

Code as a Crime Scene



Uncovering complexity in large-scale software projects

diversify

Adam Petersen
adam.petersen@diversify.se
<http://www.adampetersen.se/>

Five Symptoms of Complexity

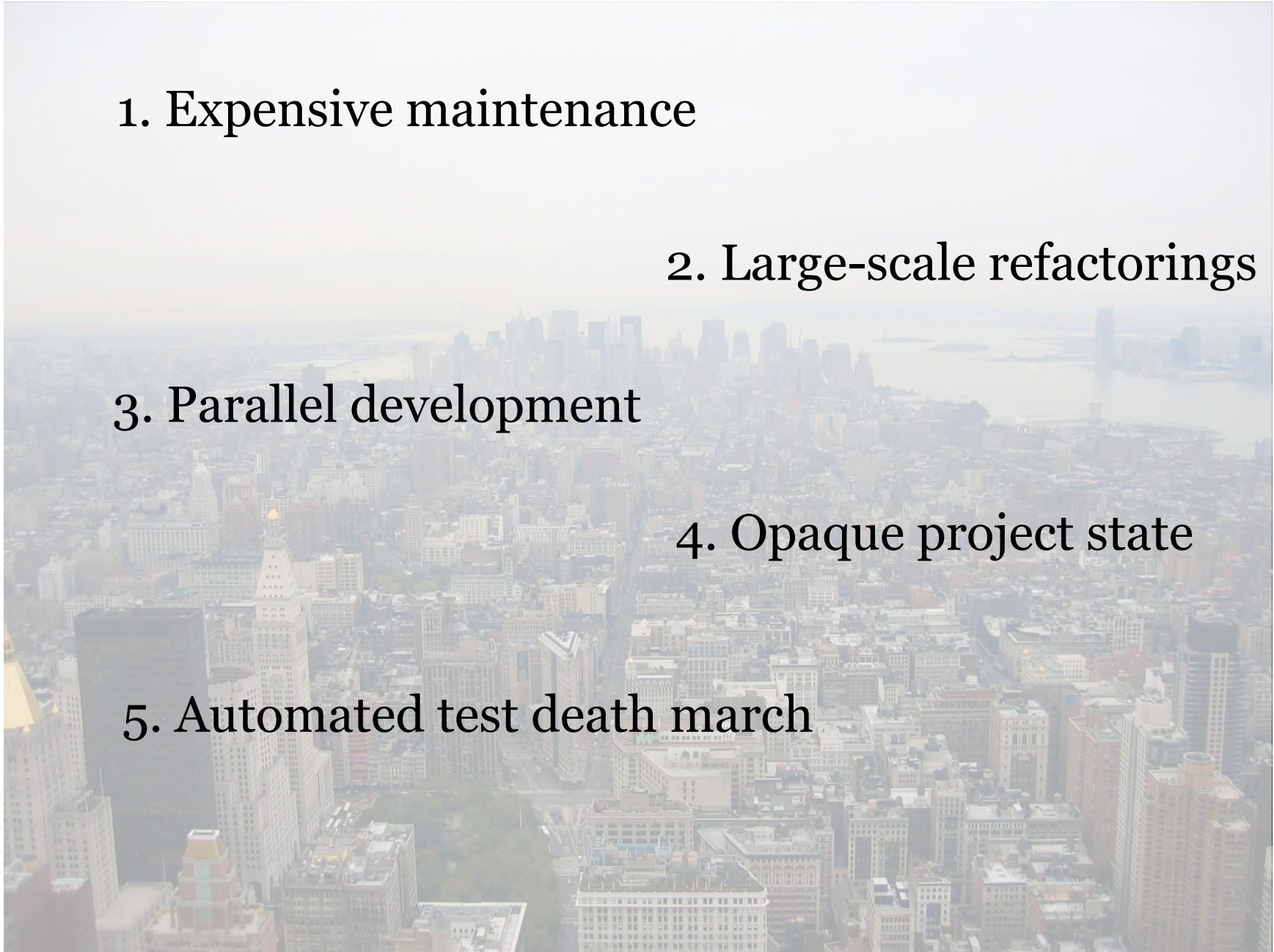
1. Expensive maintenance

2. Large-scale refactorings

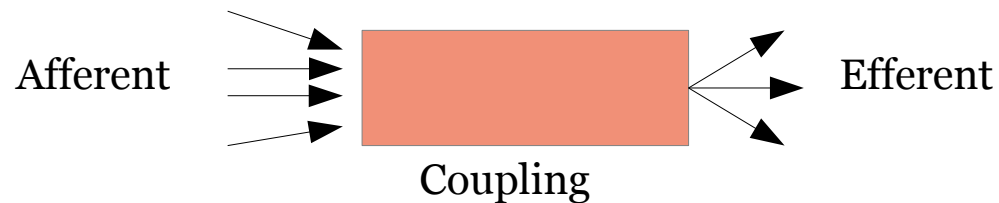
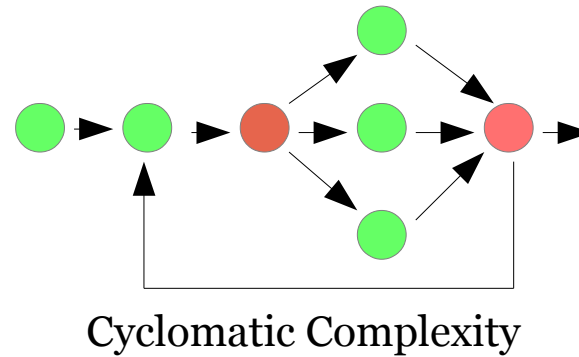
3. Parallel development

4. Opaque project state

5. Automated test death march



Complexity Measures to the Rescue?

