



Open Source XML Applications

24 September 2009



Open Source Application Development

Norman Walsh

Mark Logic Corporation



Licensed under a Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported License

www.xmlsummerschool.com

Learning Objectives

- Obtain an overview of the landscape of open source application development tools
- Survey a few of the XML libraries
- Learn how to build an end-to-end XML application with open source

The examples in this presentation are biased towards open source tools and are intended to be illustrative, not exhaustive. In every category, there are bound to be other choices. No disrespect is intended.

Contents

- Open source tools
- XML Libraries
- Examples
- Building your own



Open Source Tools

What is open source?

- Open source can mean many things:
 - <http://opensource.org/docs/osd>
 - Source code
 - Free redistribution
 - Derived works
 - “Free as in speech”
 - “Free as in beer”
- I’m agnostic
 - Free as in speech is OK
 - Free as in beer is OK
 - Commercial is OK too!

What is application development?

- The implementation of some idea in software (or hardware)
- What do you need?
 - A platform on which to implement it
 - A programming language (or languages)
 - An editor (or editing environment)
 - A source code control system
 - Some way to run it
 - Some way to test it
 - Some way to deploy it

Pick a platform

- Linux / BSD / Solaris
- Mac OS X (BSD)
- Windows
 - Cygwin
- There are others: Android, iPhone, PS/3...
- Many programming languages, editing environments, and other tools are available for more than one platform

Pick a programming language

- Bash
- C#
- C/C++
- Groovy
- Java
- JavaScript
- PHP
- Perl
- Python
- Ruby

- XProc
- XQuery
- XSLT

http://en.wikipedia.org/wiki/Categorical_list_of_programming_languages

Pick an editor

- Text editors
 - Emacs
 - Vi
 - NotePad
- Integrated Development Environments
 - Eclipse
 - NetBeans

Pick a source code control system

- Centralized systems
 - ~~CVS~~
 - Subversion
- Decentralized (peer-to-peer)
 - Git
 - Mercurial
 - Bazaar

What is a source code control system?

- A sanity preservation device
- Incredibly useful if you're working by yourself
- Absolutely necessary if you're working in a team

What is a source code control system? (2)

- A tool for helping manage changes to a common repository:
 - Make sure Bob doesn't accidentally overwrite Mary's changes
- A tool that answers important questions:
 - How is my version of the file different from the version currently in production?
 - How is the production version today different from the production version of last Thursday?
 - Exactly which files *were* in production last Thursday?

Version control vocabulary

- Branches
 - Different versions being developed in parallel
- Tags
 - A specific, identified version of the repository
- Trunk or Head
 - The current “main” branch
- Check in (or Commit) / Check out
 - Put into/copy out of the repository
- Merge
 - Combining changes from multiple users



Pick an environment to host your application

- Native applications
 - Command line
 - GUI
- The LAMP stack
 - Linux
 - Apache
 - MySQL
 - Perl/PHP/Python
- Ruby on Rails (Ruby)
- Django (Python)
- Catalyst (Perl)
- Glassfish (Java)



Testing your application

- Test frameworks vary by programming language and development environment
- Many focus on unit testing
- Test early, test often

Deploying your application

- Generally determined by the hosting environment
- Often useful to have separate environments
 - Test/development
 - Quality assurance
 - Production



XML Libraries

Stream or object model or...?

- Streaming
 - Push
 - Pull
- Object Model
 - XML data models
 - Native language data models
- Never underestimate the power of a regular expression

Streaming push or pull?

- Push
 - SAX
- Pull
 - Cursor
 - StAX, XmlReader
 - XML event objects
 - StAX

Streaming push

```
myHandler = new ContentHandler()  
parser = new Parser()  
parser.parse(xmlsource, myHandler)
```

```
Class ContentHandler {  
    method startDocument()  
    method startElement()  
    method characters()  
    ...  
}
```

Streaming pull

```
parser = new Parser(xmlsource)
```

```
while (parser.hasMoreEvents()) {  
    Event event = parser.getNextEvent()  
    if (event.type == START_DOCUMENT) {...}  
    elseif (event.type == START_ELEMENT) {...}  
    elseif (event.type == CHARACTERS) {...}
```

Object model: XML or native?

- XML data models
 - DOM
 - JDOM
 - XOM
 - XDM
- Native language
 - Hashes
 - Arrays
 - Etc.

- Streaming
 - XML::SAX, XML::SAX2Perl
- XML object model
 - XML::DOM, XML::DOM2
- Native object model
 - XML::Simple
- Application interfaces
 - XML::LibXML, XML::LibXSLT, XML::Xerces
 - App::XML::DocBook, RPC::XML, XML::Atom

- Streaming
 - Sax
- XML object model
 - MiniDom, PullDom
- Native object model
 - ElementTree, 4Suite, Amara
- Application Interfaces
 - Amara, lxml
 - PyXMPP, XmlRpc

- Streaming
 - `org.xml.sax.*`, `javax.xml.stream.*`
- XML object model
 - `org.w3c.dom.*`, `nu.xom.*`
- Native object model
 - JAXB
- Application Interfaces
 - Lots

- Streaming
 - SAX, XmlTextReader, XmlPull, XPathNavigator
- XML object model
 - XmlDocument, XPathDocument
- Native object model
 - XQuery?
- Application Interfaces
 - Lots



Examples

Examples

- A Perl application on a LAMP stack
- A command-line Java application
- An XQuery application

Perl on LAMP

- Linux
- Perl
- Emacs
- Subversion
- Apache
- Perl testing framework

Perl project

- Given an XML document containing geocoded photographs
- and a query that supplies a geo-spatial bounding box
- Return all the photographs inside that bounding box
- as a CGI-BIN script
- for use as a web service end point

Creating a subversion project

```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource
File Edit View Terminal Help
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ svnadmin create /subv
ersion/perl
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ mkdir perl perl/branch
es perl/tags perl/trunk
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ svn import perl file:
///subversion/perl -m "First import"
Adding          perl/trunk
Adding          perl/branches
Adding          perl/tags

Committed revision 1.
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ rm -rf perl
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ svn co file:///subver
sion/perl/trunk perl
Checked out revision 1.
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ ls -l perl
total 0
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$ ls -la perl
total 2
drwxr-xr-x 1 ndw ndw 102 2009-09-08 16:09 .
drwxr-xr-x 1 ndw ndw 306 2009-09-08 16:09 ..
drwxr-xr-x 1 ndw ndw 272 2009-09-08 16:09 .svn
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource$
```

Geocoded photographs

```

emacs@vbox
File Edit Options Buffers Tools SGML Help

<points>
<point lat='50.7051861111111' lon='-1.5031'
  url='http://farm2.static.flickr.com/1126/1402489877_afa79e8990_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Waste oil here</title>
</point>
<point lat='50.7103' lon='-1.50458333333333'
  url='http://farm2.static.flickr.com/1164/1403348448_1442a4a181_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Big ship on the Solent</title>
</point>
<point lat='50.7143472222222' lon='-1.50640555555556'
  url='http://farm2.static.flickr.com/1173/1402458091_a456c0fd2b_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Sailing toward the sunset on the Solent</title>
</point>
<point lat='50.7196055555556' lon='-1.493525'
  url='http://farm2.static.flickr.com/1189/1402456861_d1a1c45e69_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Sailing along</title>
</point>
<point lat='50.71985' lon='-1.49300555555556'
  url='http://farm2.static.flickr.com/1244/1403343974_5444574fa0_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Enjoying the sail</title>
</point>
<point lat='50.7204527777778' lon='-1.49185555555556'
  url='http://farm2.static.flickr.com/1182/1402453553_202765a83b_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Guy Reece</title>
</point>
<point lat='50.72785' lon='-1.48456944444444'
  url='http://farm2.static.flickr.com/1182/1402453553_202765a83b_s.jpg'>
  <by>Norman Walsh</by>
  <place>Freshwater, GB, ENG</place>
  <title>Guy Reece</title>
</point>
</points>
--:-- points.xml Top L5 (XML)

