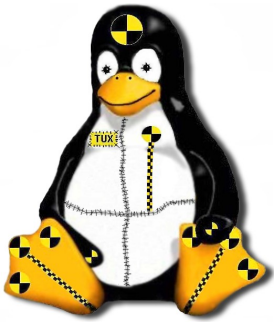


Linux Ecosystem around Test

Masatake YAMATO
Linux Test Project

About myself

- Red Hat employee(yamato@redhat.com),
- working as a consultant ,
- having worked on Crackerjack for two years,
- porting some test cases from crackerjack to *LTP*,
- having interests about FOSS project management
  - how to involve more people to a project,
  - how to push my patch to a project, etc.

Crackerjack and LTP

Linux Test Project(LTP)

- a test project testing linux kernel,
- covers not only system calls but other aspects of kernel such as kdump, scheduler, numa, etc.
- started by SGI in 2000(?).
- led by Subrata Modak.
  - very aggressive maintainer
  - an employee of IBM.
- used widely by kernel developers.
  - Some developers have submitted test cases to LTP.
- used and developed by QA people of each GNU/Linux distributions.

Testing environments and cycle

- The maintainer runs the tests
  - on many different architectures (i386, x86_64, ppc, ia64 and s390x),
  - on many versions between lower(> 2.6.5) and the latest,
  - on many different distributions including (RH, SuSE, and Debian),
  - frequently(may be daily).
  - for long time (> 24h) for stress tests.
- Community people run tests their own environments.
  - architectures for embedded system like arm.

Relationship between Crackerjack and LTP The first contact

LTP maintainer

On Tue, 2008-05-27 at 20:46 +0530, Subrata Modak wrote:
> Hi Crackerjack Users/Developers,
>
> I was happening to browse through your home page:
>
> https://sourceforge.net/projects/crackerjack,
>
> and also went through your OLS 2007 paper on
>
> Regression Test Framework and Kernel Execution Coverage.
> (http://ols.108.redhat.com/2007/Reprints/yoshioka-Reprint.pdf)
>
> Your paper mentions that you can work with LTP to bring Regression
> Testing to more greater heights. I would also be interested for it. Let
> me know how we can start and move forward on this. I would also be
> interested to see if we can leverage your tests for LTP.
>
> Regards--
> Subrata

Paper about Crackerjack

Relationship between Crackerjack and LTP
The official response from Crackerjack

- Nothing till 2008-07-11
  - 1 month is enough to make people believe the project is dead.
- What can I do?

Relationship between Crackerjack and LTP
The official response from Crackerjack

- Nothing till 2008-07-11
  - 1 month is enough to make people believe the project is dead.
- What can I do?
  - asking to put a hyperlink to crackerjack web page at LTP project web page,
  - asking to have a face to face meeting,
  - porting test cases of Crackerjack to LTP.

My choice

Relationship between Crackerjack and LTP
My response

```
On Fri, 2008-06-13 at 18:35 +0900, Masatake YAMATO wrote:
> Subrata, I'll take my spare time.
> (But please don't expect too much, I got a baby:-)
```

What I have done

- focusing only on porting the concrete test cases,
- temporary ignoring “regression test” concept of crackerjack,
- writing original 2 test cases for LTP,
- porting 13 test cases from Crackerjack to LTP,
- fixing 1 kernel bug through writing a test,
- fixing 1 numactl library bug through reading a test case, and
- Fixing 1 man page bug through porting a test.
- 1 new test case and 1 ported test case are in review stage now.

Issues faced in porting

- native language used in comments
- no copyright notice

Issues met with in porting
native language used in comments

I want to port a test case with its flavor and intention of the original author to LTP. However, I found following code in a crackerjack test case.

```
   switch (iminor(inode)) {
      case 1:
         filp->f_op = &mem_fops;
         break;
      case 3:
        /*  你好 ...  */
         filp->f_op = &null_fops;
```

If I don't understand the comment, should I delete the comment in ported code? I don't want to do so.

Issues met with in porting
native language used in comments

I want to port a test case with its flavor and intention of the original author to LTP. However, I found following code in a crackerjack test case.

```
   switch (iminor(inode)) {
      case 1:
         filp->f_op = &mem_fops;
         break;
      case 3:
        /* inode のマイナー番号を見て、
           /dev/null 専用の関数テーブル
          に置き換えている。 */
         filp->f_op = &null_fops;
```

If I don't understand the comment, should I delete the comment in ported code? I don't want to do so.

Issues met with in porting
native language used in comments

I want to port a test case with its flavor and intention of the original author to LTP. However, I found following code in a crackerjack test case.

```
   switch (iminor(inode)) {
      case 1:
         filp->f_op = &mem_fops;
         break;
      case 3:
        /* 안녕하십니까 ... */
         filp->f_op = &null_fops;
```

If I don't understand the comment, should I delete the comment in ported code? I don't want to do so.

Issues met with in porting
no copyright notice

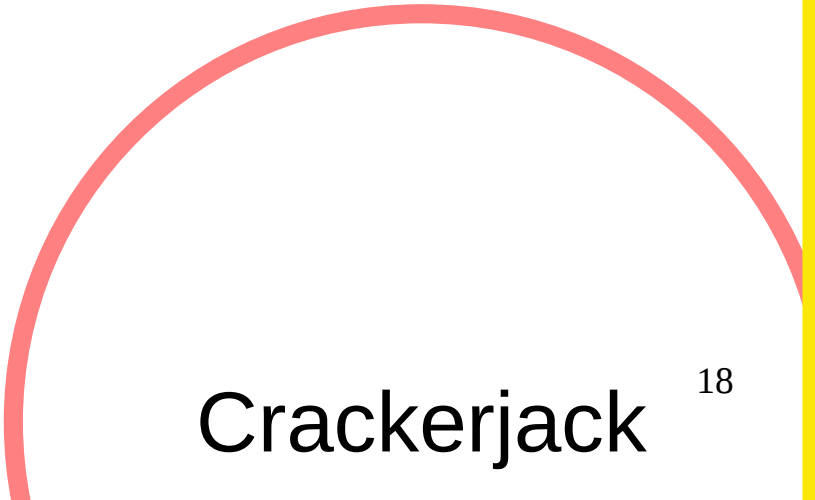
Some test cases I wanted to port to LTP have no copyright notice in top of the source code files. LTP expects GPL2 for imported code. So I aborted to port these test cases.

My current target in LTP

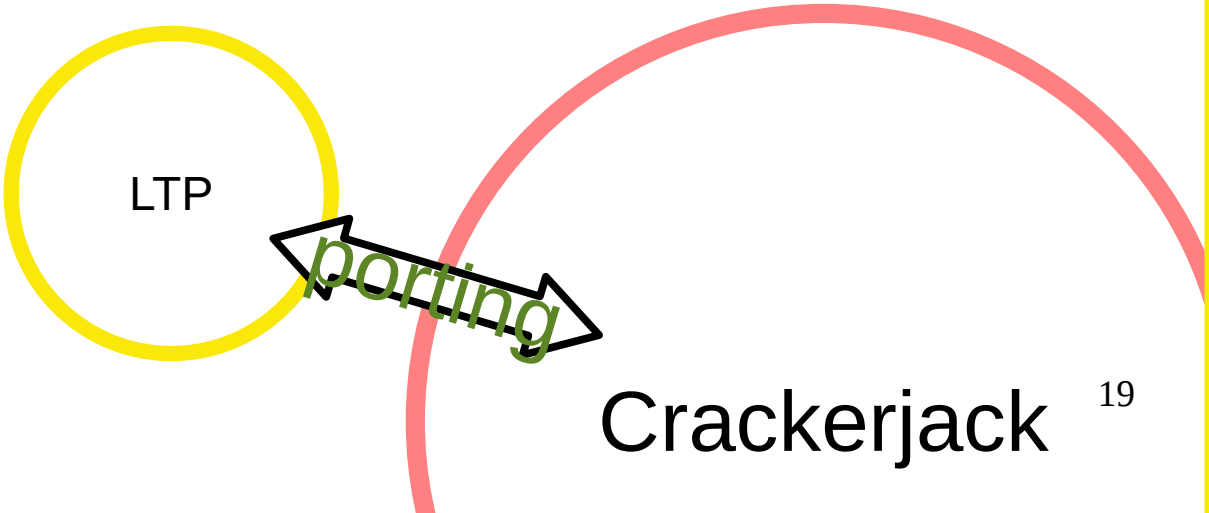
- porting three system calls written by Taruishi-san because his code uses experimental LTP friendly coding style,
- porting nine 64bit newer system calls like stat64, and
- porting fifteen 16bit system calls for keeping compatibility like getuid16.