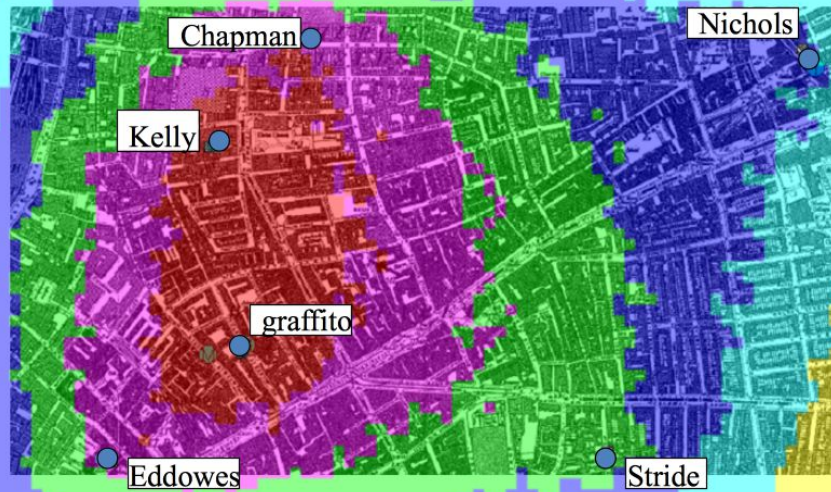


“Syntactic complexity metrics cannot capture the whole picture of software complexity” (Herraiz & Hassan)

“The use of metrics to manage software projects has not even reached a state of infancy” (Glass)