```


Perl project files

perl - File Browser

File Edit View Go Bookmarks Tabs Help

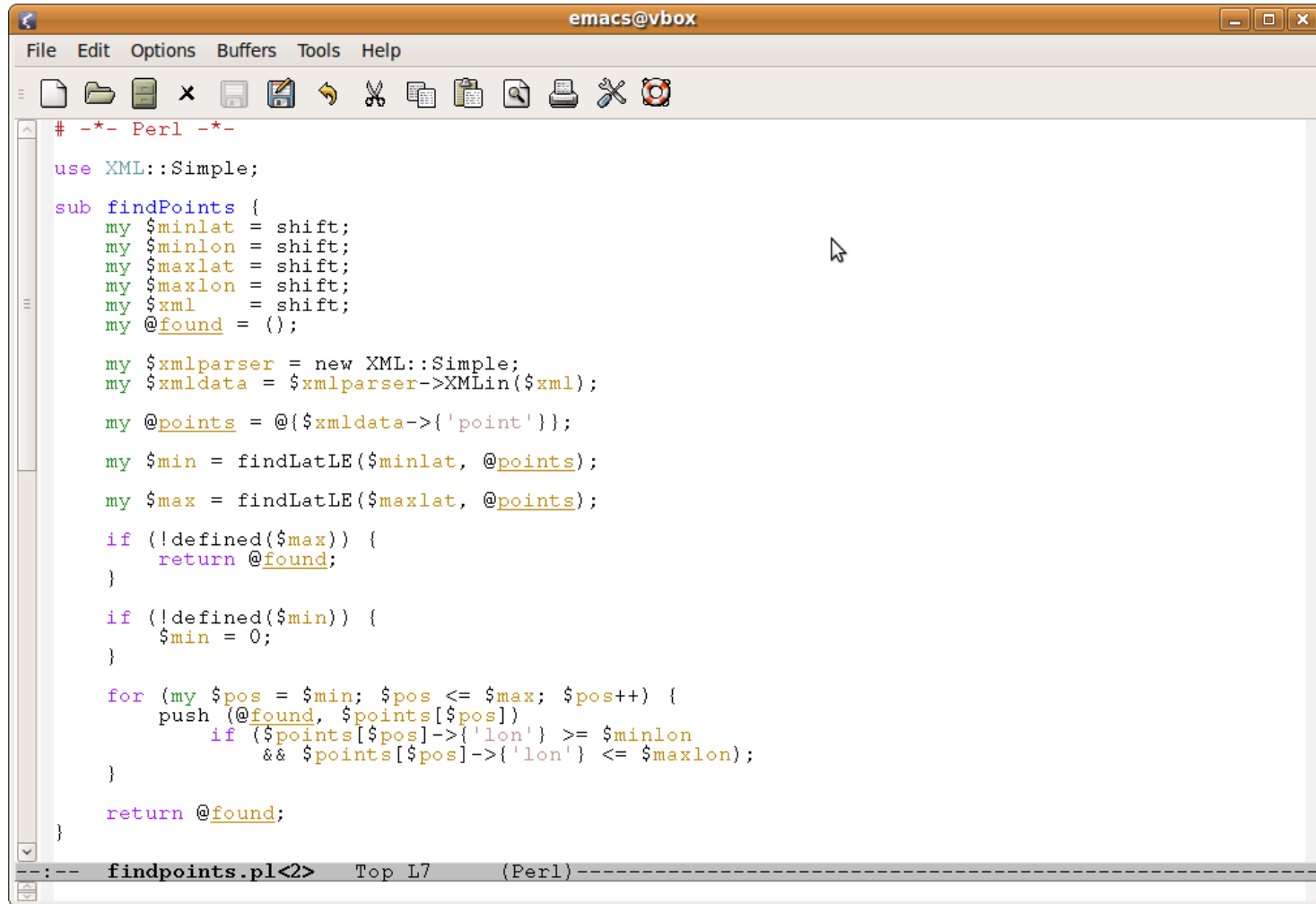
Back Forward Up Stop Reload Home Computer Search

presentations 2009 09-xmlls opensource perl 50% List View

Name	Size	Type	Date Modified
bin	1 item	folder	Tue 08 Sep 2009 01:29:49 PM EDT
findpics.pl	715 bytes	Perl script	Tue 08 Sep 2009 01:11:29 PM EDT
cgi-bin	1 item	folder	Tue 08 Sep 2009 01:24:16 PM EDT
findpics	1.2 KB	Perl script	Tue 08 Sep 2009 01:24:16 PM EDT
etc	1 item	folder	Tue 08 Sep 2009 01:02:28 PM EDT
points.xml	29.3 KB	XML document	Tue 08 Sep 2009 10:41:35 AM EDT
lib	1 item	folder	Tue 08 Sep 2009 01:30:00 PM EDT
findpoints.pl	1.6 KB	Perl script	Tue 08 Sep 2009 01:03:55 PM EDT
t	1 item	folder	Tue 08 Sep 2009 01:03:32 PM EDT
findtests.t	763 bytes	Troff document	Tue 08 Sep 2009 01:03:32 PM EDT
index.html	156 bytes	HTML document	Tue 08 Sep 2009 01:18:58 PM EDT
Makefile	70 bytes	Makefile	Tue 08 Sep 2009 01:03:21 PM EDT

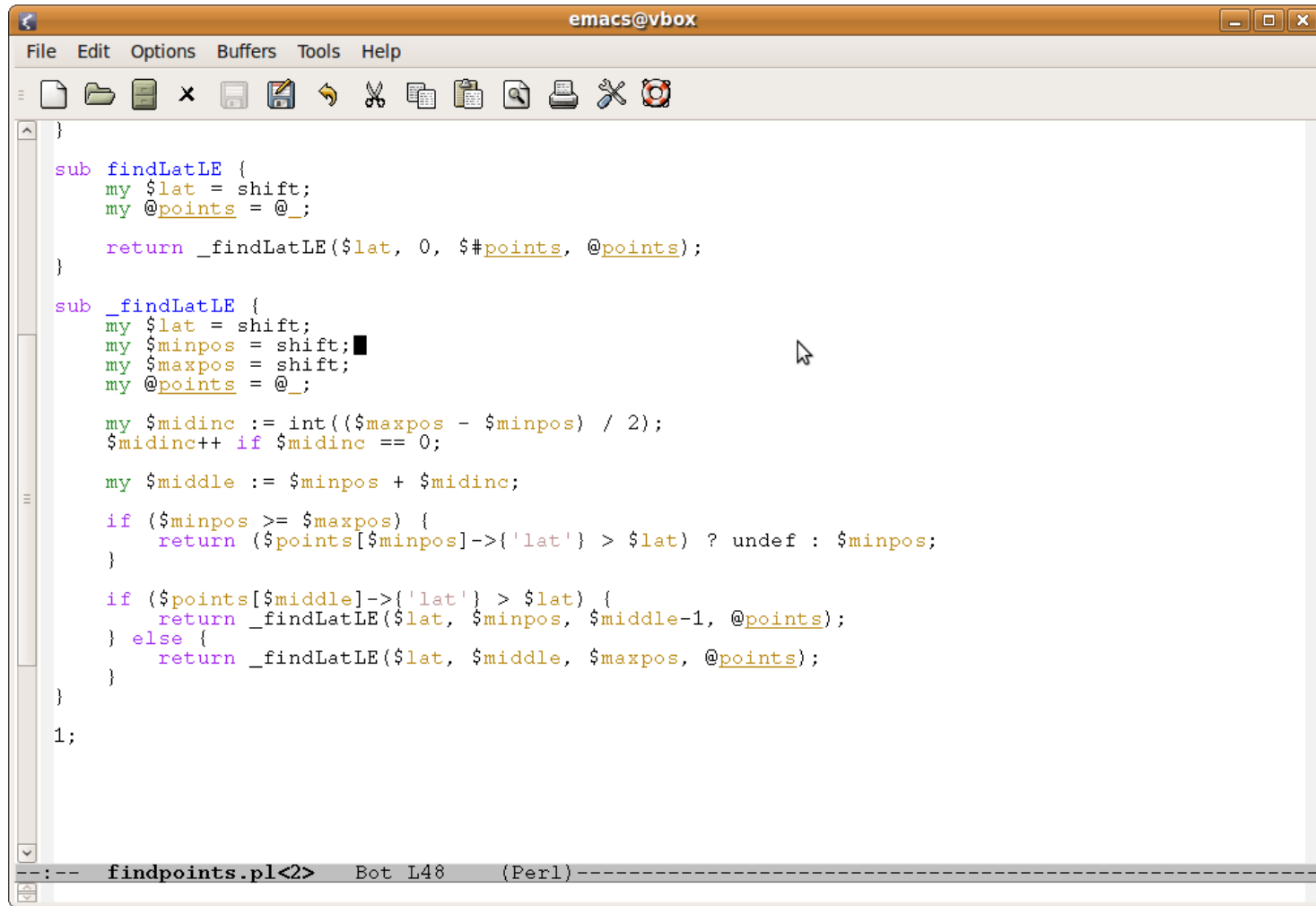
7 items, Free space: 28.7 GB

findpoints.pl, part 1



```
# -*- Perl -*-  
  
use XML::Simple;  
  
sub findPoints {  
    my $minlat = shift;  
    my $minlon = shift;  
    my $maxlat = shift;  
    my $maxlon = shift;  
    my $xml = shift;  
    my @found = ();  
  
    my $xmlparser = new XML::Simple;  
    my $xmldata = $xmlparser->XMLin($xml);  
  
    my @points = @{$xmldata->{'point'}};  
  
    my $min = findLatLE($minlat, @points);  
    my $max = findLatLE($maxlat, @points);  
  
    if (!defined($max)) {  
        return @found;  
    }  
  
    if (!defined($min)) {  
        $min = 0;  
    }  
  
    for (my $pos = $min; $pos <= $max; $pos++) {  
        push (@found, $points[$pos])  
            if ($points[$pos]->{'lon'} >= $minlon  
                && $points[$pos]->{'lon'} <= $maxlon);  
    }  
  
    return @found;  
}  
  
--:-- findpoints.pl<2>   Top L7   {Perl}-----
```