Ecosystem: *Philosophy*

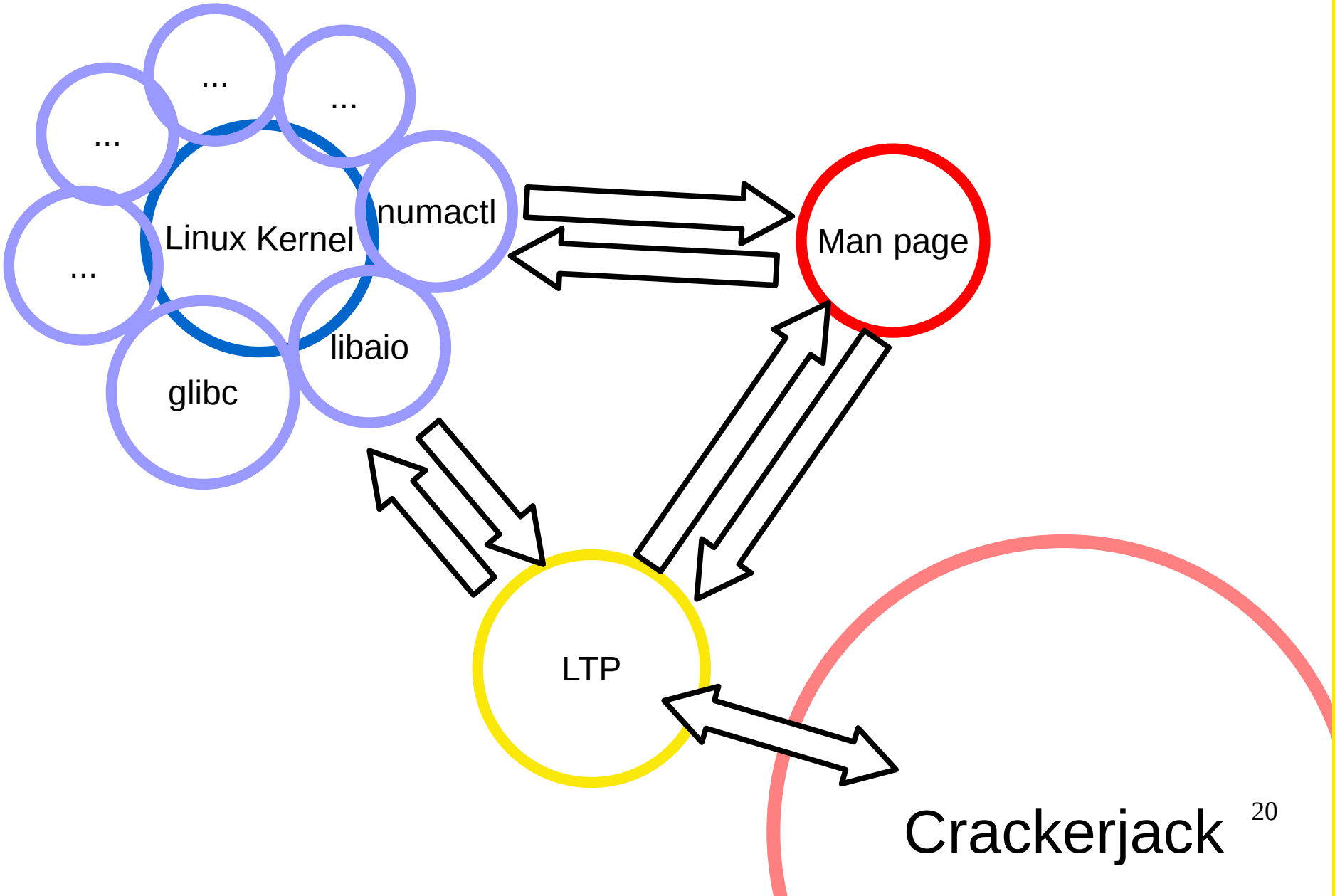
Position of Crackerjack



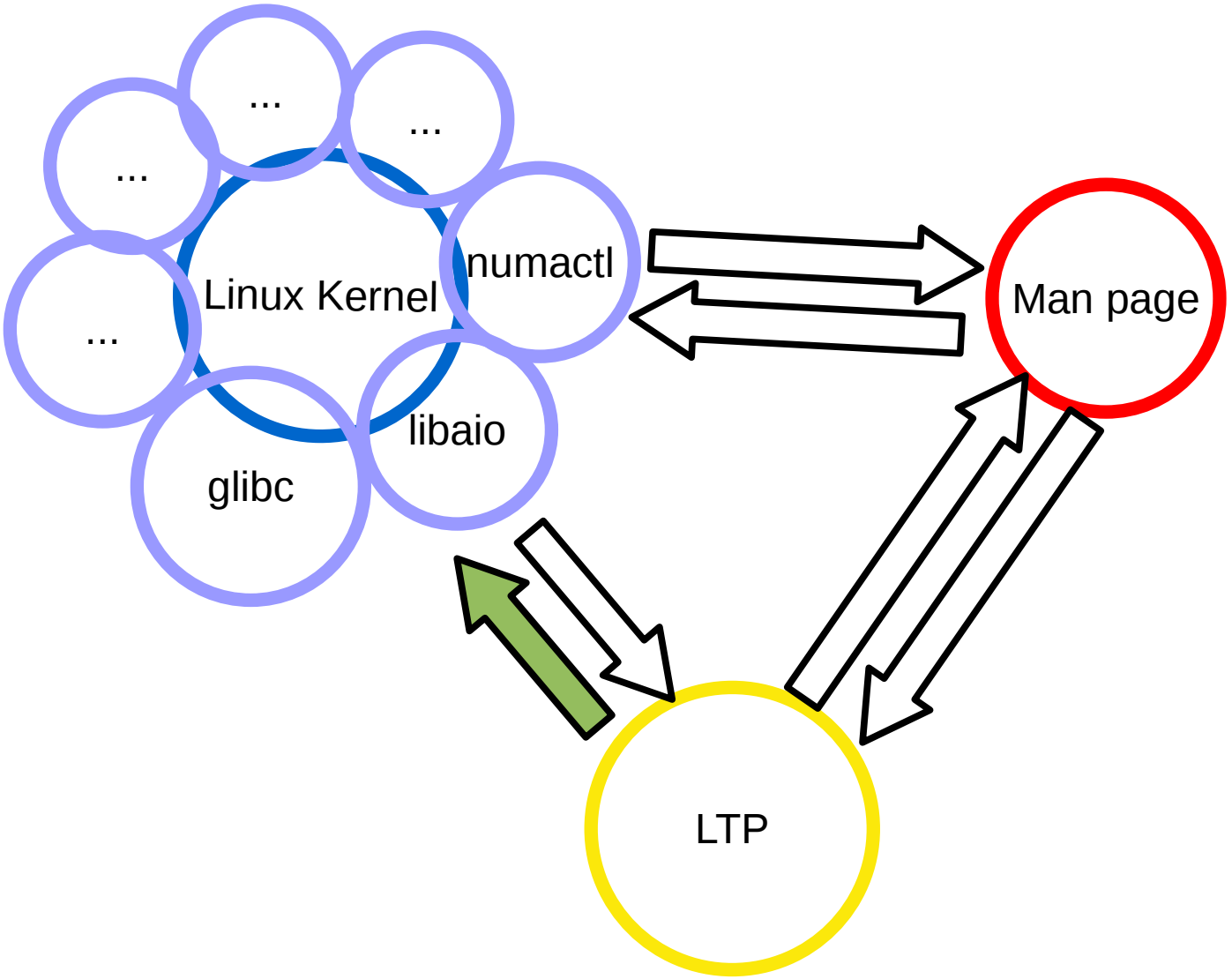
Position of Crackerjack



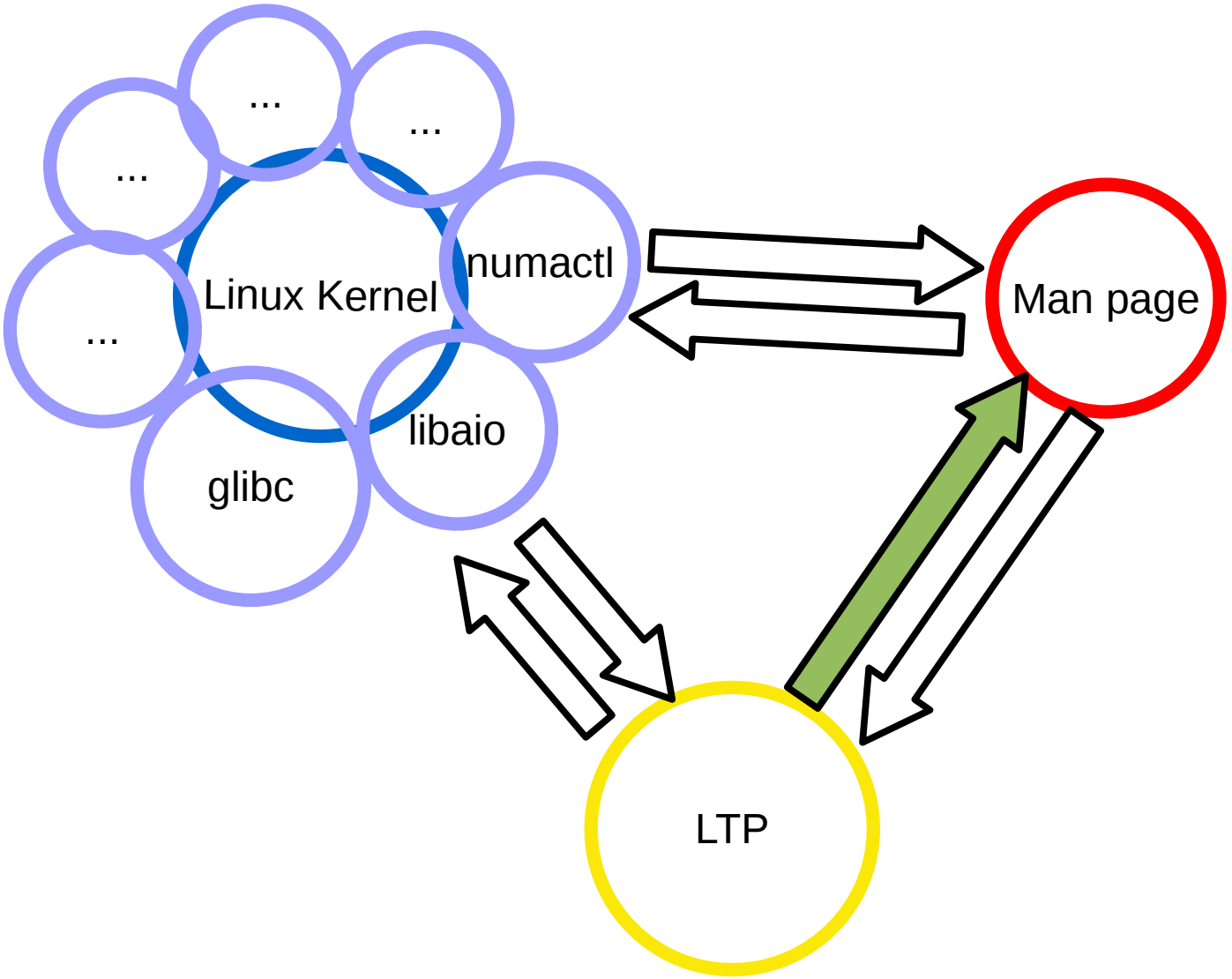
Projects around LTP