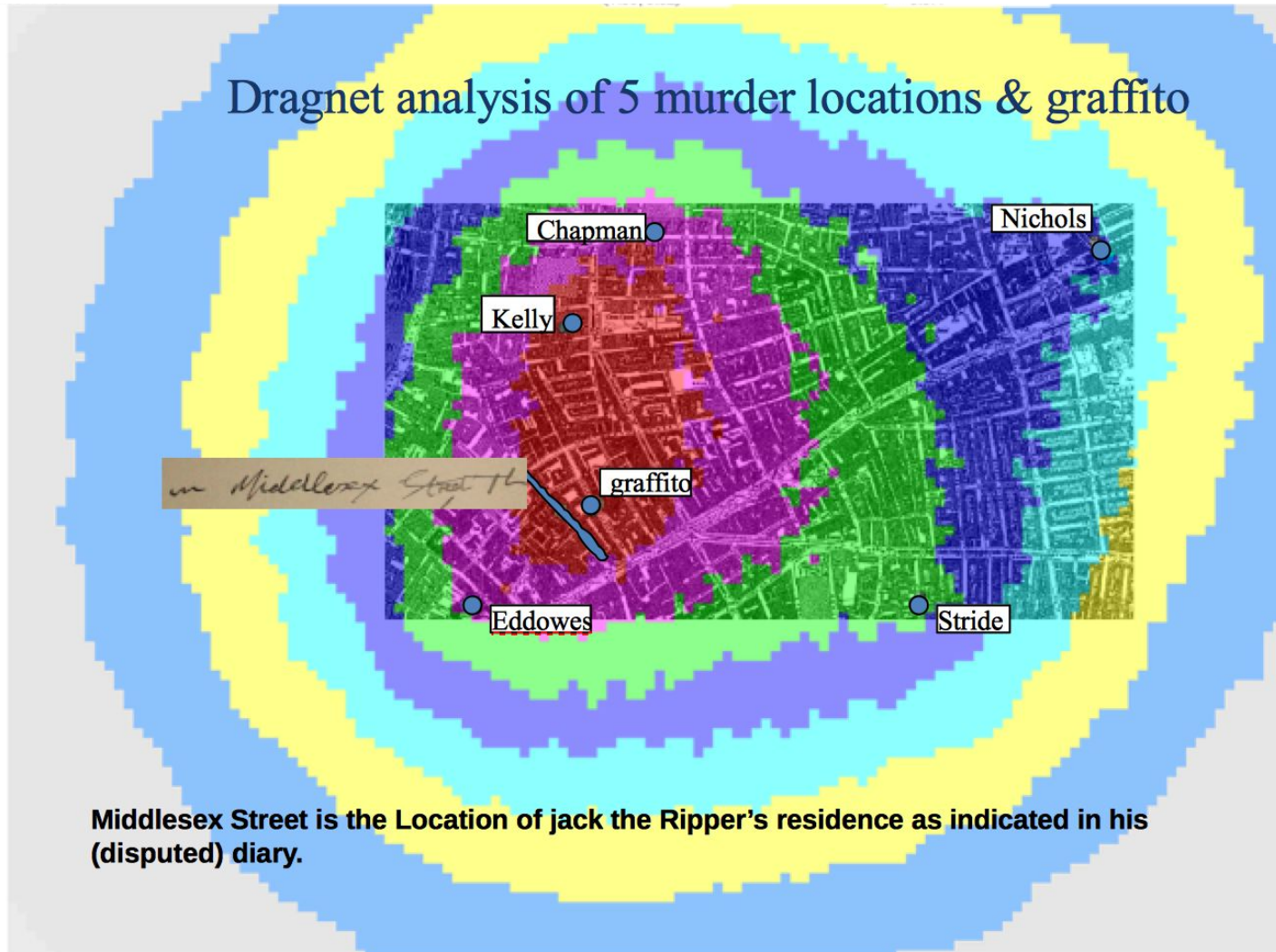
Profiling The Ripper

Dragnet analysis of 5 murder locations & graffiti

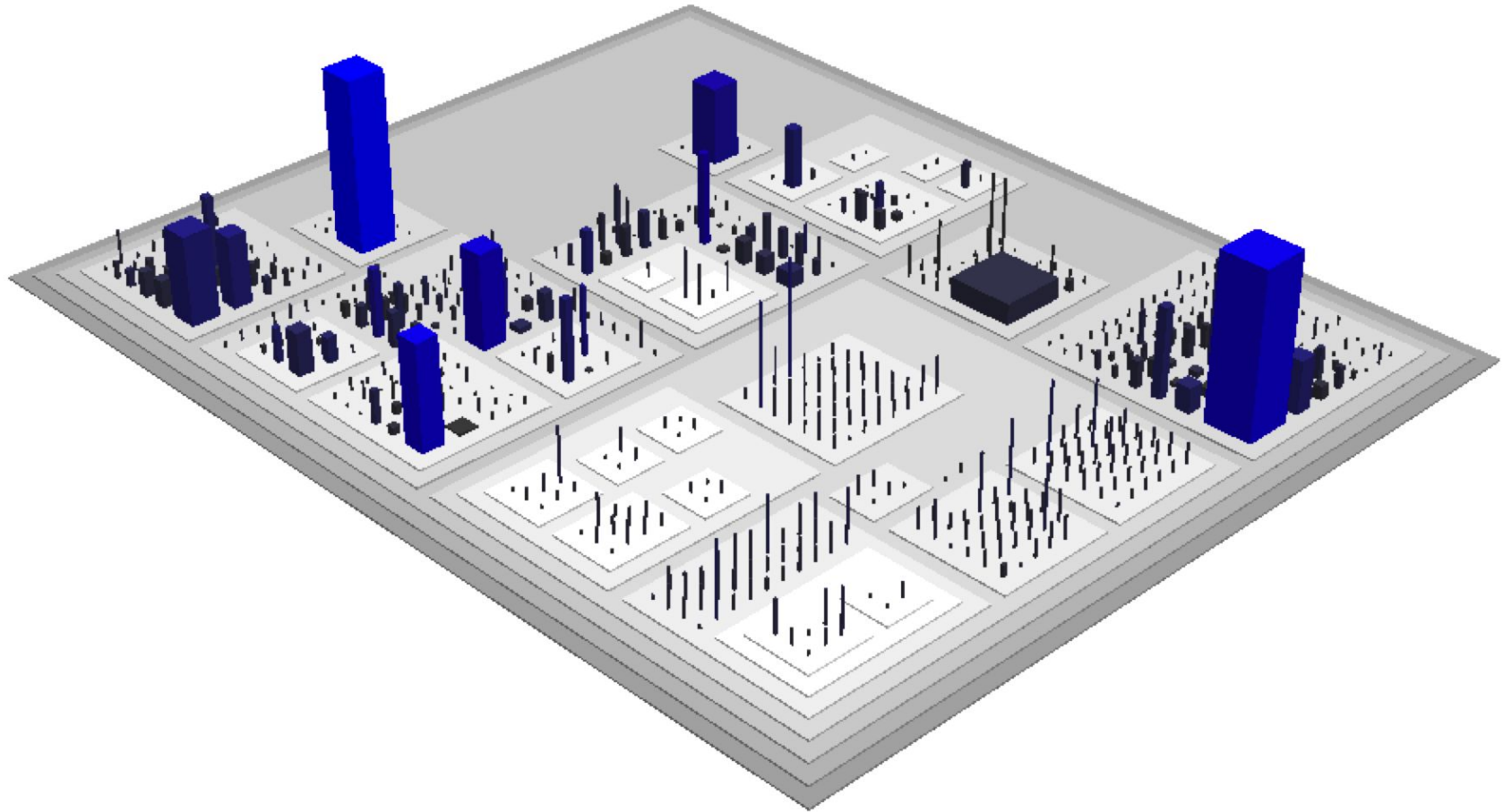


The application of Dragnet software developed by David Canter to the main crimes attributed to Jack the Ripper. Further details from dvcanter@btinternet.com www.davidcanter.com