findpoints.pl, part 2



```
sub findLatLE {
    my $lat = shift;
    my @points = @_;

    return _findLatLE($lat, 0, $#points, @points);
}

sub _findLatLE {
    my $lat = shift;
    my $minpos = shift;
    my $maxpos = shift;
    my @points = @_;

    my $midinc := int(($maxpos - $minpos) / 2);
    $midinc++ if $midinc == 0;

    my $middle := $minpos + $midinc;

    if ($minpos >= $maxpos) {
        return ($points[$minpos]->{'lat'} > $lat) ? undef : $minpos;
    }

    if ($points[$middle]->{'lat'} > $lat) {
        return _findLatLE($lat, $minpos, $middle-1, @points);
    } else {
        return _findLatLE($lat, $middle, $maxpos, @points);
    }
}

1;
```

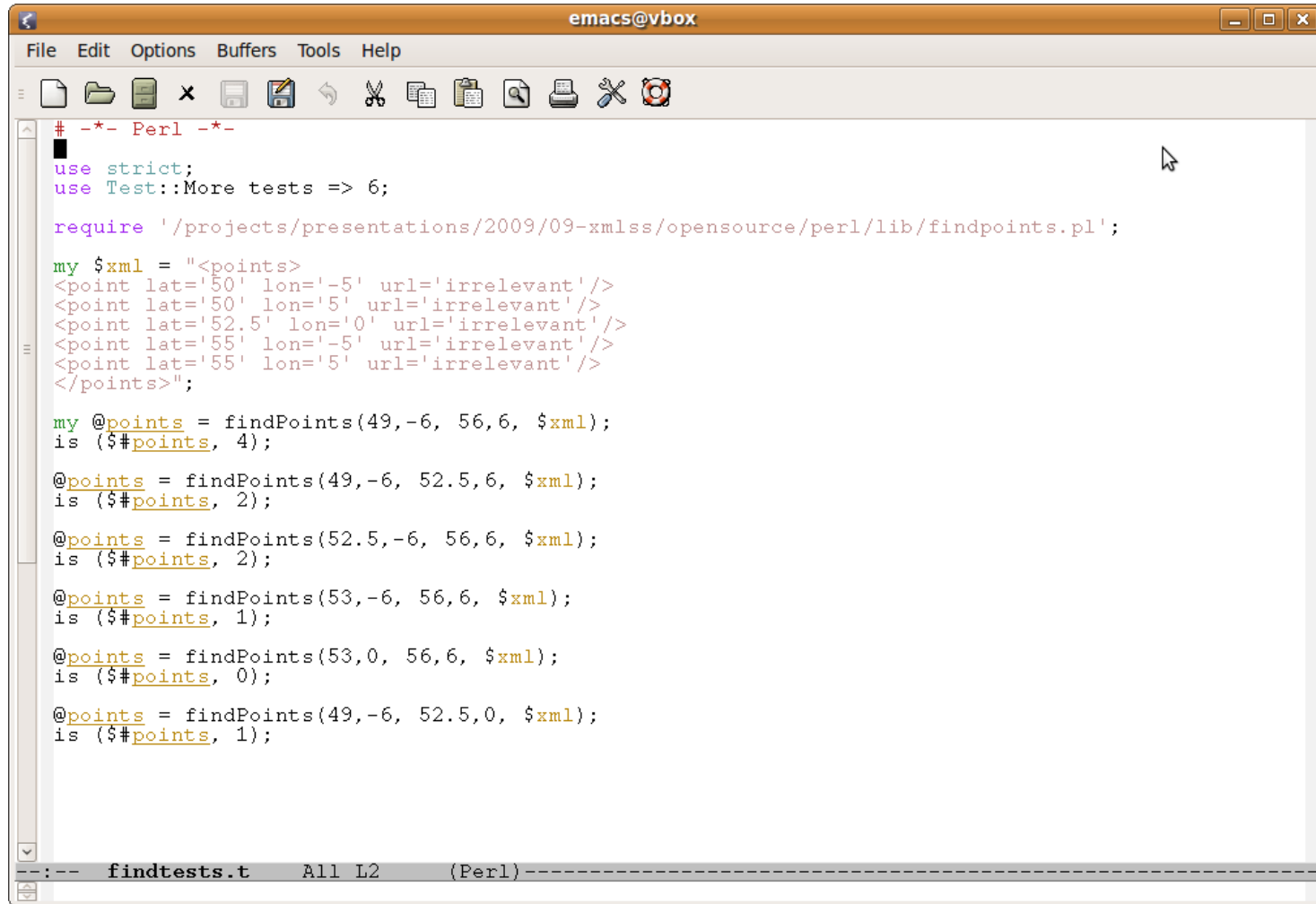
Add files to subversion repository

```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl
File Edit View Terminal Help
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$ svn status
?      t
?      lib
?      bin
?      cgi-bin
?      index.html
?      etc
?      Makefile
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$ svn add t lib bi
n cgi-bin index.html etc Makefile
A      t
A      t/findtests.t
A      lib
A      lib/findpoints.pl
A      bin
A      bin/findpics.pl
A      cgi-bin
A      cgi-bin/findpics
A      index.html
A      etc
A      etc/points.xml
A      Makefile
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$
```

Save project in subversion

```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl
File Edit View Terminal Help
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$ svn commit -m "Initial checkin"
Adding          Makefile
Adding          bin
Adding          bin/findpics.pl
Adding          cgi-bin
Adding          cgi-bin/findpics
Adding          etc
Adding          etc/points.xml
Adding          index.html
Adding          lib
Adding          lib/findpoints.pl
Adding          t
Adding          t/findtests.t
Transmitting file data .....
Committed revision 2.
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$
```