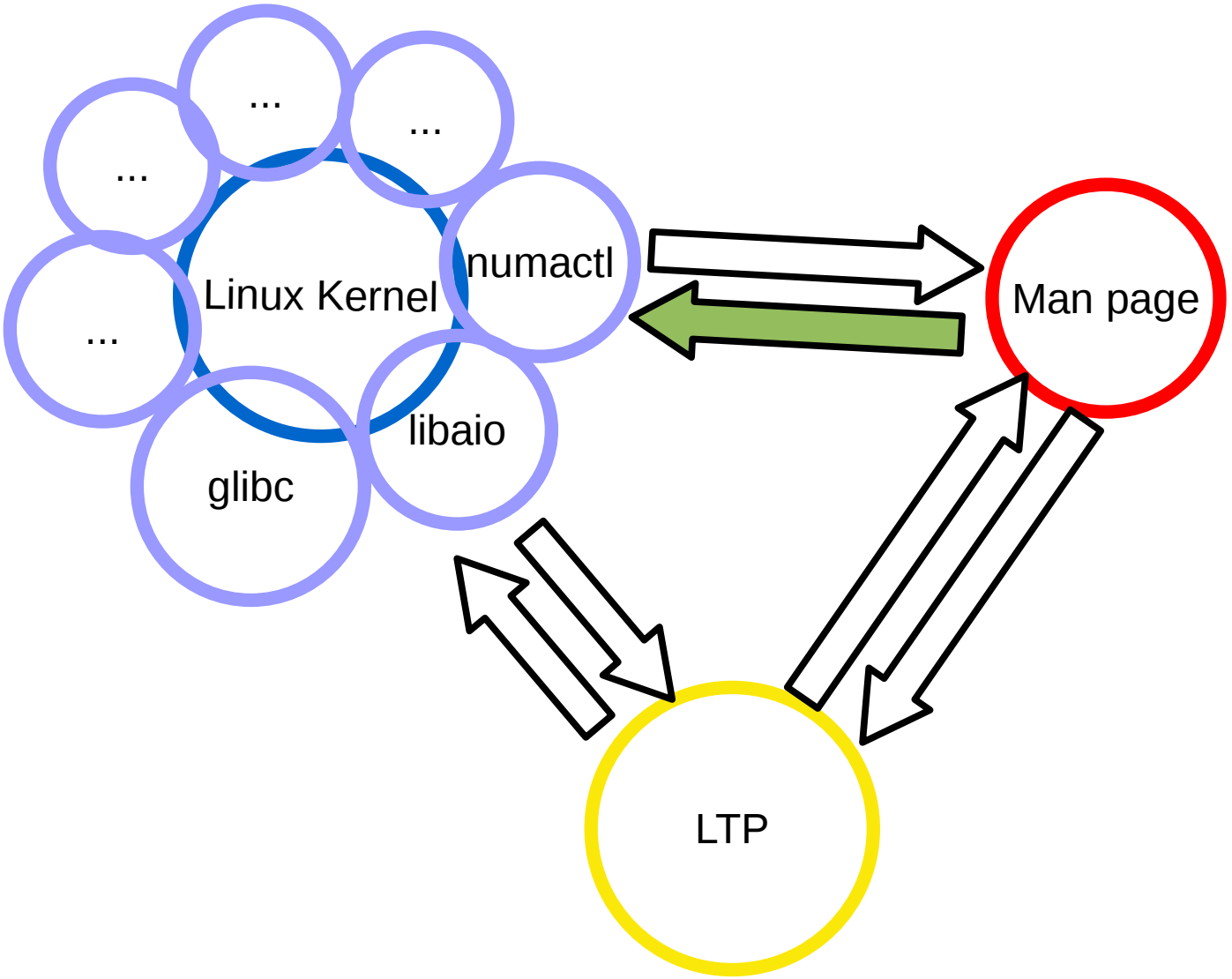
A. LTP tests the kernel.



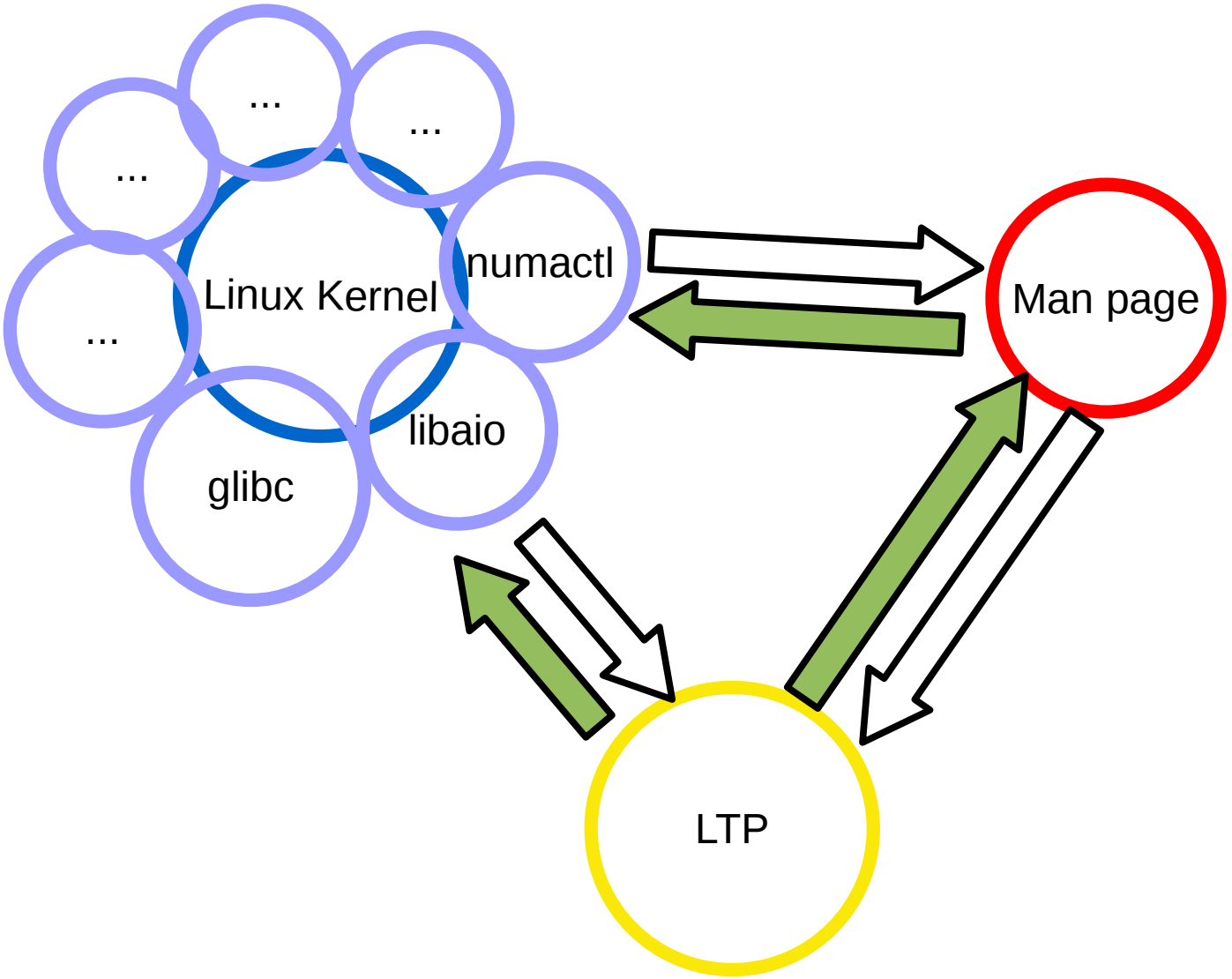
B. LTP refers man pages to write test cases.



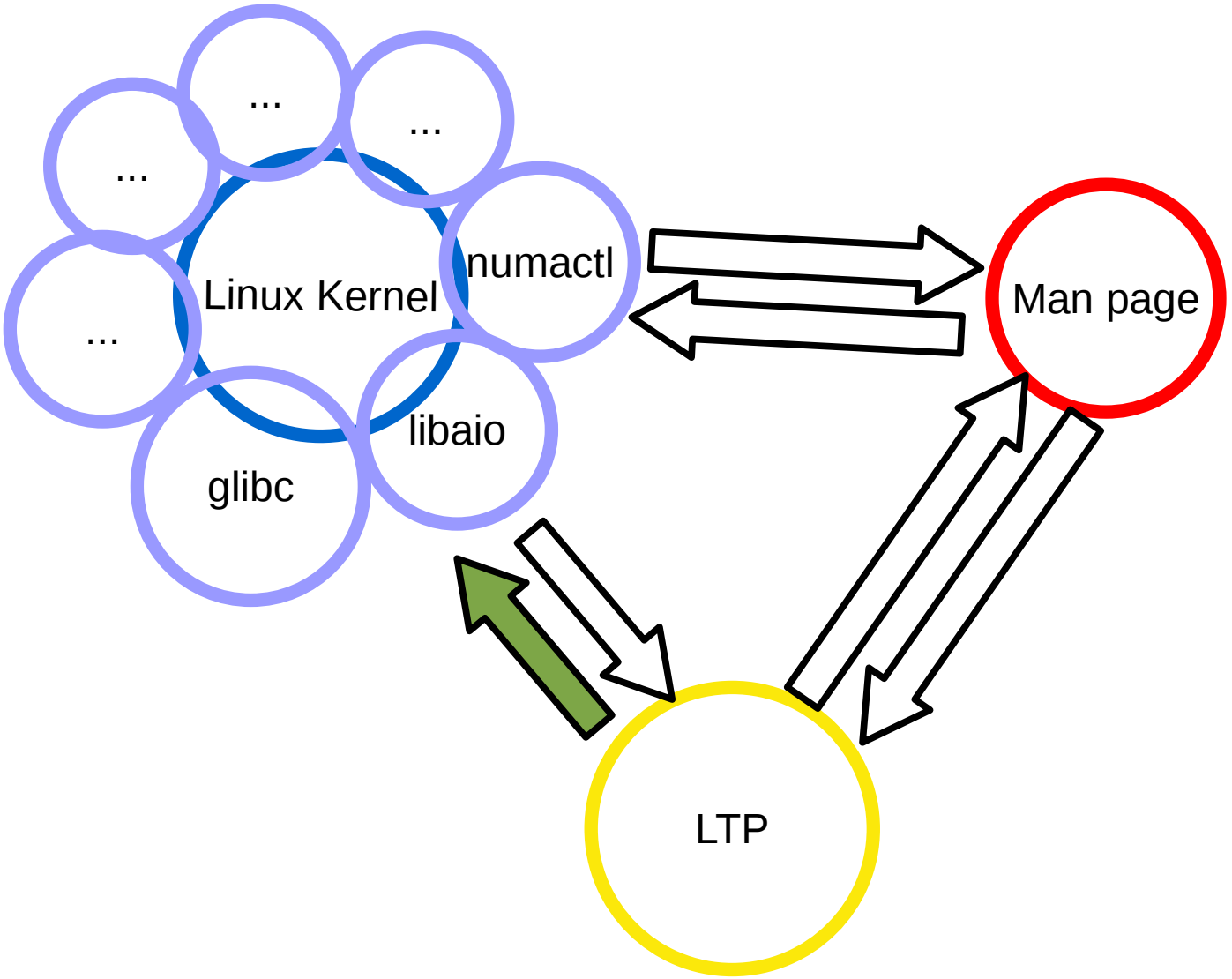
C. Man pages explain the system calls.