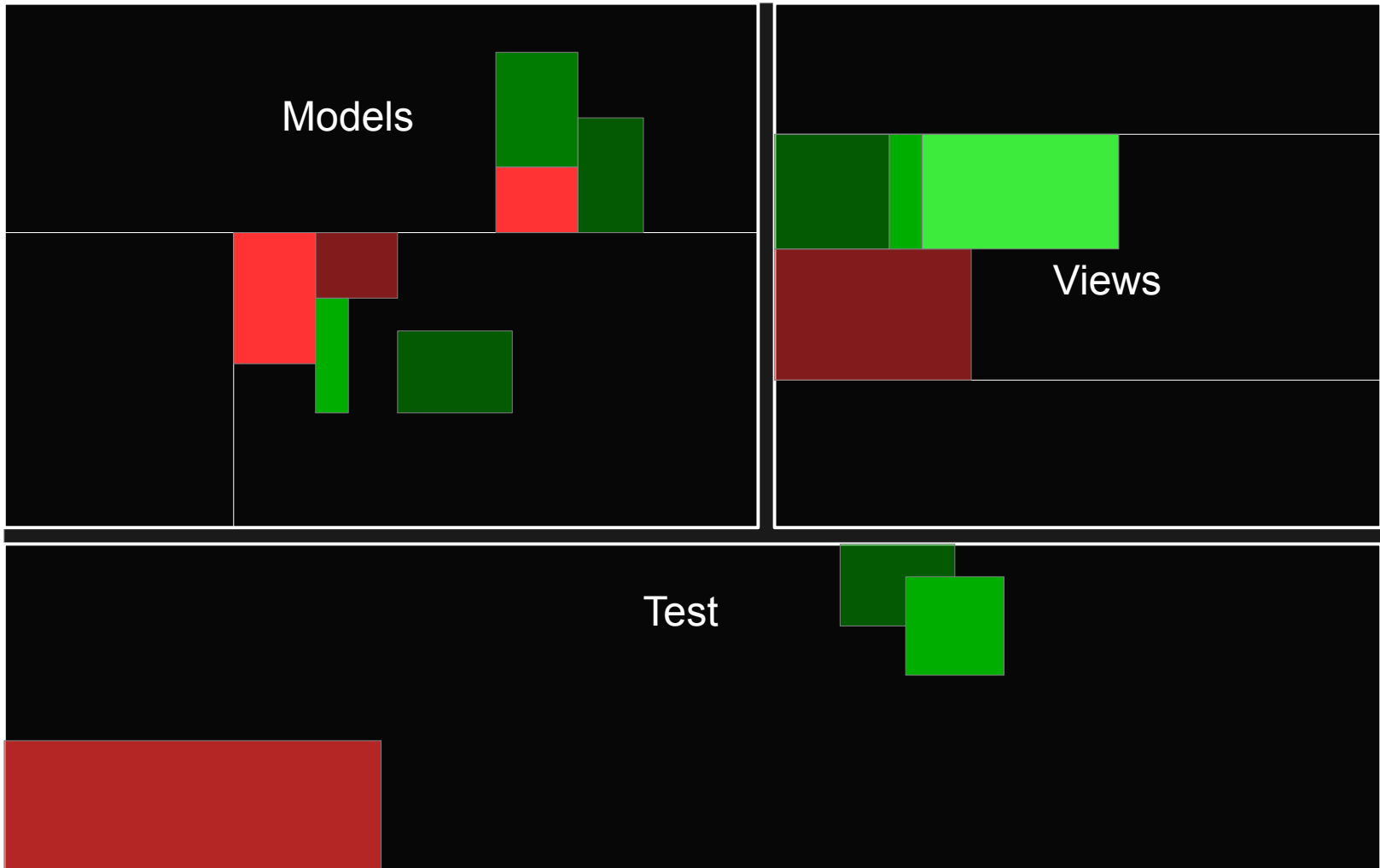
James Maybrick – The Ripper?