Perl test configuration



```
# -*- Perl -*-
use strict;
use Test::More tests => 6;

require '/projects/presentations/2009/09-xmlss/opensource/perl/lib/findpoints.pl';

my $xml = "<points>
<point lat='50' lon='-5' url='irrelevant'/>
<point lat='50' lon='5' url='irrelevant'/>
<point lat='52.5' lon='0' url='irrelevant'/>
<point lat='55' lon='-5' url='irrelevant'/>
<point lat='55' lon='5' url='irrelevant'/>
</points>";

my @points = findPoints(49,-6, 56,6, $xml);
is ($#points, 4);

@points = findPoints(49,-6, 52.5,6, $xml);
is ($#points, 2);

@points = findPoints(52.5,-6, 56,6, $xml);
is ($#points, 2);

@points = findPoints(53,-6, 56,6, $xml);
is ($#points, 1);

@points = findPoints(53,0, 56,6, $xml);
is ($#points, 0);

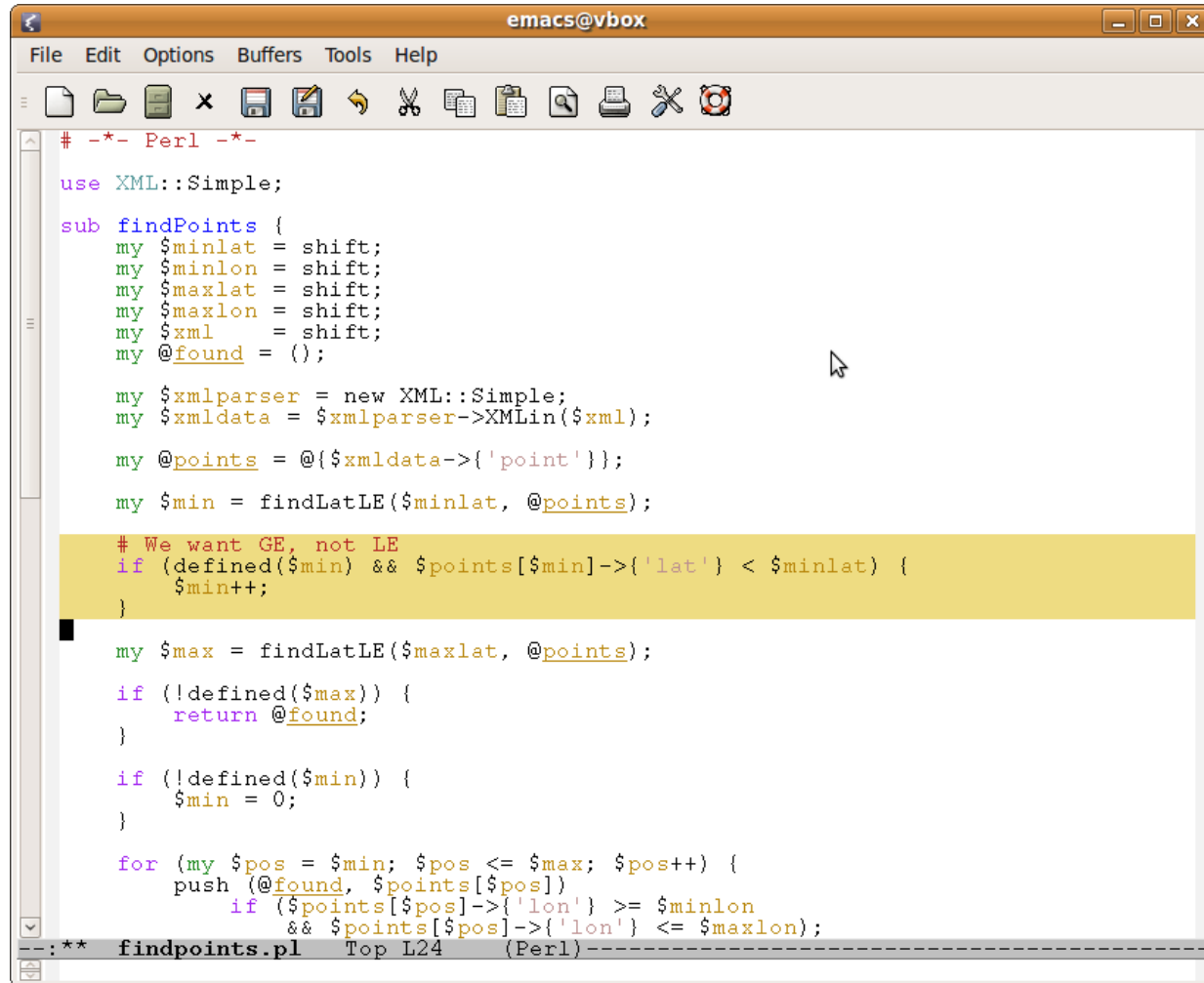
@points = findPoints(49,-6, 52.5,0, $xml);
is ($#points, 1);
```

findtests.t All L2 (Perl)

Perl test results (1)

```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl
File Edit View Terminal Help
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$ make
perl -MTest::Harness -e "runtests 't/findtests.t';"
t/findtests....NOK 4/6
#   Failed test at t/findtests.t line 26.
#       got: '2'
#   expected: '1'
t/findtests....NOK 5/6
#   Failed test at t/findtests.t line 29.
#       got: '1'
#   expected: '0'
# Looks like you failed 2 tests of 6.
t/findtests....dubious
    Test returned status 2 (wstat 512, 0x200)
DIED. FAILED tests 4-5
    Failed 2/6 tests, 66.67% okay
Failed Test   Stat Wstat Total Fail  List of Failed
-----
t/findtests.t    2   512     6     2  4-5
Failed 1/1 test scripts. 2/6 subtests failed.
Files=1, Tests=6,  1 wallclock secs ( 0.07 cusr +  0.27 csys =  0.34 CPU)
Failed 1/1 test programs. 2/6 subtests failed.
make: *** [test] Error 2
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$
```

Perl bug fix



```
# -*- Perl -*-

use XML::Simple;

sub findPoints {
    my $minlat = shift;
    my $minlon = shift;
    my $maxlat = shift;
    my $maxlon = shift;
    my $xml = shift;
    my @found = ();

    my $xmlparser = new XML::Simple;
    my $xmldata = $xmlparser->XMLin($xml);

    my @points = @{$xmldata->{'point'}};

    my $min = findLatLE($minlat, @points);

    # We want GE, not LE
    if (defined($min) && $points[$min]->{'lat'} < $minlat) {
        $min++;
    }

    my $max = findLatLE($maxlat, @points);

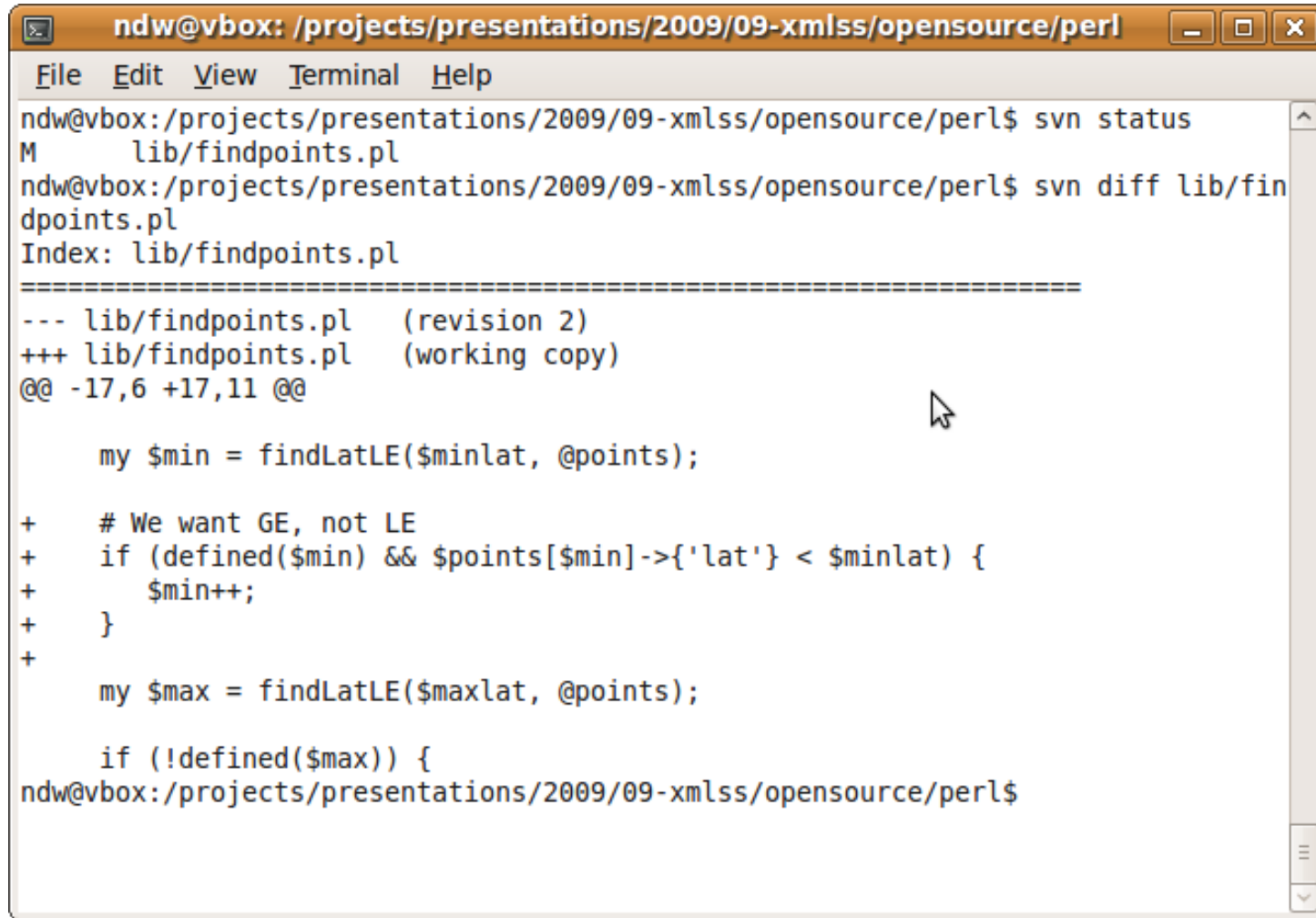
    if (!defined($max)) {
        return @found;
    }

    if (!defined($min)) {
        $min = 0;
    }

    for (my $pos = $min; $pos <= $max; $pos++) {
        push (@found, $points[$pos])
            if ($points[$pos]->{'lon'} >= $minlon
                && $points[$pos]->{'lon'} <= $maxlon);
    }
}

--:** findpoints.pl Top L24 (Perl)-----
```