Enough? - NO. More to FOSS than this.

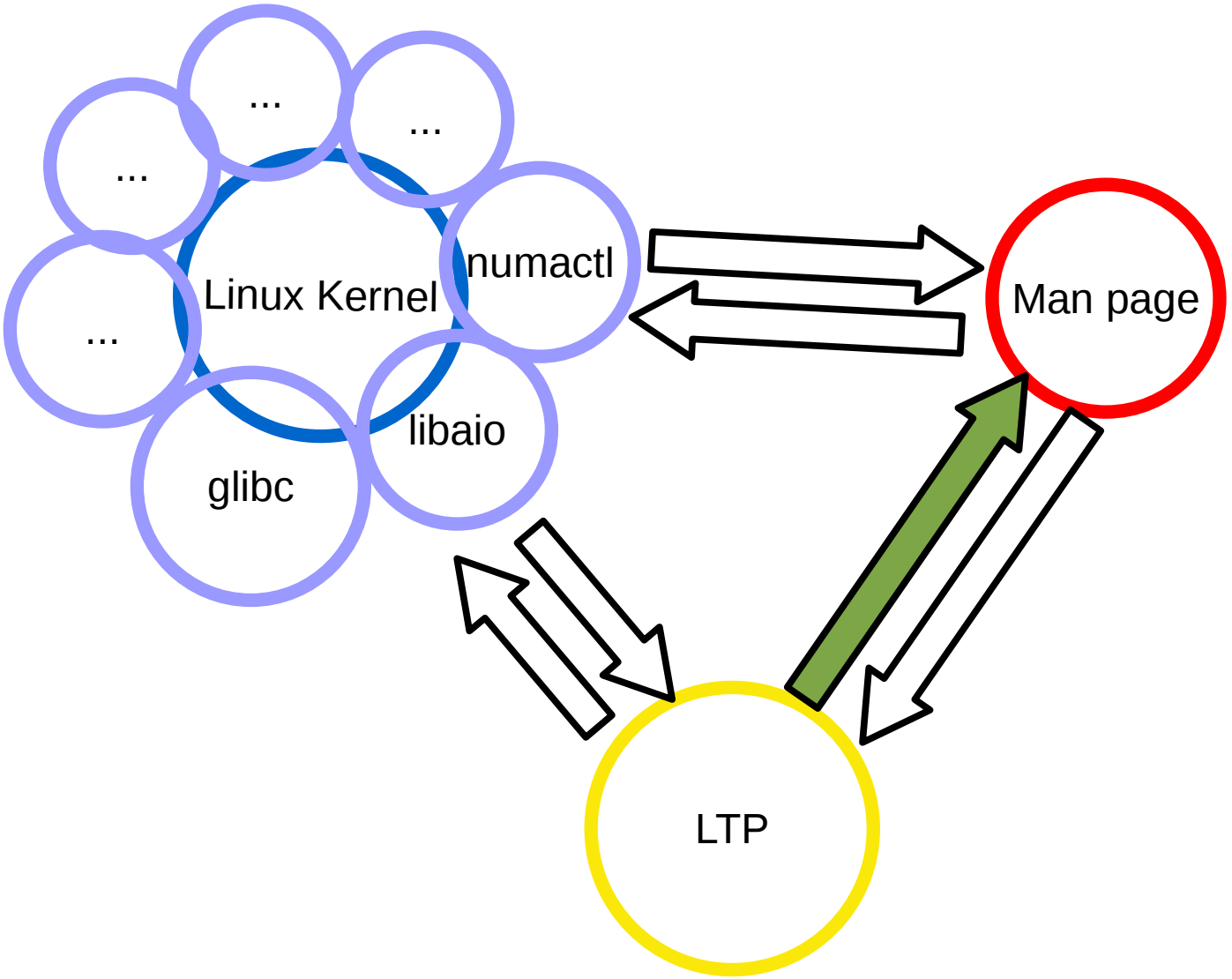


A'. Test authors can hunt/report bugs of the kernel and libraries around it.



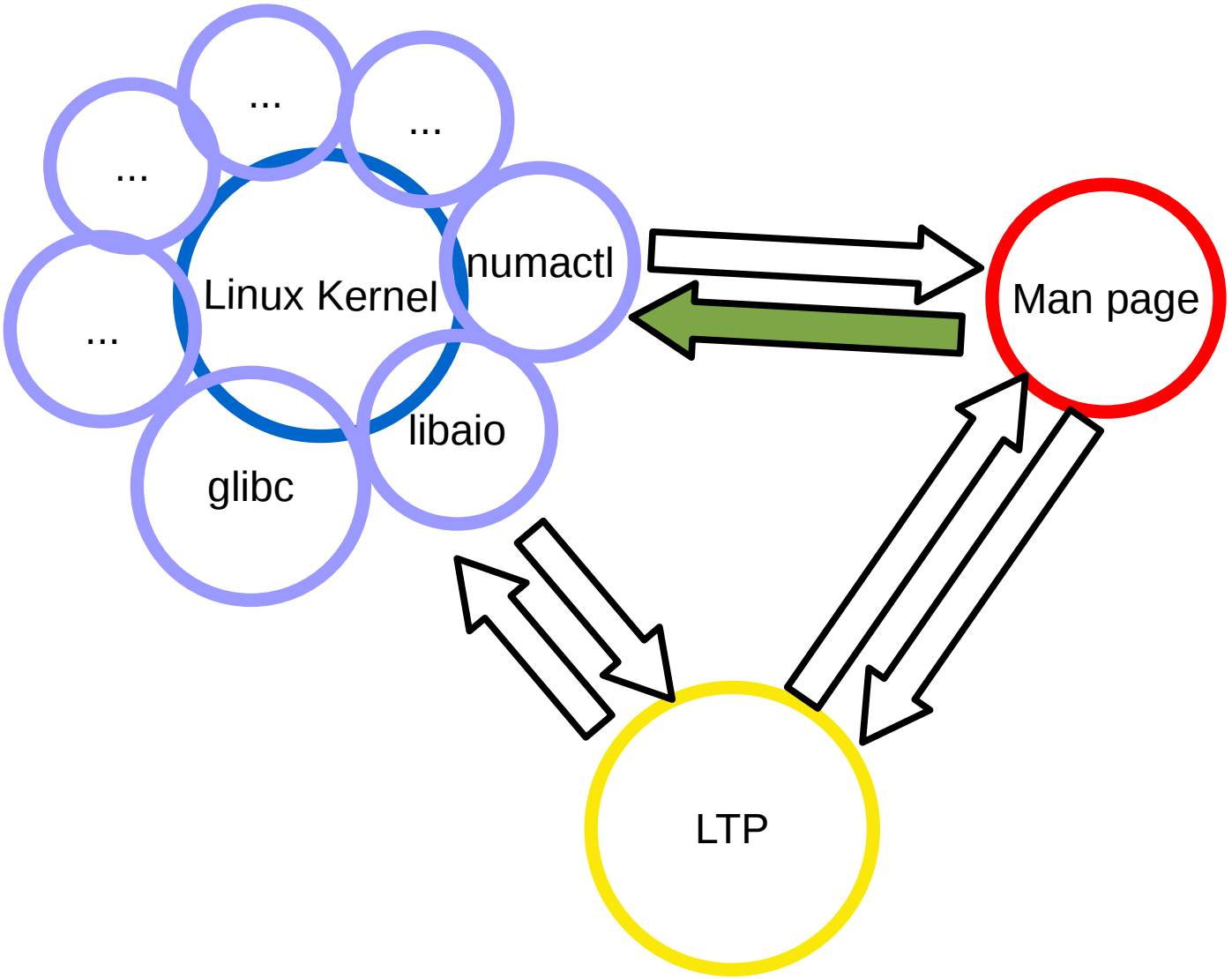
Tests put pressure on kernel code. As the result kernel discloses bugs.

B'. Test authors can review man pages and fix typos in them.



