Ending Software Patents in Australia

Ben Sturmfels
ben@sturm.com.au
endsoftpatents.org/australia

To the Honorable the Speaker and members of the House of Representatives

This petition of Australian software industry members draws to the attention of the House the harm to society of patents that restrict computation and information processing.

Such patents actively inhibit, rather than promote innovation. For small to medium-sized software developers, it is neither viable to search patents, nor defend against patent lawsuits. The government's 2009 *Venturous Australia* report found that "in new areas of patenting such as software and business methods, there is strong evidence that existing [patent] arrangements are hampering innovation."

The Australian software industry has a long history of innovating without software patents. This shows that patents are not necessary for innovation. Examples include:

- VET (antivirus, 1989)

- Trumpet Winsock (internet connection, 1993)

- rsync (data synchronisation, 1996)

- netfilter/iptables (firewall, 1998)

Further, due to the rapid evolution of software techniques and the context in which they are used, withholding a technique for 20 years renders it effectively useless to society.

In 2008, the "Advisory Council on Intellectual Property" held a public consultation during their *Review of Patentable Subject Matter*. Microsoft Corporation was the only respondent regarding software. The Australian software industry cannot, therefore, be well-represented in the Review's findings. We were unaware of the public consultation, so could not make a submission.

We therefore ask the House to introduce and pass legislation to exclude computation and information processing activities from patentable subject matter.

Me, I'm learning a lot.

Patents are for the benefit of society.

1

Making software is faster and cheaper.

2

Software systems are comparatively complex.

3

Less overhead means more people get involved.

4

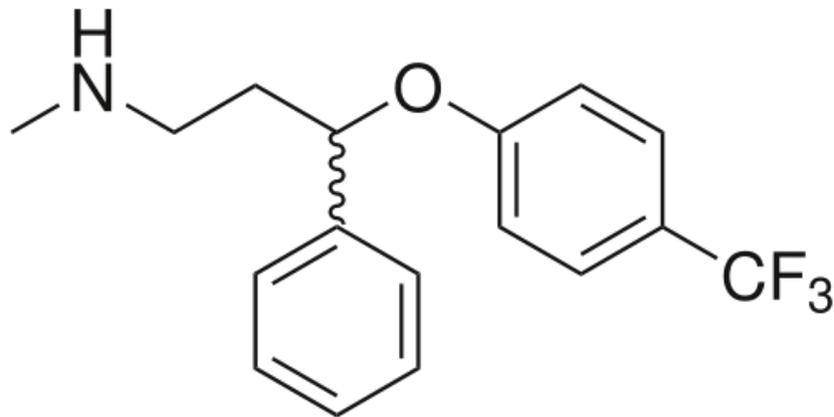## Independent invention is common.

5

Reuse is best practise.

5

Software patents cover ideas, not implementations.

A selective serotonin reuptake inhibitor

Prozac

The Fibonacci sequence

Python language

```
def fib(n):
    a, b = 0, 1
    for i in range(n):
        a, b = b, a + b
    return a
```

Haskell language

```
fib :: Integer -> Integer
fib 0 = 1
fib 1 = 1
fib n = fib (n-1) + fib (n-2)
```

6

20 years is a lifetime in computing.

7

We were doing fine without patents, thank you.

8

After 1991, innovation continues merrily without relying on patents.

Aussie inventions without patents:

- VET (antivirus, 1989)
- Trumpet Winsock (networking, 1993)
- rsync (data synchronisation, 1996)
- netfilter/iptables (firewall, 1998)

10

Computing is interesting, challenging and profitable.

The winners: mega-corporations

The losers: everyone else

venturousaustralia

Intellectual property is also critical to the creation and successful use of new knowledge – particularly the 'cumulative' use of knowledge as an input to further, better knowledge. In this regard, particularly in new areas of patenting such as software and business methods, there is strong evidence that existing intellectual property arrangements are hampering innovation. To address this, the central design aspects of all intellectual property needs to be managed as an aspect of economic policy. Arguably, the current threshold of inventiveness for existing patents is also too low. The inventive steps required to qualify for patents should be considerable, and the resulting patents must be well defined, so as to minimise litigation and maximise the scope for subsequent innovators.

Please exclude *computation and information processing* from patentable subject matter.

More information

End Software Patents in Australia campaign
endsoftpatents.org/australia

Patent Absurdity documentary
patentabsurdity.com

Ben Sturmfels
ben@sturm.com.au