

# **The Life and Journey of a Professional Open-Source Developer**

Bob McWhirter



**redhat.**

# **Note!**

At any point, ask  
any questions  
you want.

**Director of Research & Prototyping**  
JBoss Middleware, Red Hat

**Project co-lead of WildFly Swarm**

**At Red Hat for almost 10 years**

How did I get here?

# via open-source!



open source  
initiative

Studied theatre at  
university



building scenery



Moved into graphic  
design.

*On paper.*

Newsletters. Menus.  
Business cards.

Remember, this was  
1997

**Then websites...**

**And someone wanted a  
shopping cart.**

**WTF is cgi-bin?**

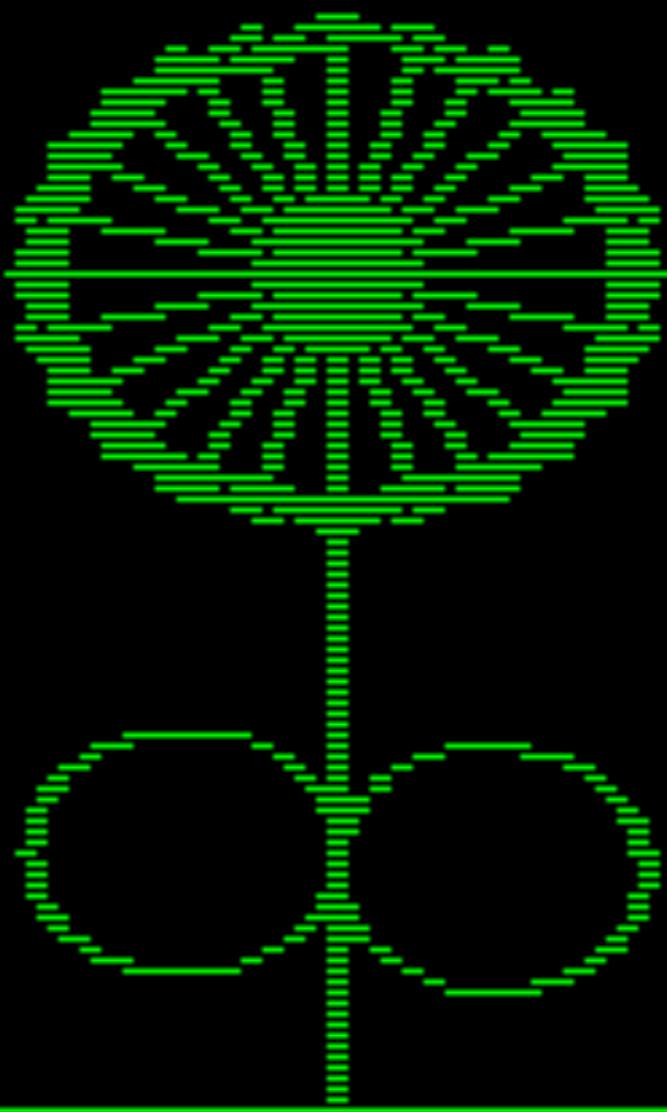
**And Apache?**

*Perl?*

You've got to be  
kidding me.

\*sigh\*

I'd done **Apple //e**  
**BASIC** in 3rd grade



And Logo



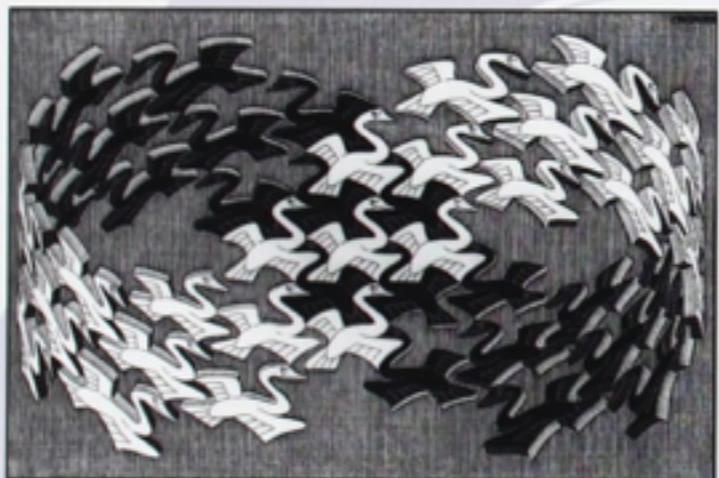
# Perl

C++

# Design Patterns

Elements of Reusable  
Object-Oriented Software

Erich Gamma  
Richard Helm  
Ralph Johnson  
John Vlissides



Cover art © 1994 M.C. Escher / Cordon Art - Baarn - Holland. All rights reserved.