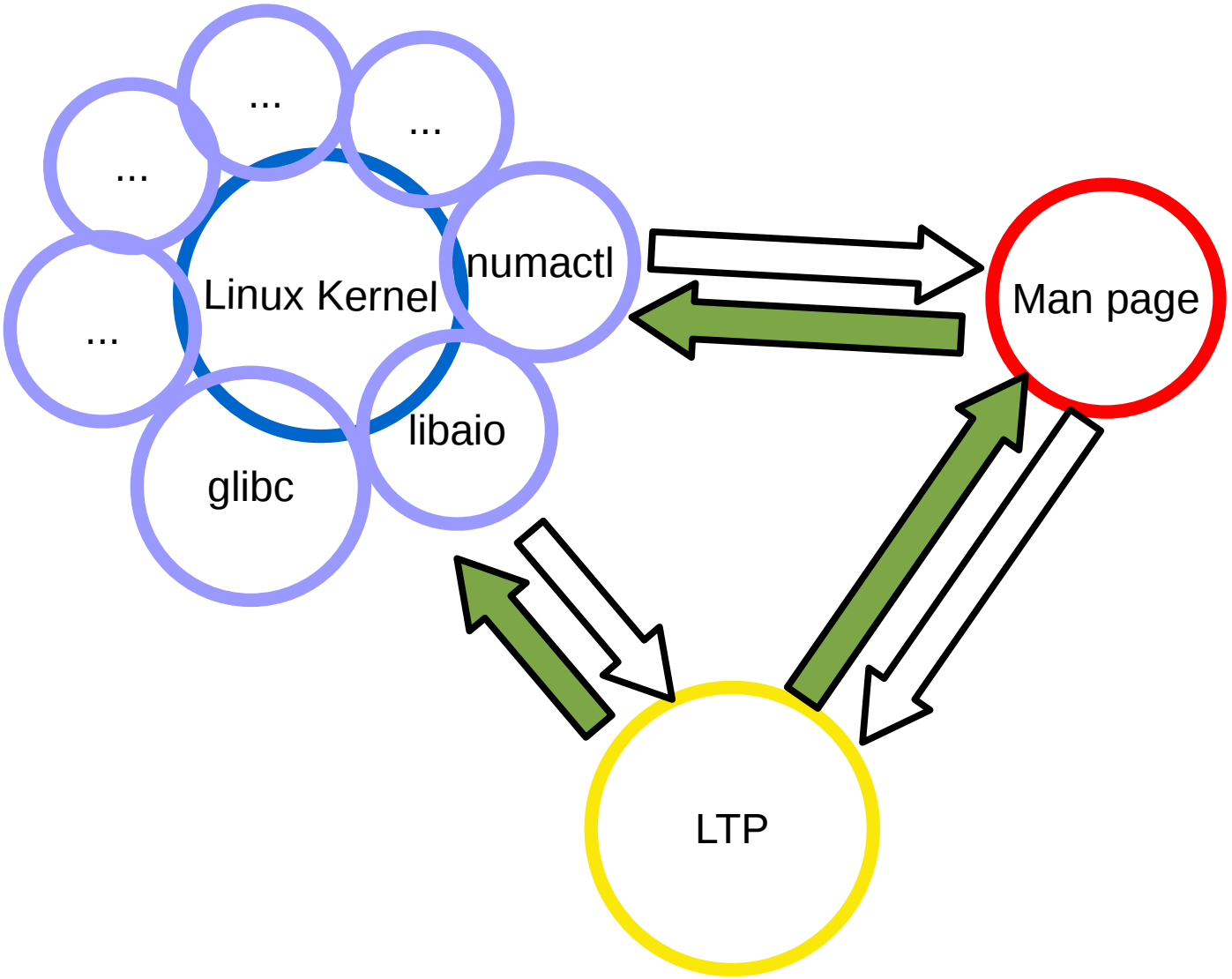
To write test case, the authors have to read man pages very carefully.

C'. Man page authors can hunt/report bugs of the kernel and libraries around it.

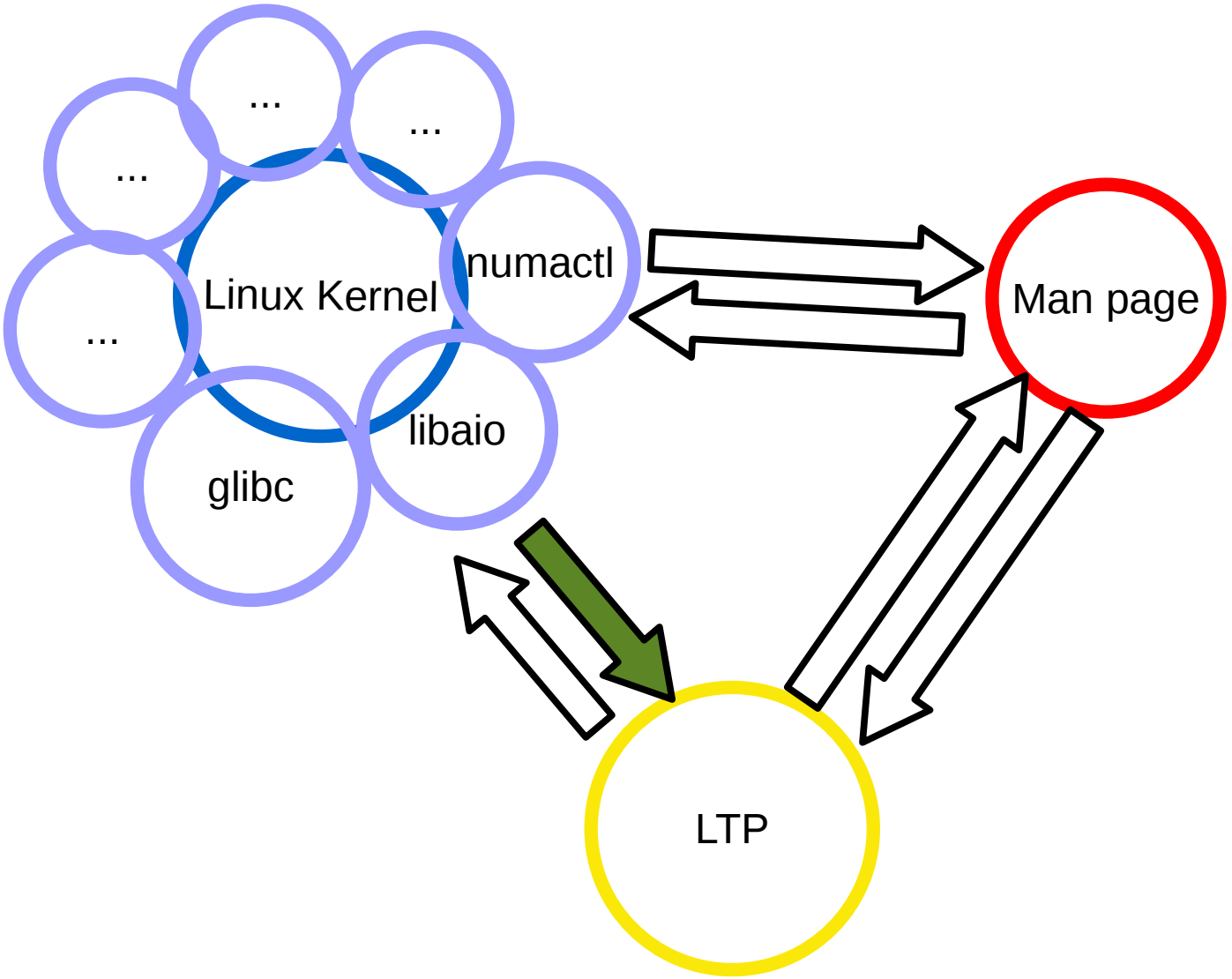


To write man page, the authors have to understand the functions deeply.

Enough? - NO. Much More to FOSS than this.