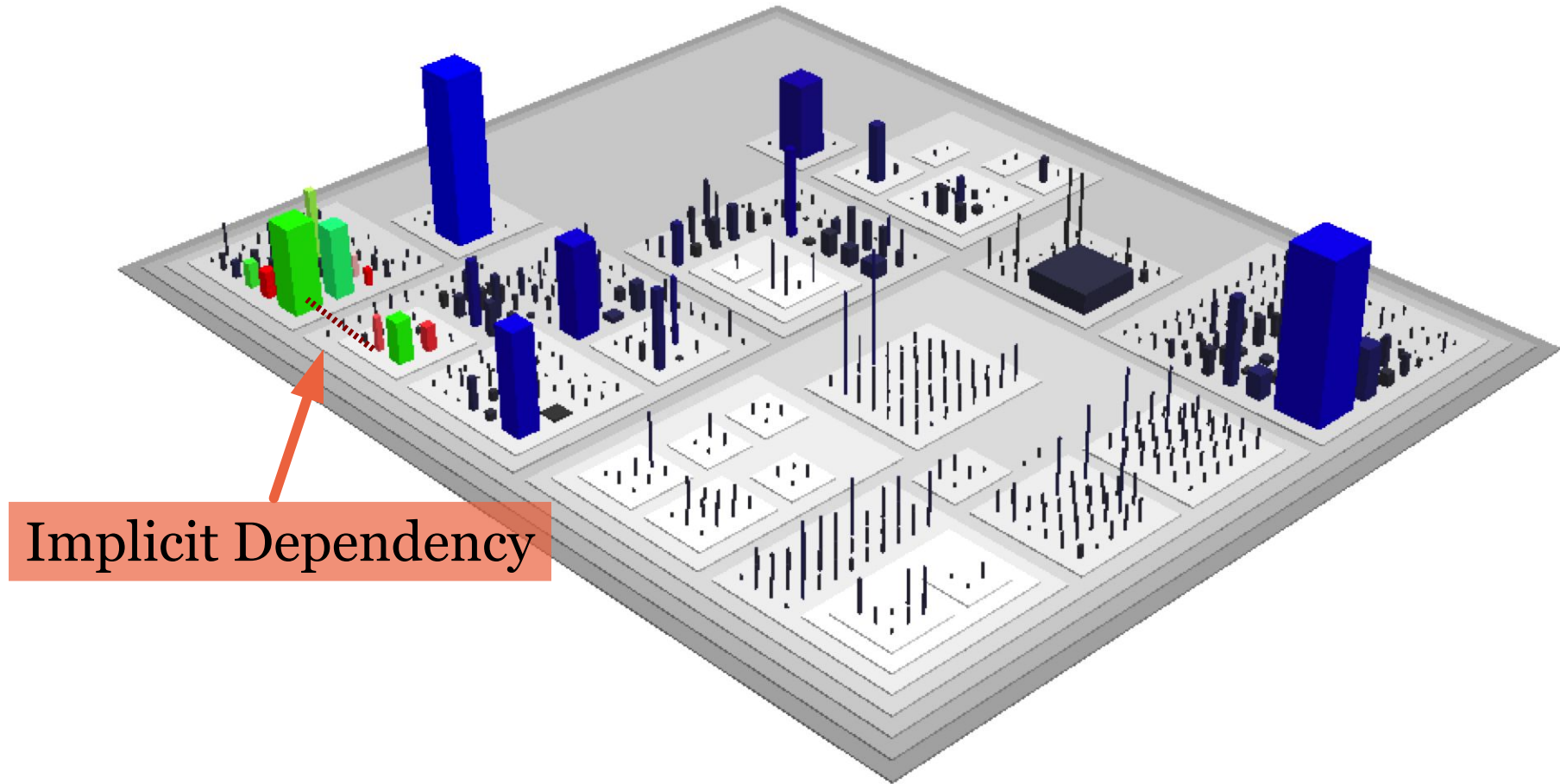
Welcome to Code City



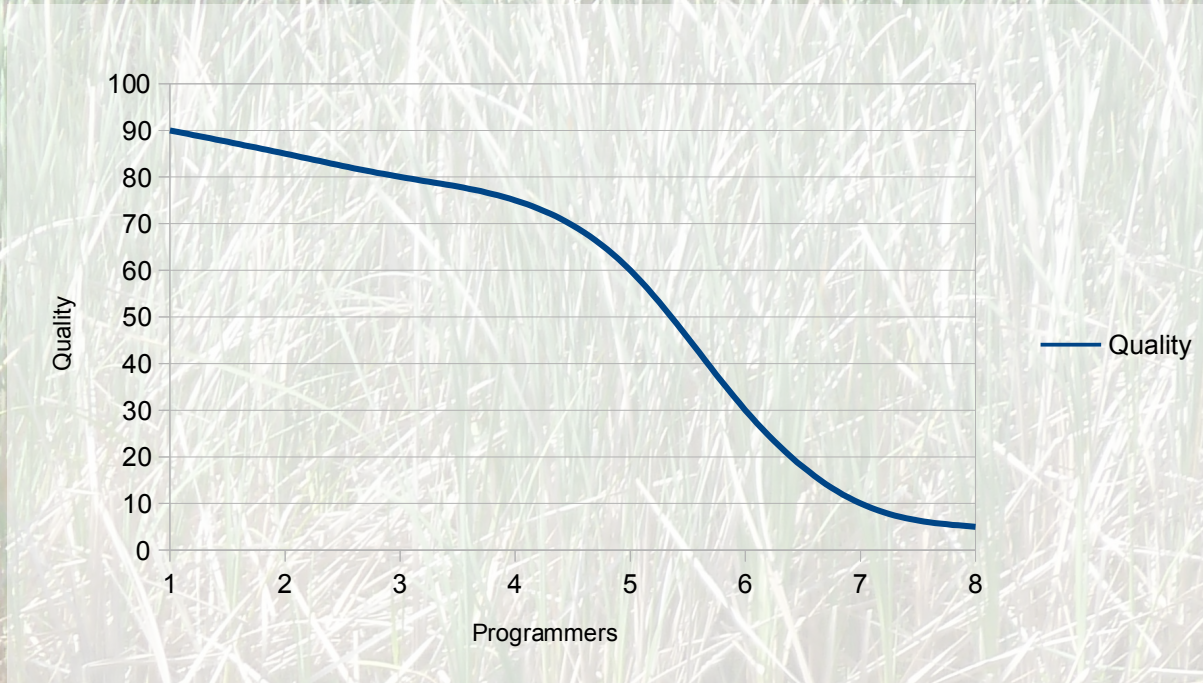
Heat Maps



Geographic Profiling of Code



Parallel Work: Quality and Defects



Interaction Patterns: Applications

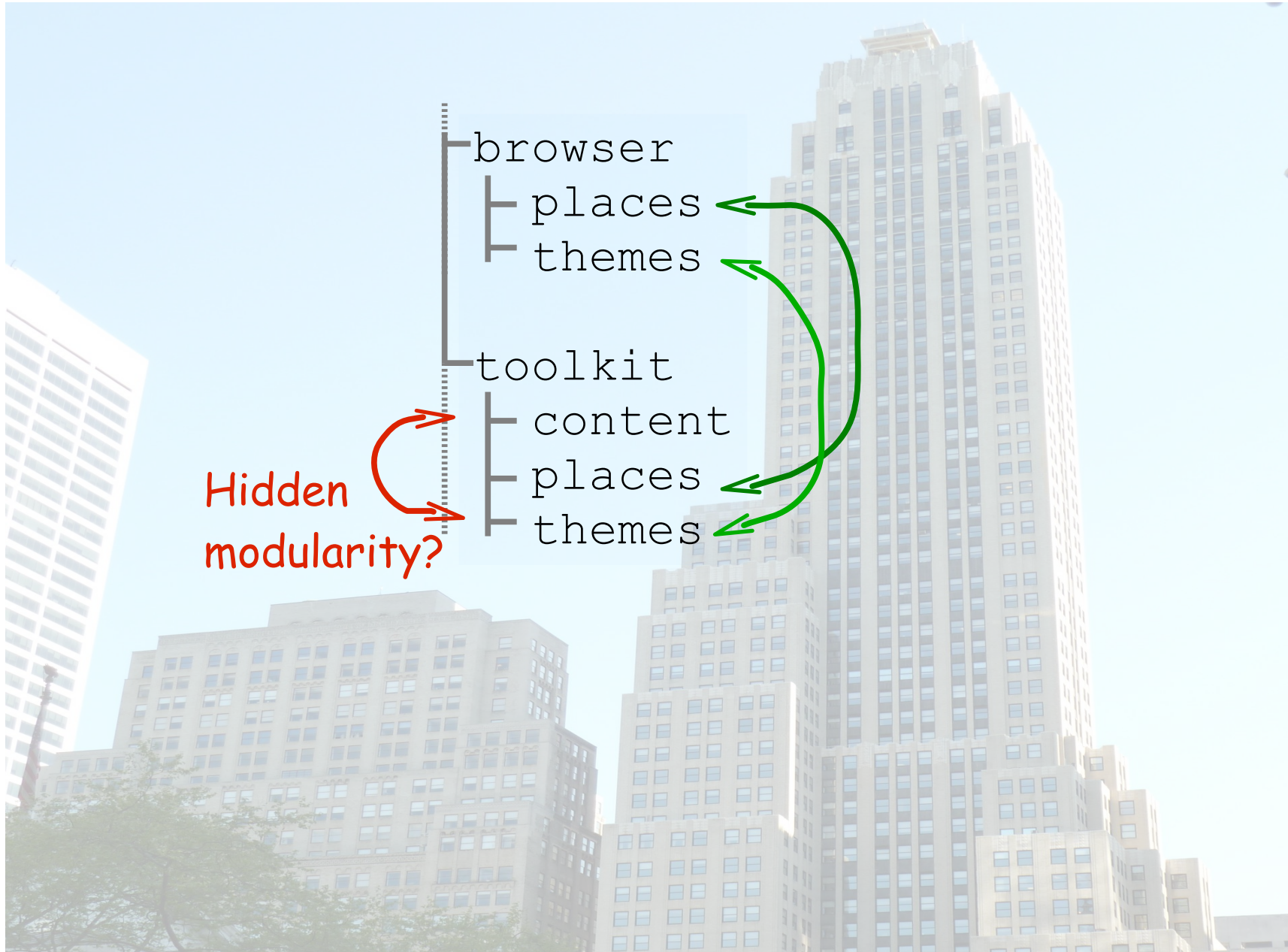


Detect

Inform

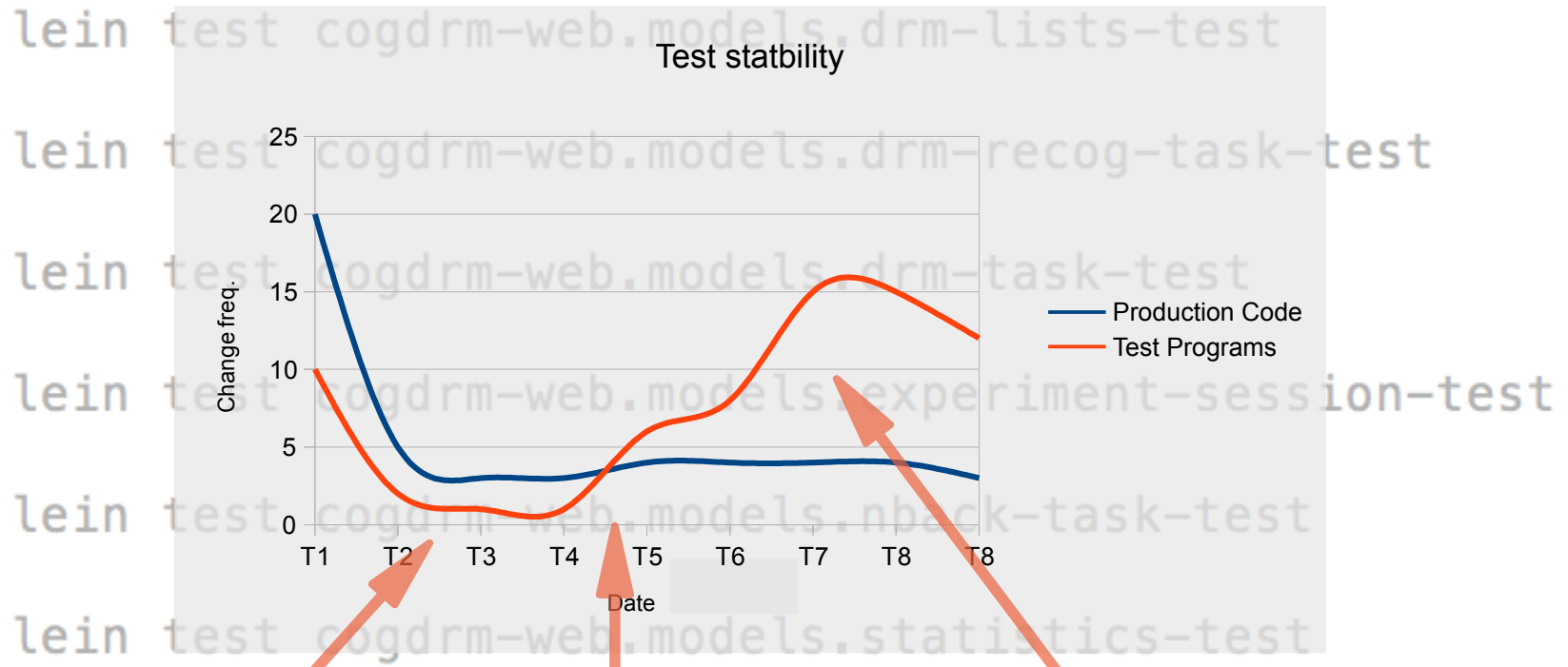