Foreword by Grady Booch

ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES



**ACE**

**Washington University**



Use.  
Learn.  
Find bugs.  
**Submit patch.**

- 413. Jeff Greif <jmg at trivida dot com>
- 414. Jeff McDaniel <jsmpcdani at gte dot com>
- 415. Andreas Geisler <Andreas dot Geisler at tiscali dot de>
- 416. Bob McWhirter <bob at werken dot nl>
- 417. Daniel Winder <Daniel dot Winder at gmx dot de>
- 418. Zheng Han <zhan at glenvan dot com>
- 419. Christa Schwanninger <christa dot schwanninger at t-online dot de>
- 420. Byron Harris <byron dot harris dot b at sciweavers dot org>

# Makefiles

Not awesome. Not fast.

Academic writings are  
form of *open-source*.

# Recursive Make Considered Harmful

Peter Miller

miller@canb.auug.org.au

## ABSTRACT

For large UNIX projects, the traditional method of building the project is to use recursive *make*. On some projects, this results in build times which are unacceptably large, when all you want to do is change one file. In examining the source of the overly long build times, it became evident that a number of apparently unrelated problems combine to produce the delay, but on analysis all have the same root cause.

This paper explores a number of problems regarding the use of recursive *make*, and shows that they are all symptoms of the same problem. Symptoms that the UNIX community have long accepted as a fact of life, but which need not be endured any longer. These problems include recursive *makes* which take “forever” to work out that they need to do nothing, recursive *makes* which do too much, or too little, recursive *makes* which are overly sensitive to changes in the source code and require constant *Makefile* intervention to keep them working.

The resolution of these problems can be found by looking at what *make* does, from first principles, and then analyzing the effects of introducing recursive *make* to this activity. The analysis shows that the problem stems from the artificial partitioning of the build into separate subsets. This, in turn, leads to the symptoms described. To avoid the symptoms, it is only necessary to avoid the separation; to use a single *make* session to build the whole project, which is not quite the same as a single *Makefile*.

This conclusion runs counter to much accumulated folk wisdom in building large projects on UNIX. Some of the main objections raised by this folk wisdom are examined and shown to be unfounded. The results of actual use are far more encouraging, with routine development performance improvements significantly faster than intuition may indicate, and without the intuitively expected compromise of modularity. The use of a whole project *make* is not as difficult to put into practice as it may at first appear.

Simon McClenahan wrote:

>  
> "Recursive Make considered Harmful".  
> <http://www.canb.auug.org.au/~millerp/rmch/recu-make-cons-harm.html>  
<snip>  
> Well, that's all I got for now. I would really like to hear from others on  
> how they implemented this and compare it to mine. What I've done so far is  
> still not ready for production work because I've still got another 120  
> makefiles to create! Having a makefile generator tool to do this would be  
> awesome. Have a look at my attachment and let me know what you think.  
>

Check out Bob McWhirter's "bake", a make/gmake replacement inspired by the "Harmful" paper:

<http://www.werken.com/people/bob/bake/>

--

---

Tres Seaver [tse...@palladion.com](mailto:tse...@palladion.com) 713-523-6582  
Palladion Software <http://www.palladion.com>

Simon McClenahan wrote:

>  
> "Recursive Make considered Harmful".  
> <http://www.canb.auug.org.au/~millerp/rmch/recu-make-cons-harm.html>  
<snip>  
> Well, that's all I got for now. I would really like to hear from others on  
> how they implemented this and compare it to mine. What I've done so far is  
> still not ready for production work because I've still got another 120  
> makefiles to create! Having a makefile generator tool to do this would be

## Check out Bob McWhirter's "bake",

"Harmful" paper:

<http://www.werken.com/people/bob/bake/>

--

---

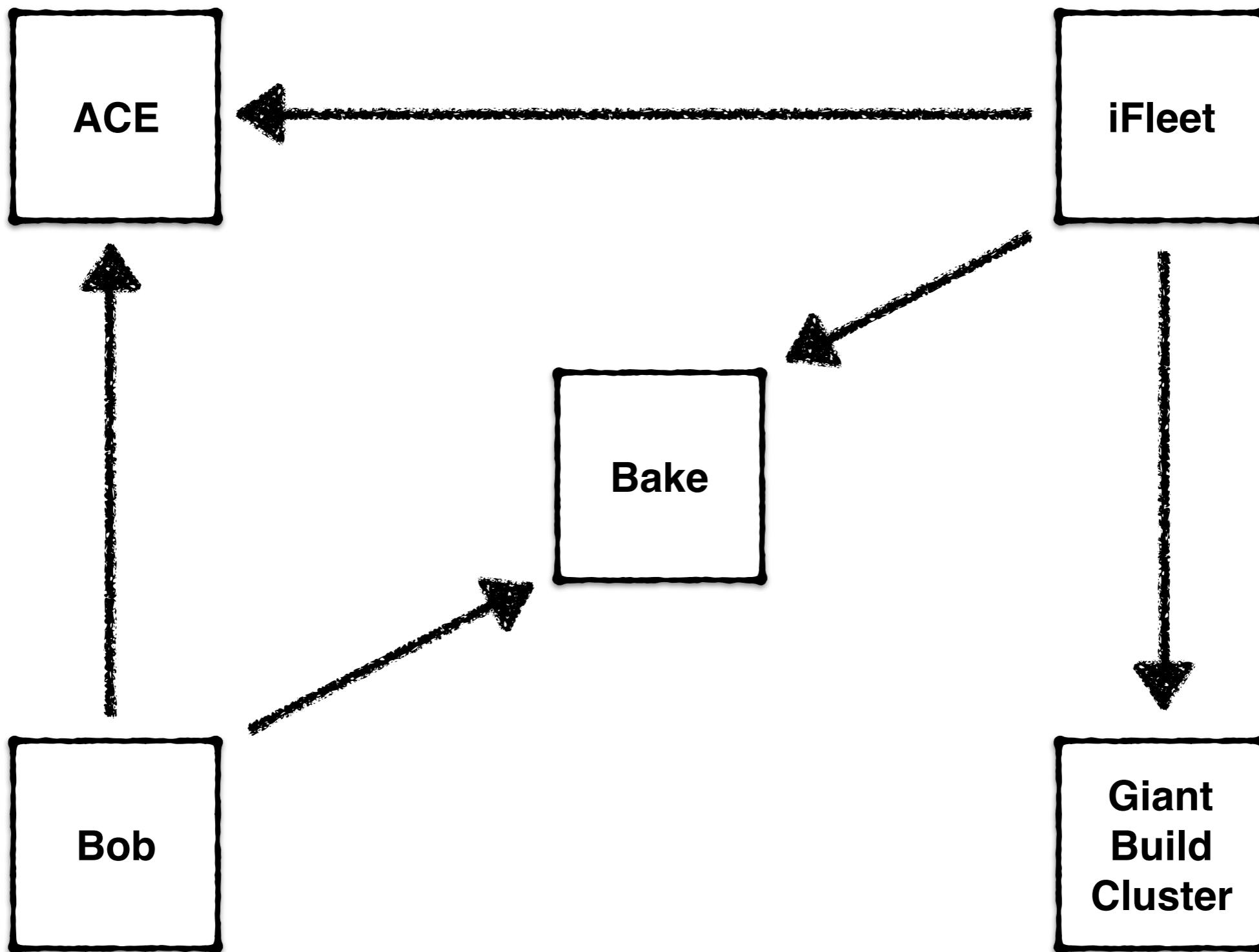
Tres Seaver      [tse...@palladion.com](mailto:tse...@palladion.com)   713-523-6582  
Palladion Software <http://www.palladion.com>

Wait, what happened to  
my hammer?

Or menus? Or business  
cards?

Or websites?

First  
*Software Engineering*  
Job



# Hired!

Linux  
Emacs  
C++ Java  
Loved Open-Source  
Released Open-Source

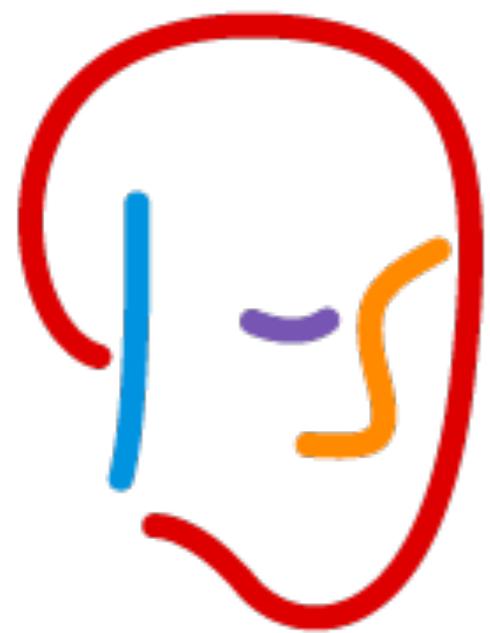
werkflow  
JDOM  
dom4j  
Jaxen XPath  
jakarta-\* (apache)

Second  
*Software Engineering*  
Job

# CAPS Logistics

The Java Guy at a  
DCOM shop.