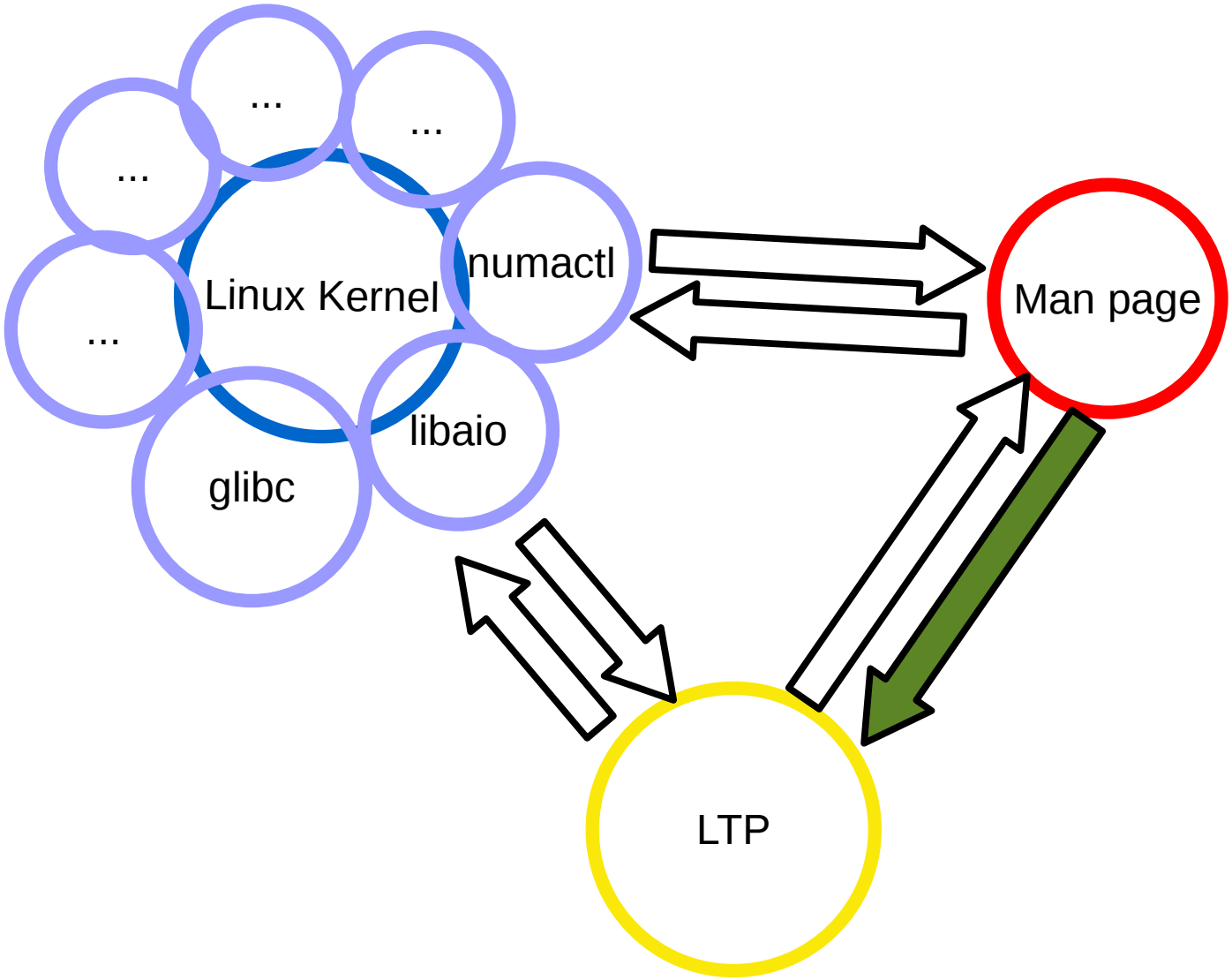


A”. Kernel developer can submit test cases to LTP



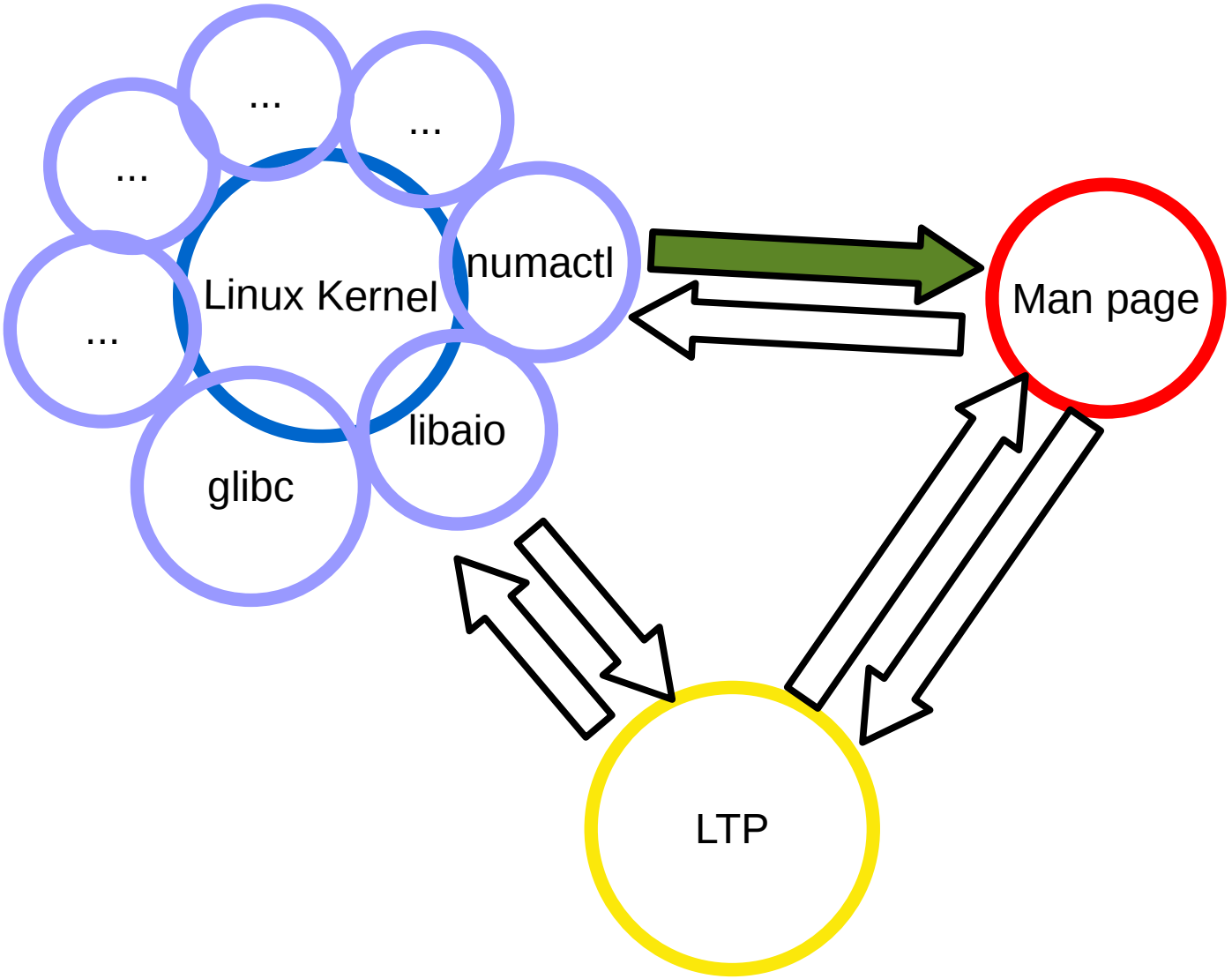
If one adds new function to kernel, s/he must have test cases for it.

B”. Man pages authors can submit test cases to LTP.



Remember C': Man page authors may run system calls to understand them. The code to run them can be seeds of test cases.

C”. Kernel developers can write/update man pages.



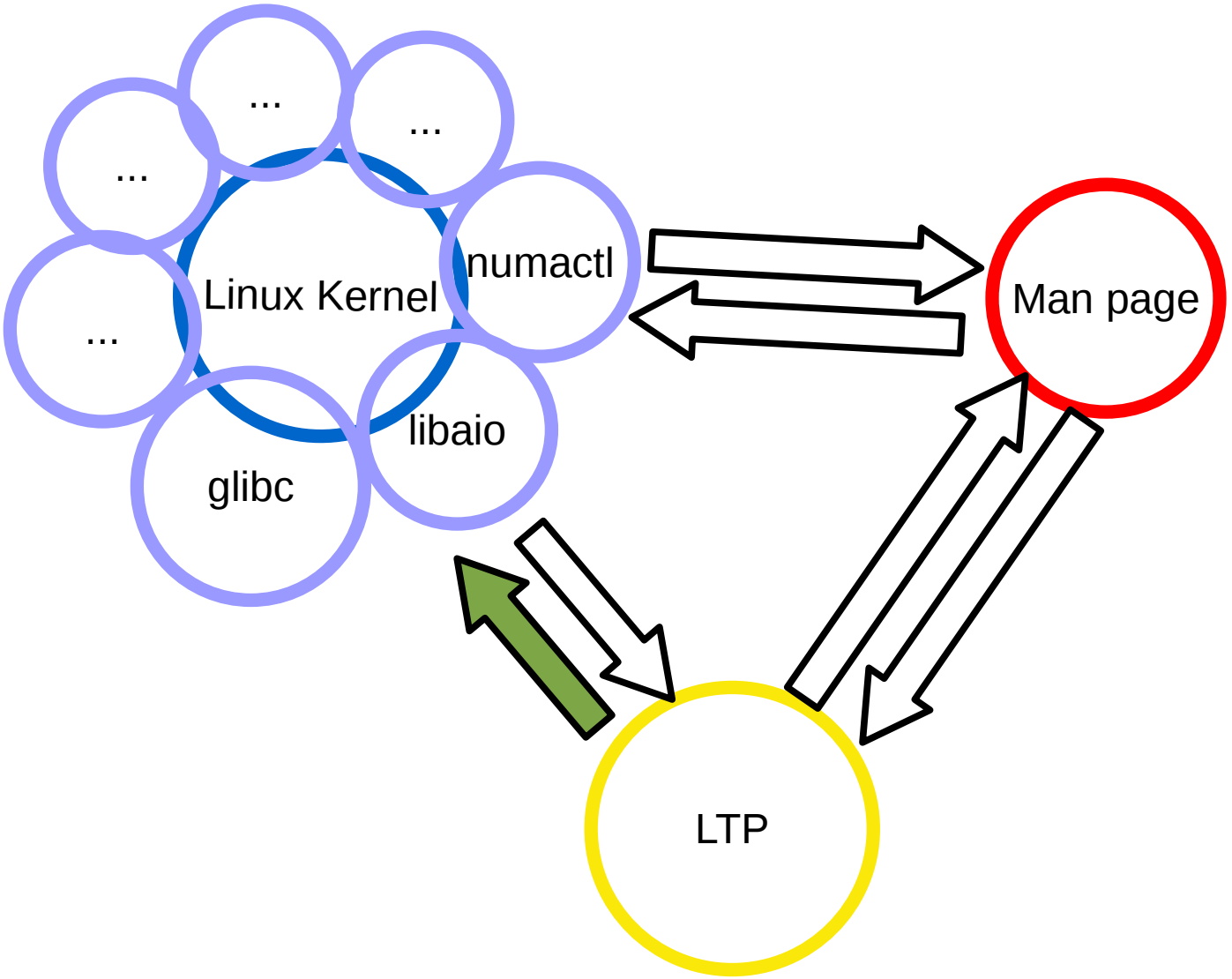
If one adds new function to kernel, s/he is the most knowledgeable person about it on the earth.

Ecosystem: *My practice*

Ecosystem: *My practice 1*

A'. Test developers can hunt bugs of Kernel

A'. Test authors can hunt/report bugs of the kernel and libraries around it.



Tests put pressure on kernel code. As the result kernel discloses bugs.

A bug report about my test case

The name of system call for which I wrote test cases

```
From: Subrata Modak <subrata@li...> - 2008-01-05 17:55
Yamato,

Can you look in to the issue of failure of posix_fadvise.

--Subrata

On Fri, 2008-01-04 at 16:08 +0530, pramod gurav wrote:
> hi sir,
> i earlier had informed about the failing tests on omap board...
> i had sent you the system information n detailed test outputs as said
> by you...
> but did not recevied any reply..
> so sending them again...
> please help me out in this..
> system information...
> [Pramod@localhost ltpstormresult]$ uname -a
> Linux localhost.localdomain 2.6.15Pramod #1 Wed Oct 17 11:39:43 IST
> 2007 i686 i686 i386 GNU/Linux
>
> board information...
> Kernel Version: 2.6.19-omap1
> Machine Architecture: armv6l
```

Where is the bug?

