45	0,17	8,25
mkdir	0,01	0,03	0,03	0,04	0,06	0,08	0,63	0,54	6,80	7,43
rm	0,01	0,09	0,02	0,08	0,04	0,08	0,46	0,56	4,89	7,11
mv	0,01	0,09	0,03	0,09	0,07	0,09	0,72	0,71	7,40	8,11
rmdir	0,01	0,08	0,02	0,08	0,04	0,16	0,44	0,60	4,49	8,08

Iterations	10		50		100		1000		10000	
Command	Win	Ant	Win	Ant	Win	Ant	Win	Ant	Win	Ant
echo	0,03	0,04	0,07	0,05	0,15	0,14	0,70	0,46	5,82	8,26
md	0,03	0,08	0,18	0,23	0,45	2,13	5,12	5,91	57,34	52,68
del	0,04	0,10	0,07	0,09	0,18	0,85	2,23	2,54	23,14	19,82
move	0,03	0,11	0,09	0,10	0,46	2,23	6,02	6,82	59,42	60,82
rd	0,03	0,09	0,06	0,11	0,19	1,37	2,44	2,91	24,19	26,59



Future work



Conclusion

- ▶ Prototype of Script2Ant
- ▶ Converting OS-dependent scripts to Ant XML files
- ▶ GCML
 - adding a new OS means only writing the related GCML file