Lots of free time.



Drools

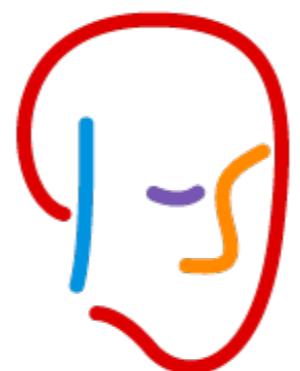
- Business rules
- Heuristics for solving vehicle routing & scheduling
- ILOG JRules was too expensive
- Lots of academic papers available



- Strong project leaders
- Autonomy
- Better tools (JIRA, Subversion)
- Commercial-friendly
- **Not Apache Software Foundation**



**Stomp** 



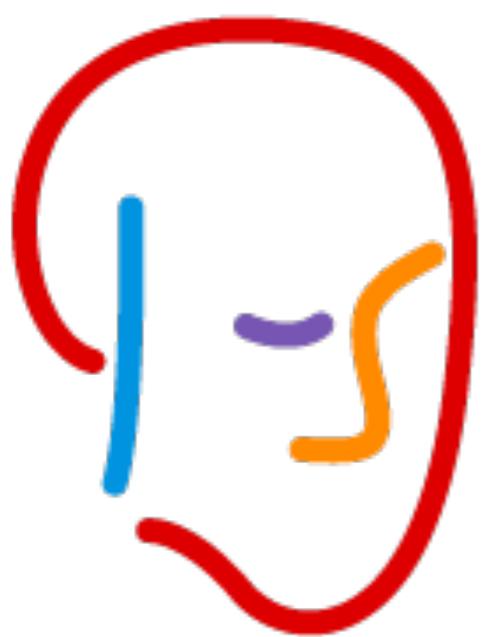
**Drools**



**GeoTools**



- Consulting around open-source strategies
- Helped clients (JotSpot, BEA) work with OSS communities



Drools



redhat.<sup>®</sup>



The simple, modern and productive way to build apps and infrastructure.

## GET STARTED

*Start using JBoss today with free  
downloads, tutorials, and quickstarts*

[GET STARTED](#)

## EXPLORE JBOSS

*Learn more about the Red Hat®  
JBoss® Middleware portfolio*

[LEARN MORE](#)

# RED HAT JBoss DEVELOPER STUDIO

## BUZZ

[Red Hat JBoss Fuse/A-MQ - Fuse and A-MQ Version 6.3 GA is released!](#)

Christina Lin

about an hour ago

Fuse and A-MQ 6.3 GA has just went out. Maybe, you would think this is just only a minor version

[View All Buzz »](#)

[Installing Open Shift Origin and Go for a Development Environment](#)

John Mazzitelli

about 12 hours ago

The Hawkular team is doing some work to develop a Hawkular Agent for running within an

## Events

[View All Events](#)

Sep  
18-22

[JavaOne 2016](#)

Join Red Hat at JavaOne San Francisco again this year. As usual, you will hear from some of JBoss's presenters and we will be present on the exhibition floor.

I missed engineering

# TorqueBox



JRuby on JBossAS

# Hire the JRuby guys





**Microservices with WildFly**

# **Lessons Learned:**

Start with small  
contributions.

Documentation and  
bug reports are  
contributions.

Contributing to  
forums & lists  
are contributions.

Implement from  
specifications or RFCs.

Strong opinions,  
Weakly held.

Scratch an itch.

Someone else probably  
has the same itch.

Communication is  
important.

Friendliness is  
important.

You can be a  
commercial-friendly  
open-source  
zealot.

Continuous  
self-promotion.

(like this today)

Champion others.

Open-source builds  
reputation.

Closed-source jobs  
make you “go dark”.

Open-source helps you  
learn.

**Open-source helps you  
make connections.**

At Red Hat, we hire  
from the community.

Community is  
*everywhere.*

**To recap:**

- You can learn via open-source
- You can improve via open-source
- You can make a career of open-source

# Thank you!

Questions?



redhat.<sup>®</sup>