- in my test case,
- in man page I referred to write the test case,
- in run time,
  - glibc, or
  - kernel?

The bug was in kernel

Subject: - check-advice-of-fadvice64_64-even-if-get_xip_page-is-given.patch removed from -mm tree
From: akpm@linux-foundation.org
To: yamato@redhat.com, cotte@de.ibm.com, mm-commits@vger.kernel.org
Date: Tue, 05 Feb 2008 14:32:24 -0800

The patch titled
    check ADVICE of fadvise64_64 even if get_xip_page is given
has been removed from the -mm tree. Its filename was
    check-advice-of-fadvice64_64-even-if-get_xip_page-is-given.patch

This patch was dropped because it was merged into mainline or a subsystem tree

The current -mm tree may be found at http://userweb.kernel.org/~akpm/mmotm/

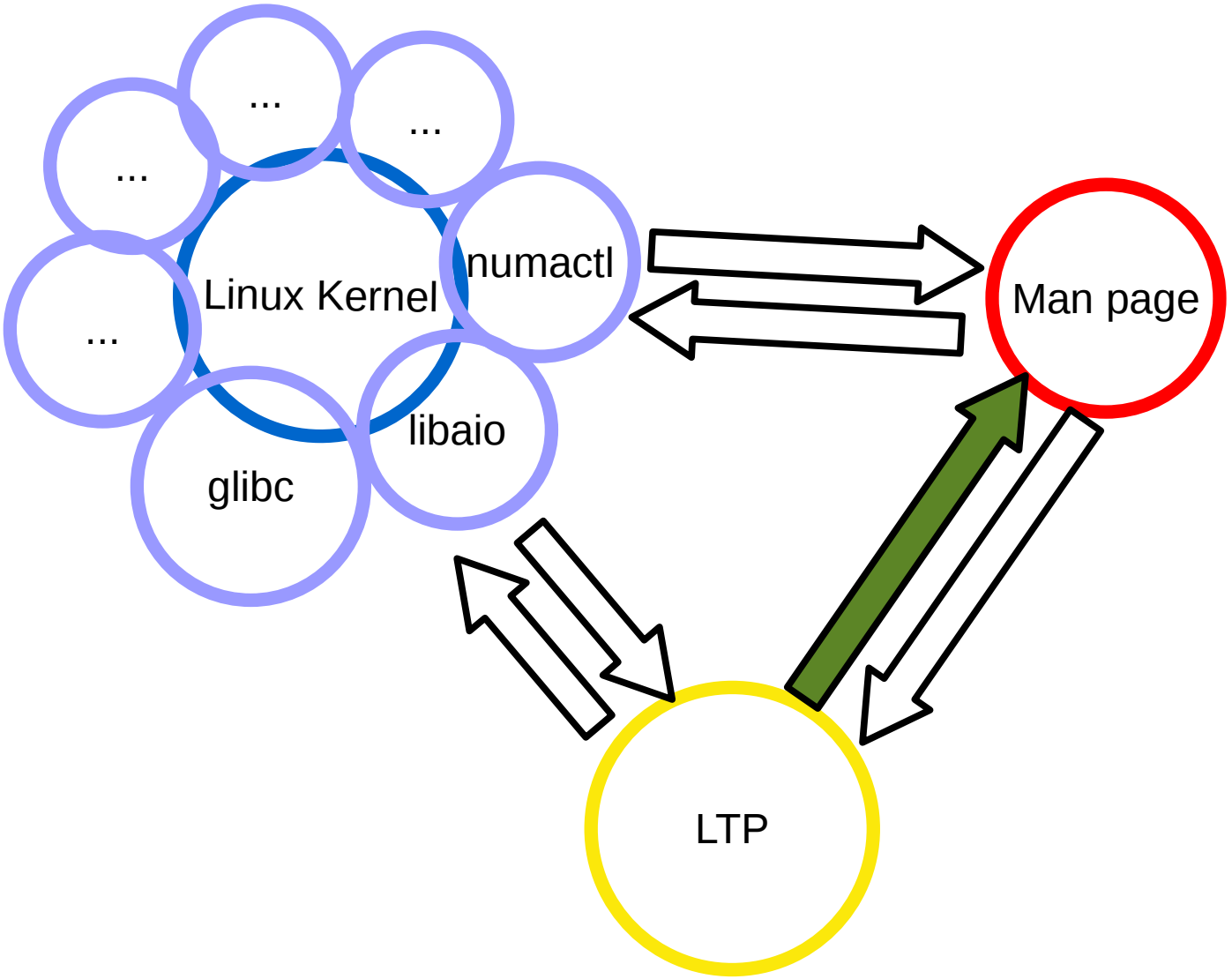
Subject: check ADVICE of fadvise64_64 even if get_xip_page is given
From: Masatake YAMATO <yamato@redhat.com>

I've written some test programs in ltp project. During writing I met an problem which I cannot solve in user land. So I wrote a patch for linux kernel. Please, include this patch if acceptable.

Ecosystem: *My practice 2*

B'. A test developer can fix typos in man pages

B'. Test authors can review man pages and fix typos in them.



To write test case, the authors have to read man pages very carefully.

Strange code found during porting a test case

Quoted from a test case for io_cancel in crackerjack:

```
/*
        EAGAIN The iocb specified was not cancelled.

        ENOSYS io_cancel is not implemented on this architecture.
*/

                ENTER(0);
                if ( io_cancel( ctx, NULL, NULL ) == -ENOSYS ) {
                        PASS;
                } else {
                        FAIL;
                }
                Tst_count++;
```

Copied from man page of io_cancel

Test code to trigger ENOSYS error

Strange code found during porting a test case

Quoted from a test case for io_cancel in crackerjack:

```
/*
        EAG                                    not cancelled.

        ENOSYS io_cancel is not implemented on this architecture.
*/

        ENTER(0);
        if ( io_cancel( ctx, NULL, NULL ) == -ENOSYS ) {
                PASS;
        } else {
                FAIL;
        }
        Tst_count++;
```

defined as POSITIVE integer in a header file

Copied from man page of io_cancel

compared the value returned from system call with NEGATIVE value.

Test code to trigger ENOSYS error

- Why the sign value for error number is inverted?
- An inconsistency exists between the man page description and test.

Where is the bug?

- my knowledge about i386 kernel and glibc:
  - In user space generally positive errno is used.
  - In kernel negative errno is used.
  - glibc inverts the sign of errno.
- so where?
  - the test code => no bug. It says “OK” when run.
  - man page
  - run time
    - libc,
    - kernel

My analysis

- Surprisingly, the C language interface for io_cancel is not part of glibc.
- It is part of libaio.
- My guessing
  - libaio uses different convention about errno from that of glibc.
  - the man page author didn't consider the difference of convention.

My report to the man page maintainer

Subject: Return value explanation about io_channel.c
From: Masatake YAMATO <yamato@redhat.com>
To: Michael Kerrisk <mtk.manpages@gmail.com>
Cc: ltp-list@lists.sourceforge.net
Date: Wed, 18 Jun 2008 17:04:20 +0900 (JST)

Dear man page maintainer. Thank you for your great work.
I'm working for test cases for io_channel(2) of linux kernel in Linux Test Project.
During writing the test cases, I found some mistakes.
Could you apply my patch to your master nroff file?

```
*** io_cancel.2.orig  2008-06-11 13:20:43.000000000 +0900
--- io_cancel.2  2008-06-18 16:46:22.000000000 +0900
***************
*** 55,70 ****
  .TP
! .B ENOSYS
  .BR io_cancel ()
  is not implemented on this architecture.
--- 55,70 ----
  .TP
! .B -ENOSYS
  .BR io_cancel ()
  is not implemented on this architecture.
```

What happens?

- After reporting I went back to my home to gave my baby a bath.
- my patch was accepted.
- I got many mails about this issue after reporting before accepted.

What really happens while I was busy at home?

- Michael might thought this is a bug of libaio.
- He contacted with the libaio maintainer and discussed this issue.
- He understood the libaio own errno convention.

What can we learn from this practice?