Subversion diff



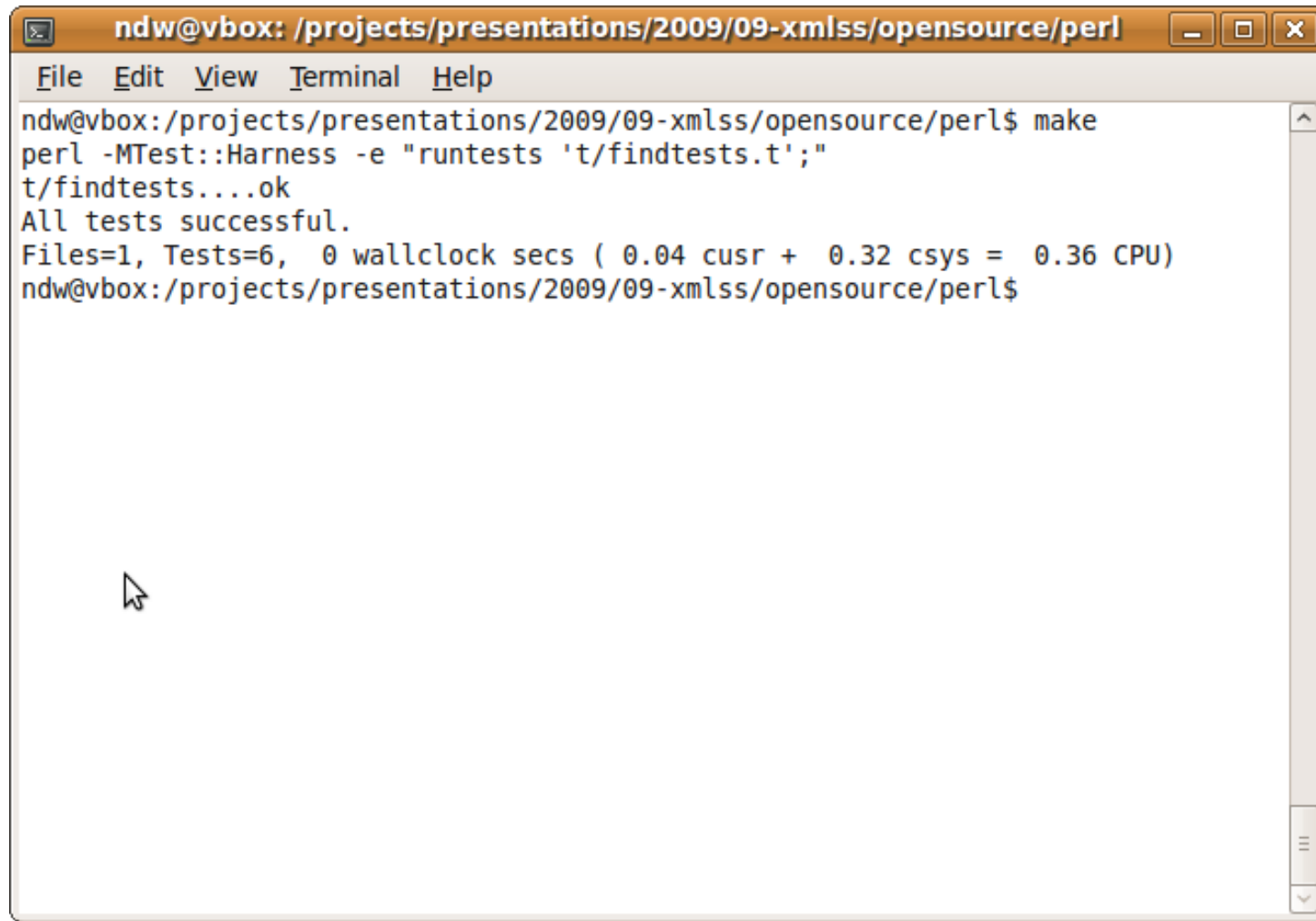
```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl$ svn status
M      lib/findpoints.pl
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl$ svn diff lib/findpoints.pl
Index: lib/findpoints.pl
=====
--- lib/findpoints.pl      (revision 2)
+++ lib/findpoints.pl      (working copy)
@@ -17,6 +17,11 @@

    my $min = findLatLE($minlat, @points);

+   # We want GE, not LE
+   if (defined($min) && $points[$min]->{'lat'} < $minlat) {
+       $min++;
+   }
+
    my $max = findLatLE($maxlat, @points);

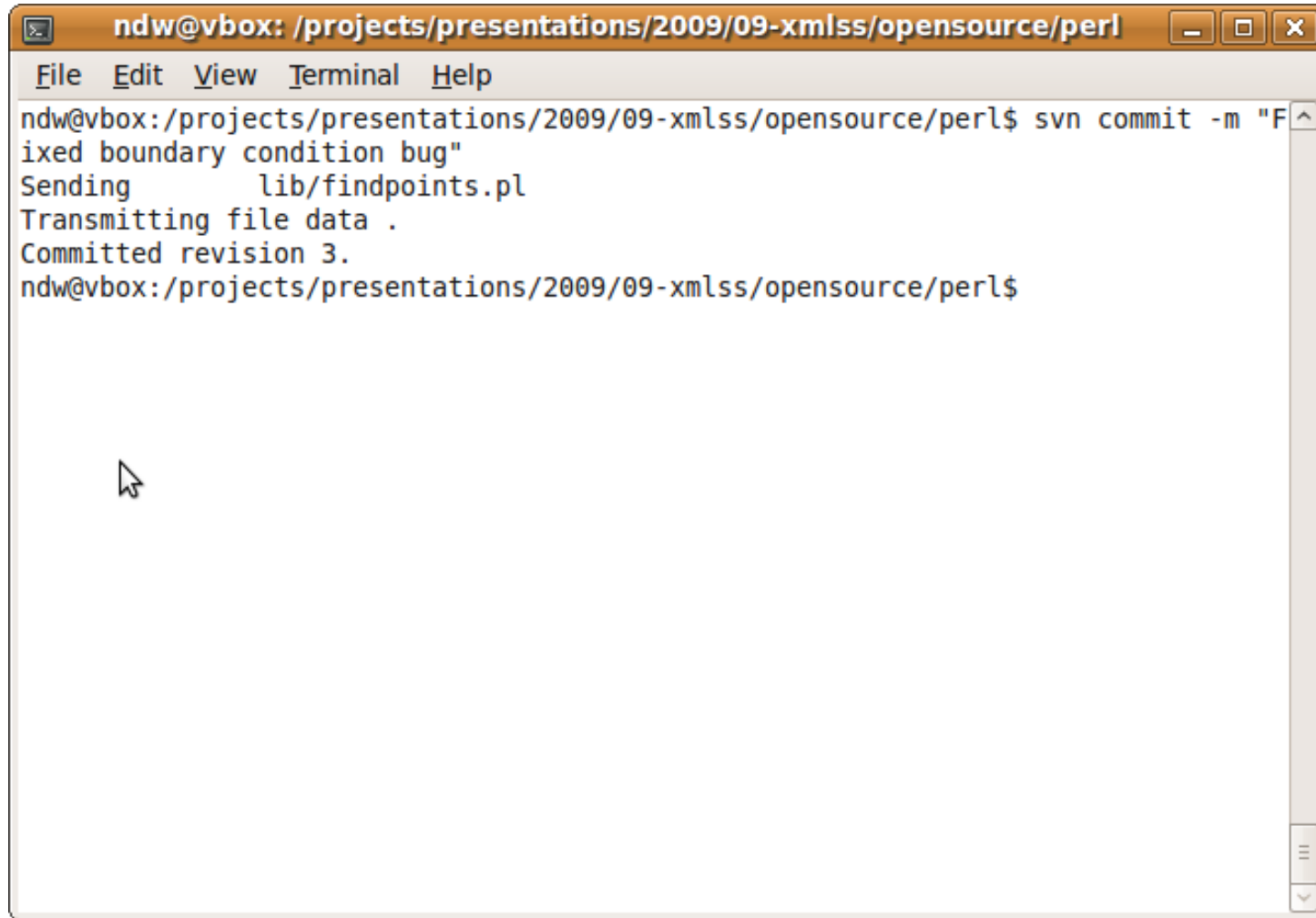
    if (!defined($max)) {
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl$
```

Perl test results (2)