Guide

Emergent Modularity



Feedback on Automated Tests

```
zarathustra:cogdrm-web adam$ lein test
```



Normal

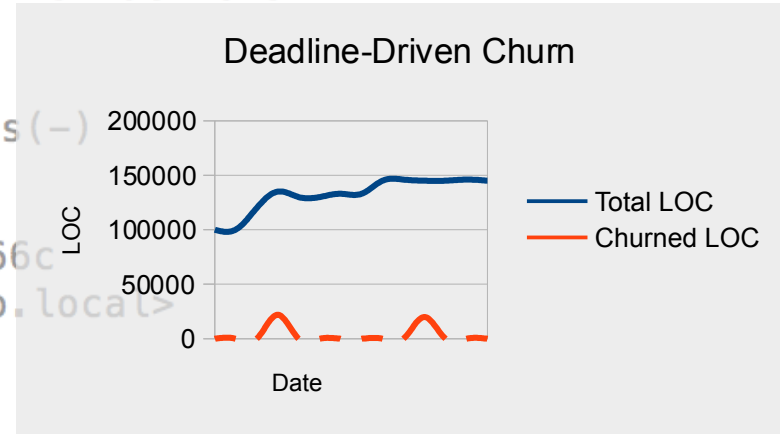
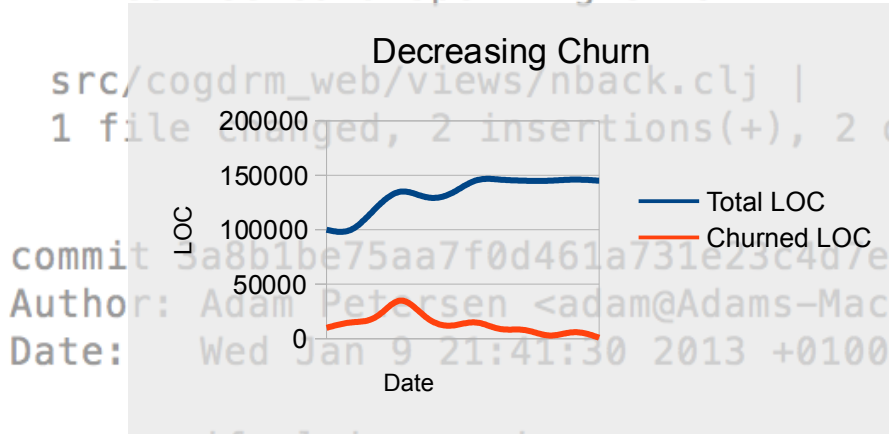
Warning

Deathmarch

```
Ran 33 tests containing 202 assertions.  
0 failures, 0 errors.
```

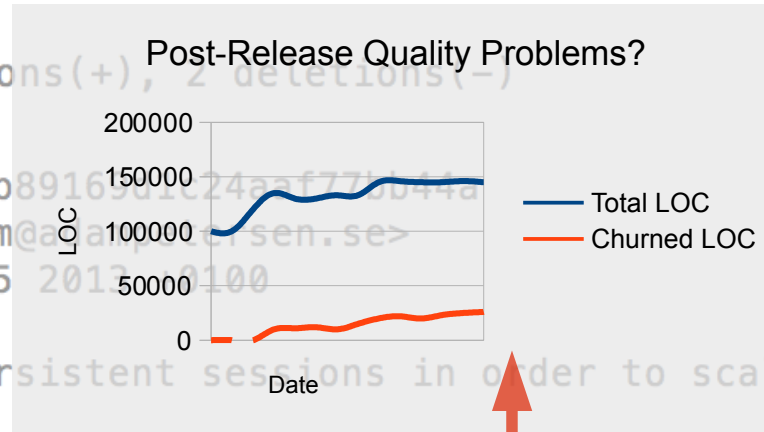
Code Churn – The DoneDone Link

Corrected a spelling error in the nback instructions



```
project.clj | 5 +++--
1 file changed, 3 insertions(+), 2 deletions(-)

commit 9fa7e32c7457092dbf4b891630124aaf77bb44a
Author: Adam Petersen <adam@adampetersen.se>
Date: Wed Jan 30 22:07:35 2013 +0100
```