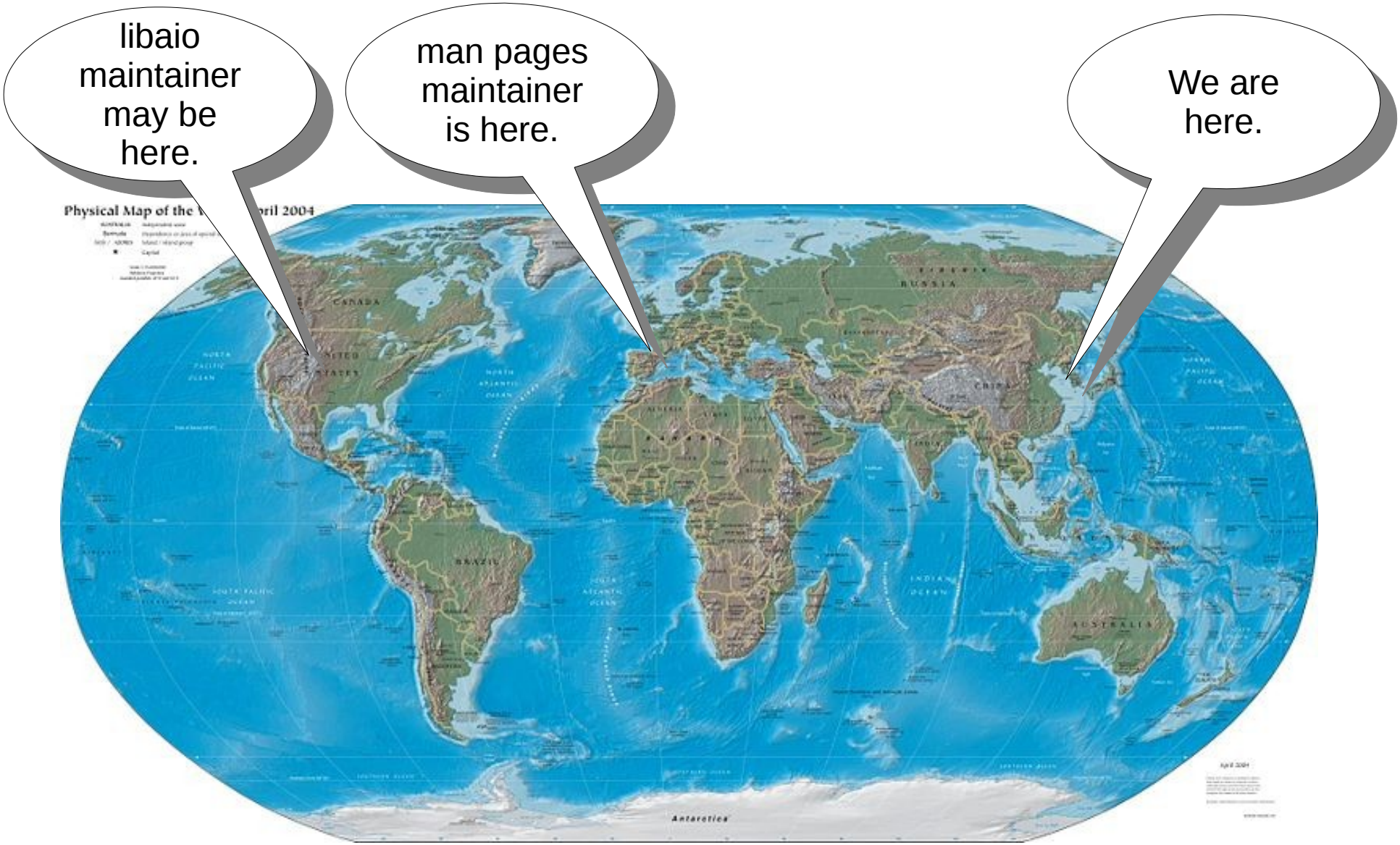


Image taken from: http://upload.wikimedia.org/wikipedia/commons/b/b3/World-map-2004-cia-factbook-large-1.7m-whitespace-removed.jpg

What can we learn from this practice?

Image taken from: http://upload.wikimedia.org/wikipedia/commons/b/b3/World-map-2004-cia-factbook-large-1.7m-whitespace-removed.jpg

Why LTP is very good project for us?

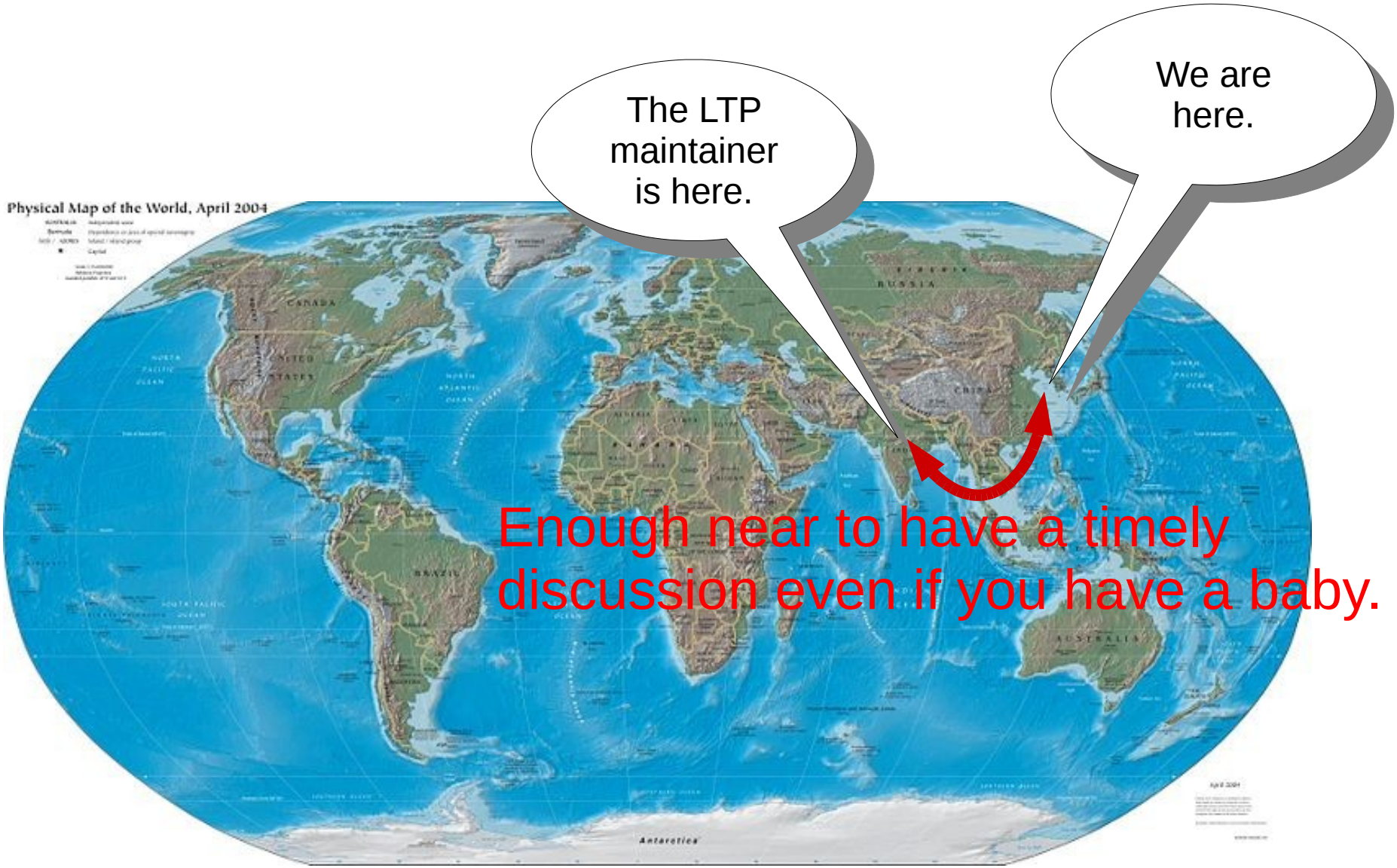
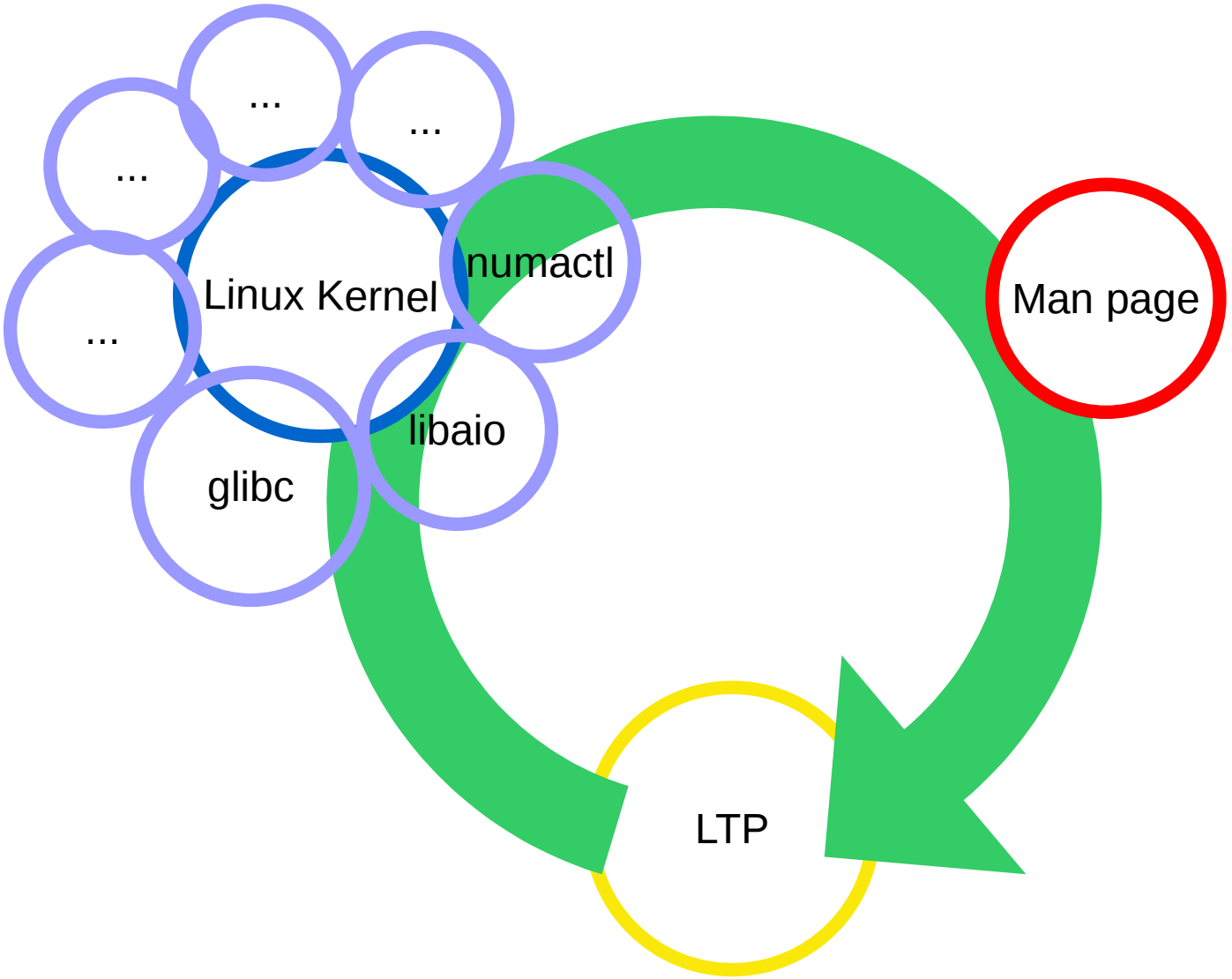


Image taken from: http://upload.wikimedia.org/wikipedia/commons/b/b3/World-map-2004-cia-factbook-large-1.7m-whitespace-removed.jpg

Join the ring. The test is a good entry point if you want small start.



Happy testing!