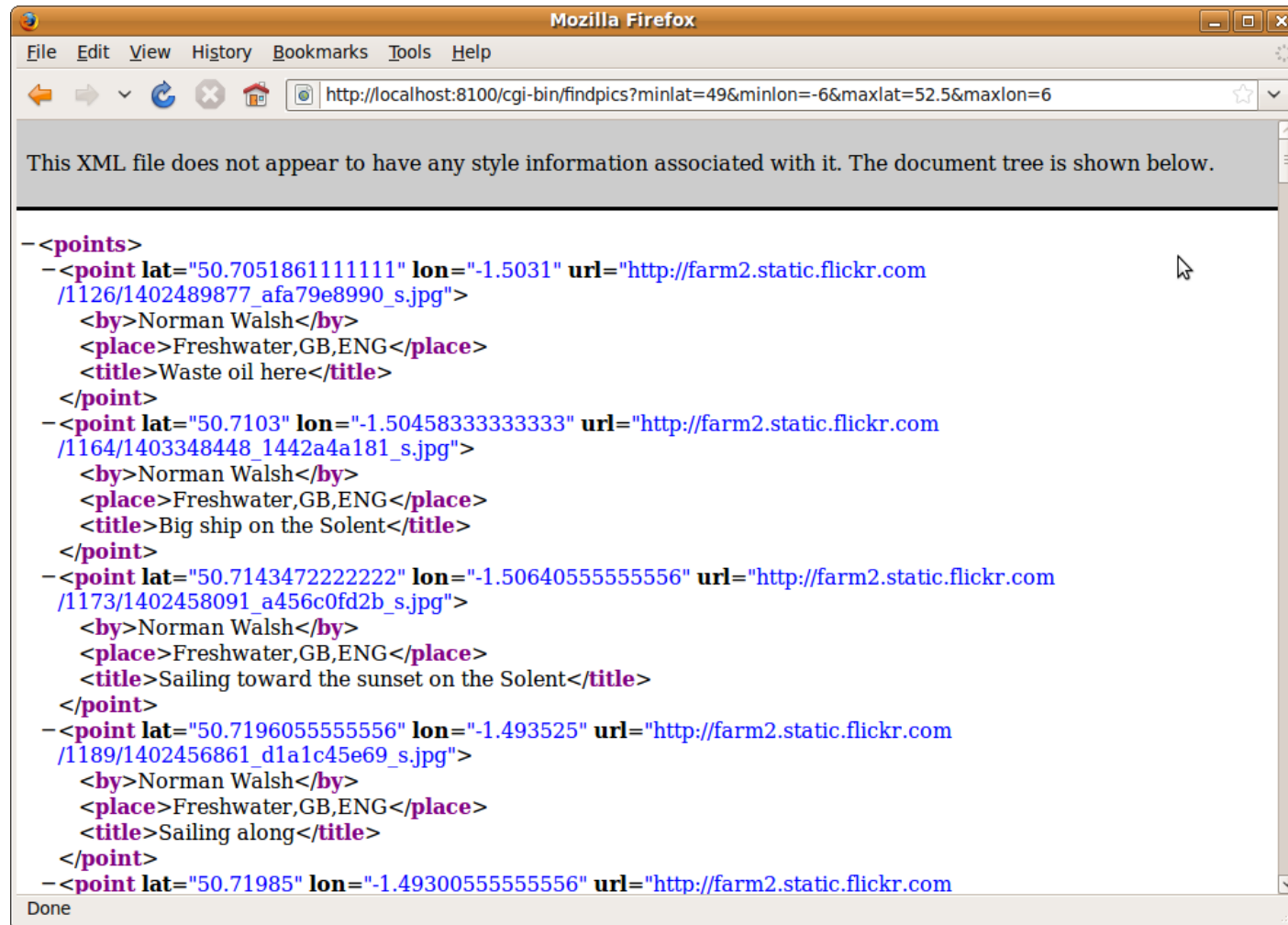
```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl$ make
perl -MTest::Harness -e "runtests 't/findtests.t';"
t/findtests....ok
All tests successful.
Files=1, Tests=6, 0 wallclock secs ( 0.04 cusr + 0.32 csys = 0.36 CPU)
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$
```

Subversion commit



```
ndw@vbox: /projects/presentations/2009/09-xmlss/opensource/perl
File Edit View Terminal Help
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$ svn commit -m "Fixed boundary condition bug"
Sending          lib/findpoints.pl
Transmitting file data .
Committed revision 3.
ndw@vbox:/projects/presentations/2009/09-xmlss/opensource/perl$
```