Switched to MongoDB persistent sessions in order to scale on several web dynos

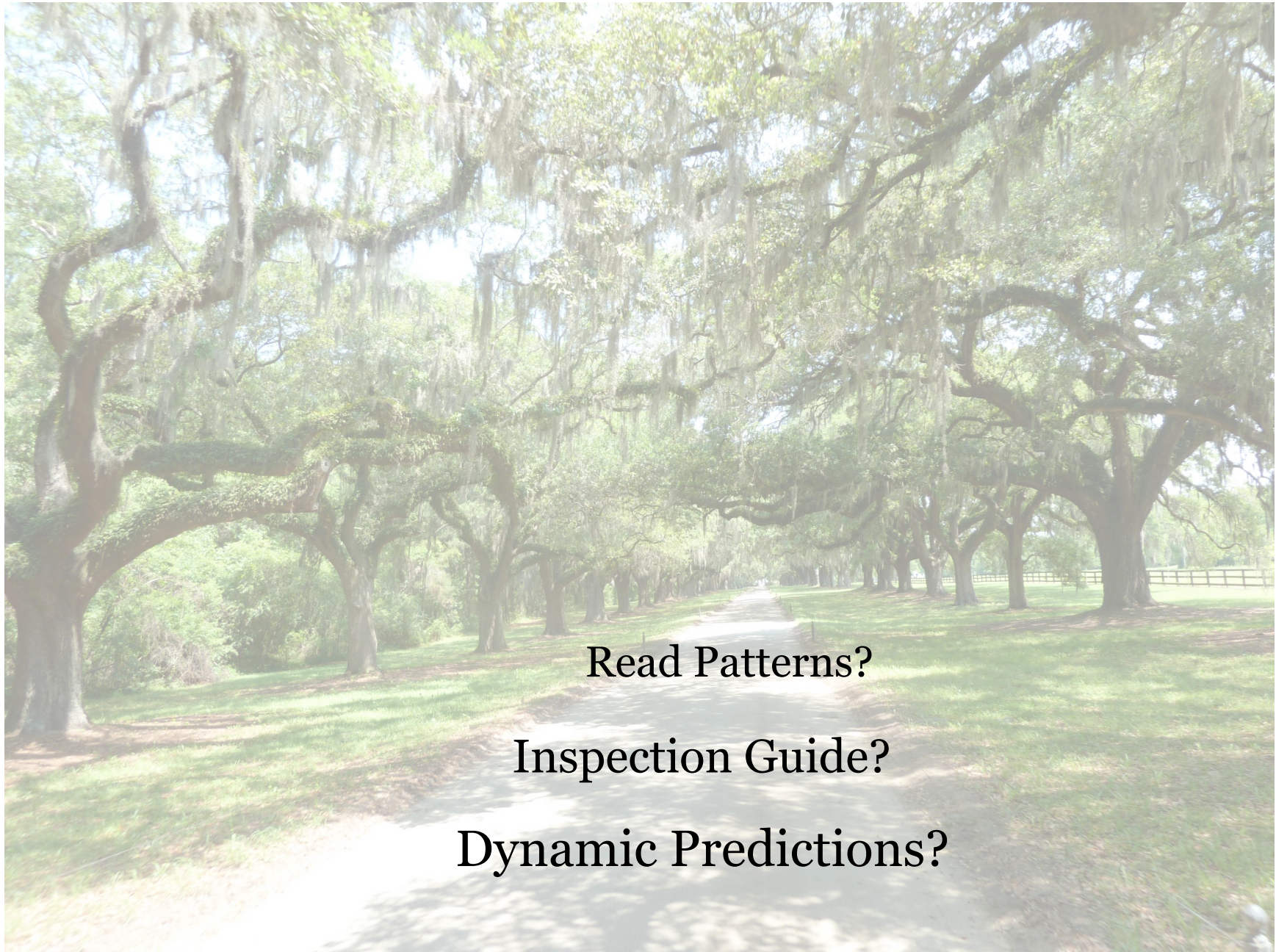
```
src/cogdrm_web/models/experiment_session.clj | 94 ++++++-----
src/cogdrm_web/models/storage.clj | 24 ++++++
src/cogdrm_web/server.clj | 1 +-
test/cogdrm_web/models/storage_test.clj | 2 +-
4 files changed, 79 insertions(+), 54 deletions(-)
```

Deadline

Modus Operandi



The Road Ahead



Read Patterns?

Inspection Guide?

Dynamic Predictions?

Thanks!



Code As A Crime Scene

diversify

Adam Petersen
adam.petersen@diversify.se
<http://www.adampetersen.se/>

References

Martin Fowler, Refactoring: Improving the Design of Existing Code, 2000

Robert L. Glass, Facts and Fallacies of Software Engineering, 2002

Israel Herraiz, Ahmed Hassan, Beyond Lines of Code: Do we need more complexity metrics? Included in Making Software, 2010.

Robert L. Glass, Software Creativity 2.0, 2006.

Perry, et. al., Parallel changes in large-scale software development, 2001.

Thomas & Murphy, How Effective is Modularization? In Making Software, 2010

Nagappan & Ball, Use of Relative Code Churn Measures to Predict System Defect Density, 2005.

Nagappan, Murphy & Basili, The Influence of Organizational Structure on Software Quality, 2008

Lewis & Ou, Bug Prediction at Google,
<http://google-engtools.blogspot.se/2011/12/bug-prediction-at-google.html>

Neil Ford, Emergent Design Through Metrics, 2009

Canter & Youngs, Principles of Geographical Offender Profiling, 2008