Perl results



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-<points>
-  <point lat="50.705186111111" lon="-1.5031" url="http://farm2.static.flickr.com/1126/1402489877_afa79e8990_s.jpg">
    <by>Norman Walsh</by>
    <place>Freshwater,GB,ENG</place>
    <title>Waste oil here</title>
  </point>
-  <point lat="50.7103" lon="-1.5045833333333" url="http://farm2.static.flickr.com/1164/1403348448_1442a4a181_s.jpg">
    <by>Norman Walsh</by>
    <place>Freshwater,GB,ENG</place>
    <title>Big ship on the Solent</title>
  </point>
-  <point lat="50.714347222222" lon="-1.5064055555556" url="http://farm2.static.flickr.com/1173/1402458091_a456c0fd2b_s.jpg">
    <by>Norman Walsh</by>
    <place>Freshwater,GB,ENG</place>
    <title>Sailing toward the sunset on the Solent</title>
  </point>
-  <point lat="50.719605555556" lon="-1.493525" url="http://farm2.static.flickr.com/1189/1402456861_d1a1c45e69_s.jpg">
    <by>Norman Walsh</by>
    <place>Freshwater,GB,ENG</place>
    <title>Sailing along</title>
  </point>
-  <point lat="50.71985" lon="-1.4930055555556" url="http://farm2.static.flickr.com
```

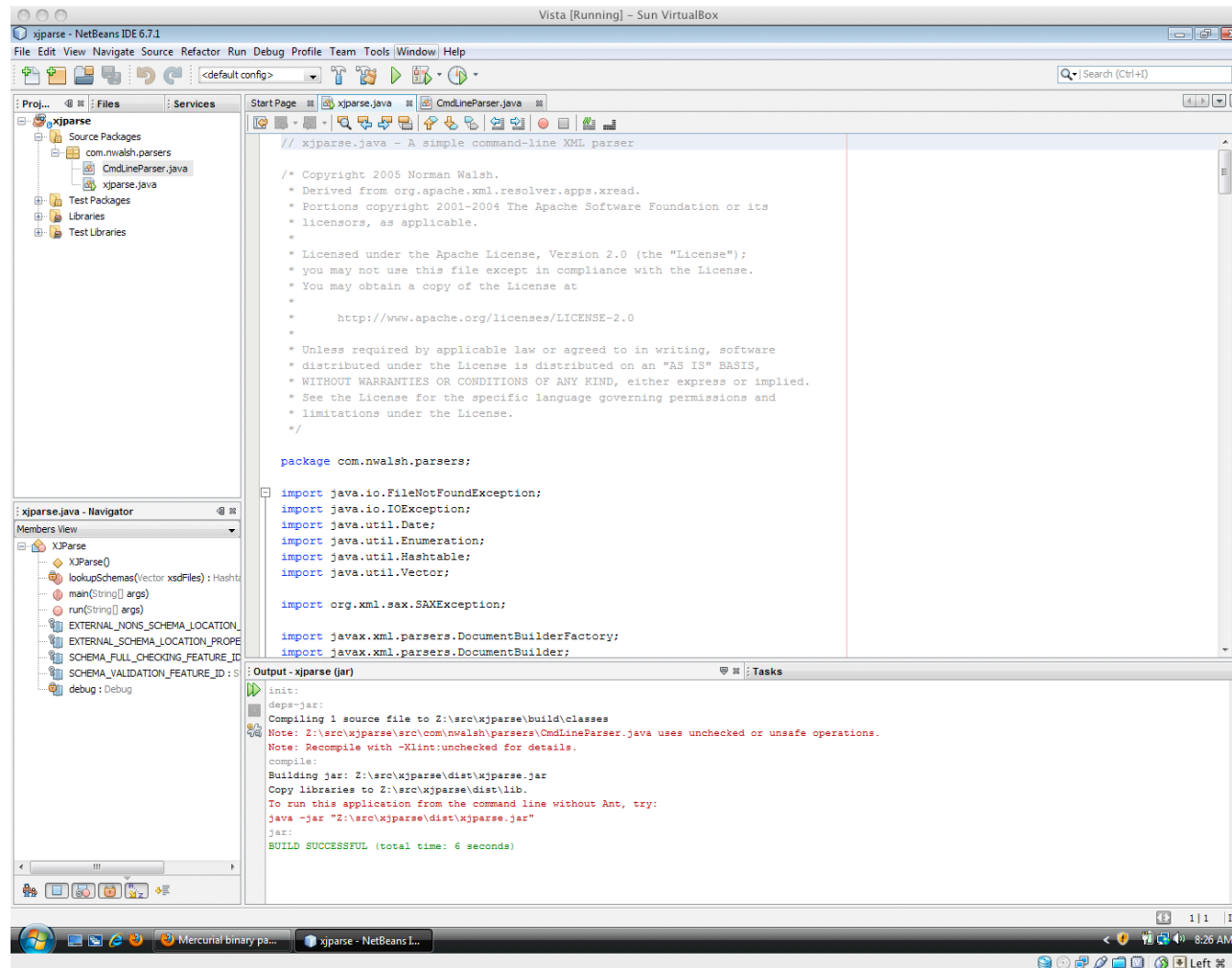
Done

Command-line Java

- Windows
- Java
- NetBeans
- Mercurial
- JUnit



NetBeans

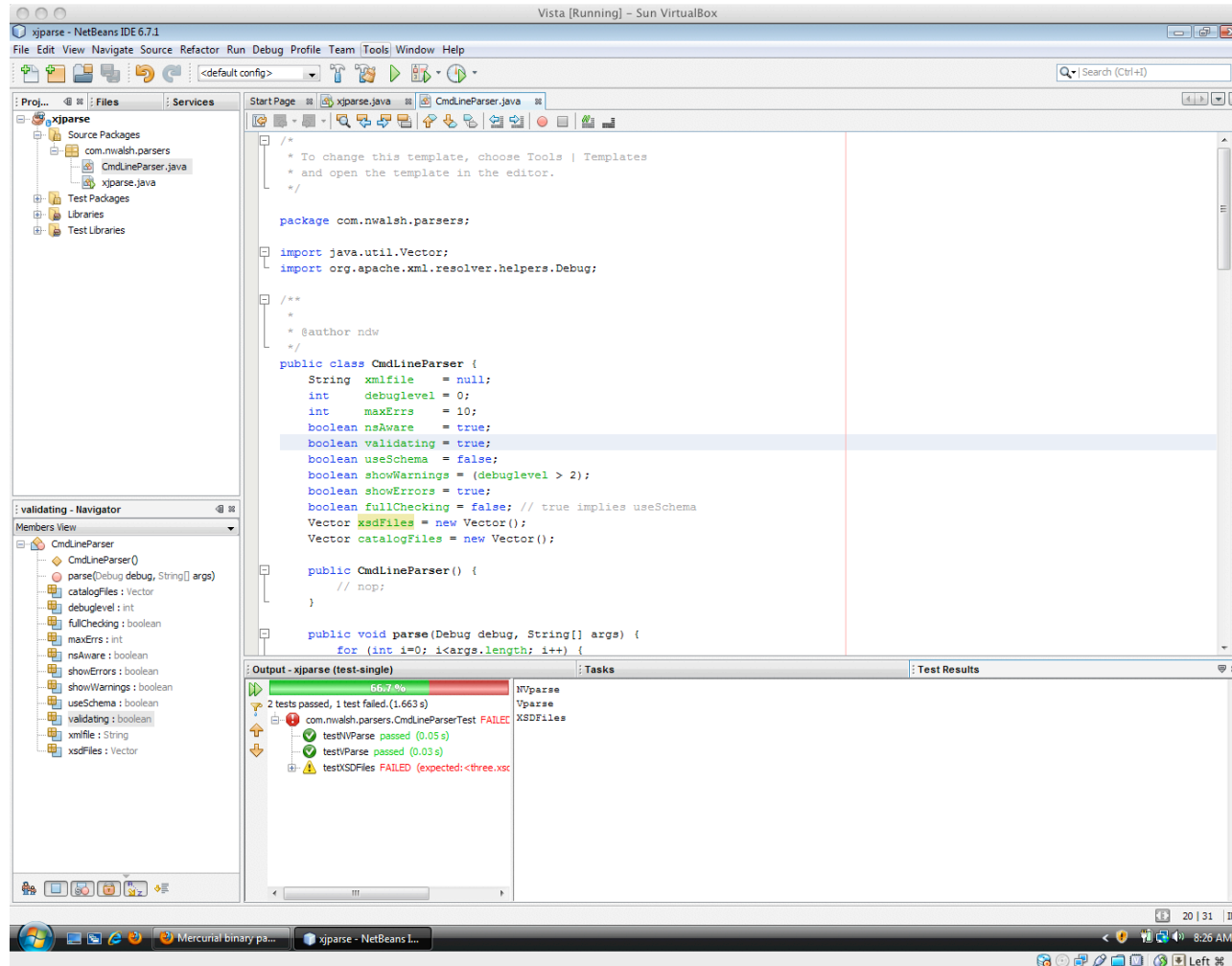


Licensed under a Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported License

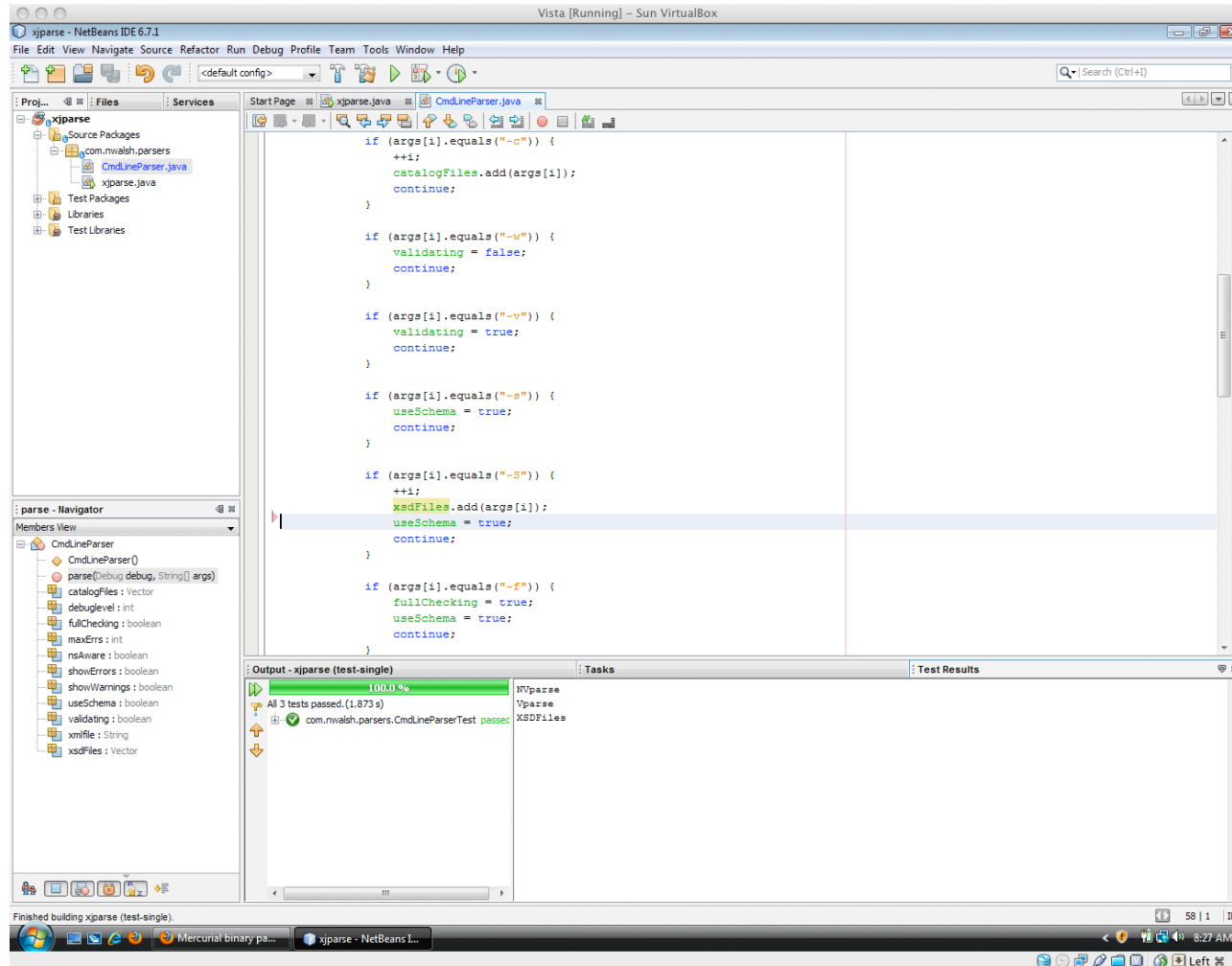
www.xmlsummerschool.com

Slide 46

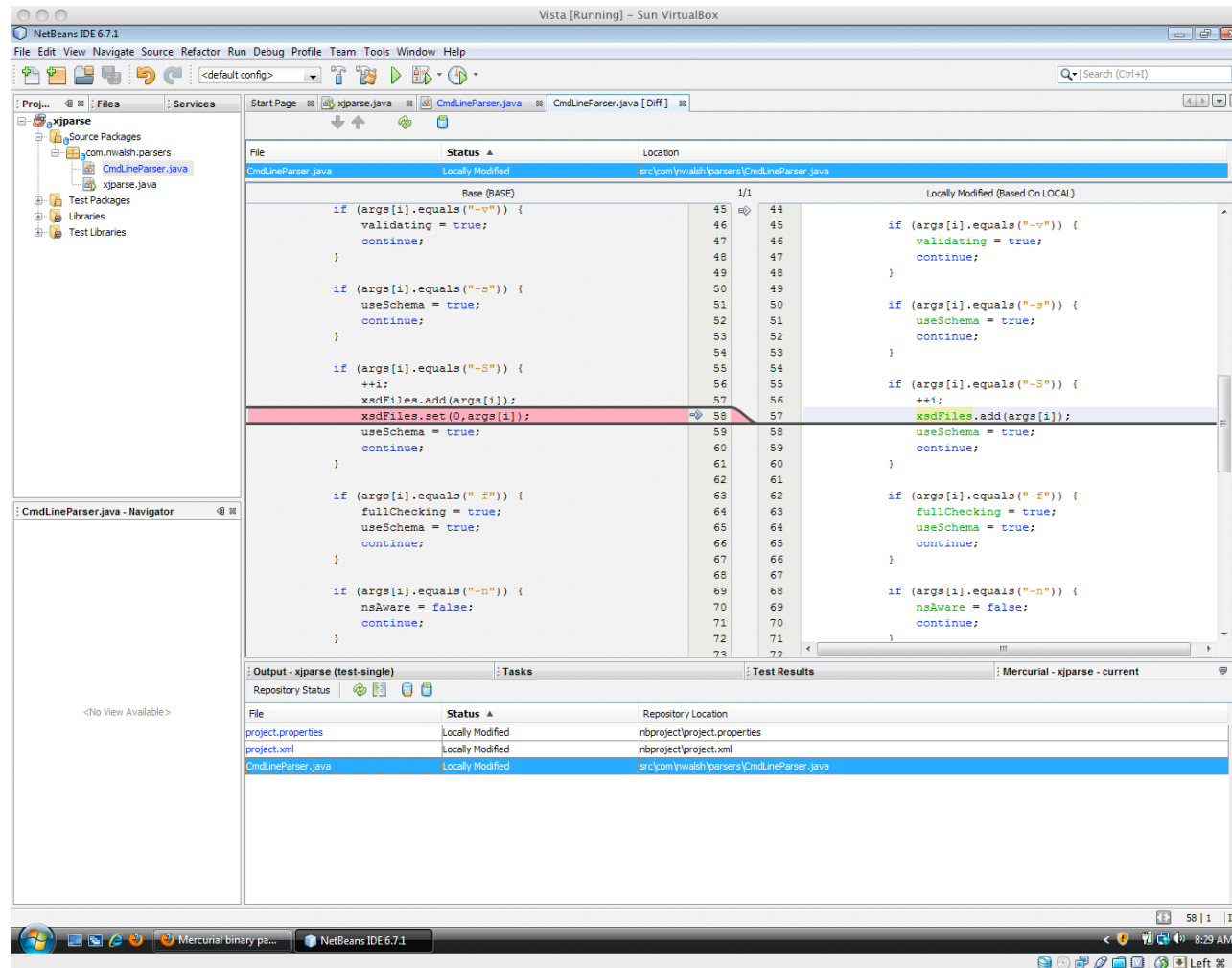
NetBeans Unite Tests Fail



NetBeans Unit Tests Pass

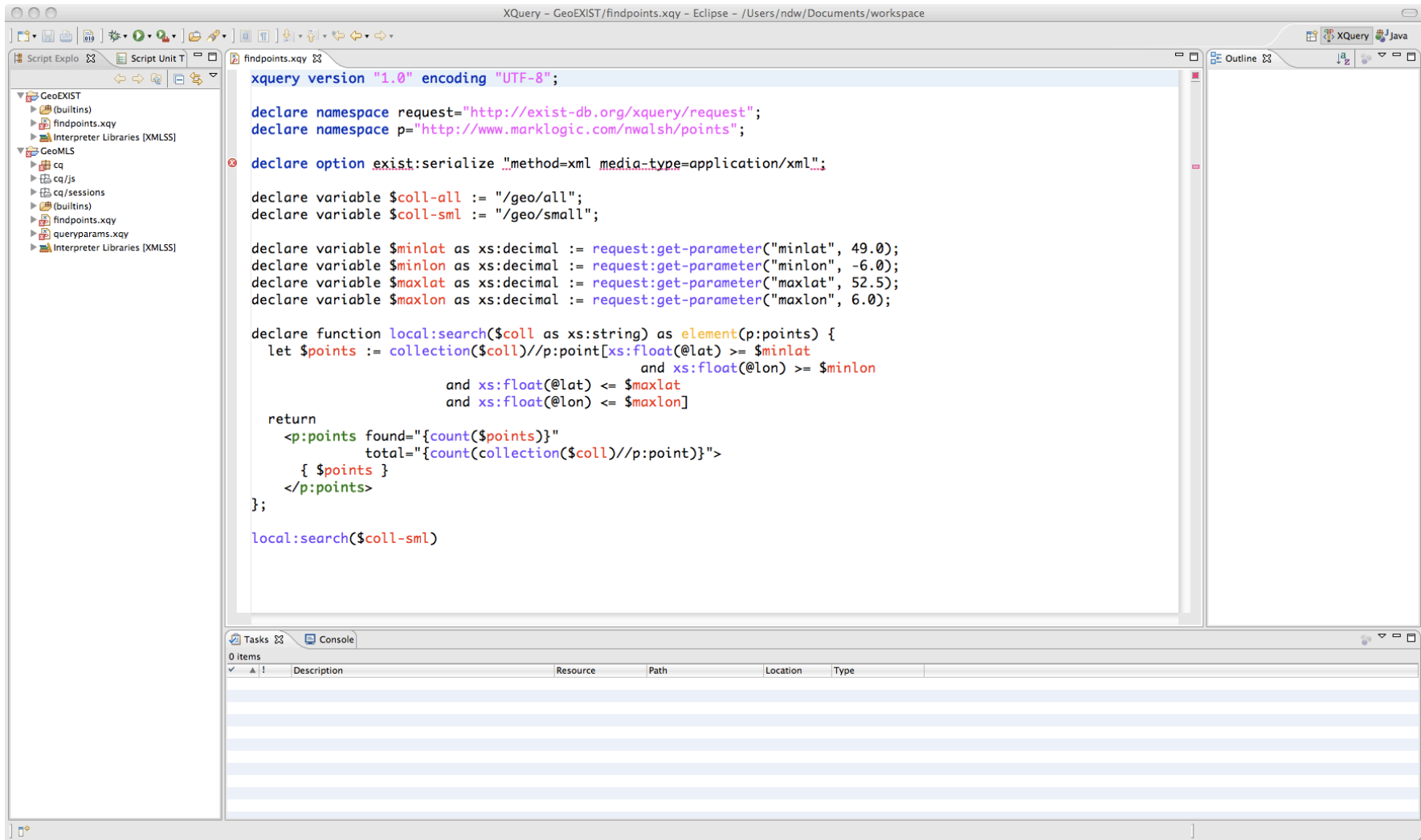


NetBeans Mercurial Diff



XQuery

- OS X
- XQuery
- Eclipse
- Subversion
- eXist & MarkLogic Server



```
xquery version "1.0" encoding "UTF-8";

declare namespace request="http://exist-db.org/xquery/request";
declare namespace p="http://www.marklogic.com/nwalsh/points";

declare option exist:serialize "method=xml media-type=application/xml";

declare variable $coll-all := "/geo/all";
declare variable $coll-sml := "/geo/small";

declare variable $minlat as xs:decimal := request:get-parameter("minlat", 49.0);
declare variable $minlon as xs:decimal := request:get-parameter("minlon", -6.0);
declare variable $maxlat as xs:decimal := request:get-parameter("maxlat", 52.5);
declare variable $maxlon as xs:decimal := request:get-parameter("maxlon", 6.0);

declare function local:search($coll as xs:string) as element(p:points) {
  let $points := collection($coll)//p:point[xs:float(@lat) >= $minlat
                                and xs:float(@lon) >= $minlon
                                and xs:float(@lat) <= $maxlat
                                and xs:float(@lon) <= $maxlon]

  return
    <p:points found="{count($points)}"
      total="{count(collection($coll)//p:point)}">
      { $points }
    </p:points>
};

local:search($coll-sml)
```



Tutorial Session



Any questions?

Thank you for your attention