Automating Automation

James Koppel

Tarski Technologies

October 10, 2012

Real-World Updating

The Problem

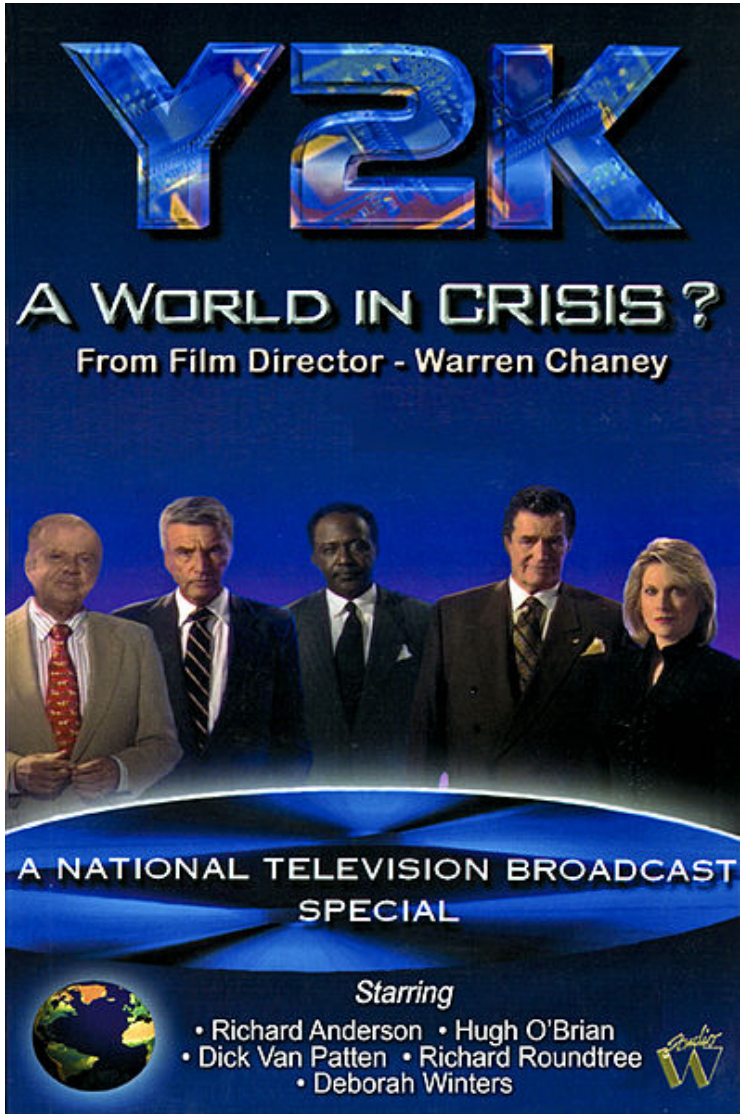
Software is currently like clay, and it needs to become like gold. Clay is soft and malleable initially, but then it hardens. [...] Gold, on the other hand, remains malleable for life.

– William Harrison

The entire edifice of modern computing rests on a fundamental irony: the software that makes it all possible is, in a very real sense, handmade.

– Moshe Vardi

The Result


Y2K
A WORLD IN CRISIS?
From Film Director - Warren Chaney
A NATIONAL TELEVISION BROADCAST SPECIAL
Starring
• Richard Anderson • Hugh O'Brian
• Dick Van Patten • Richard Roundtree
• Deborah Winters

Updating in the 21st Century

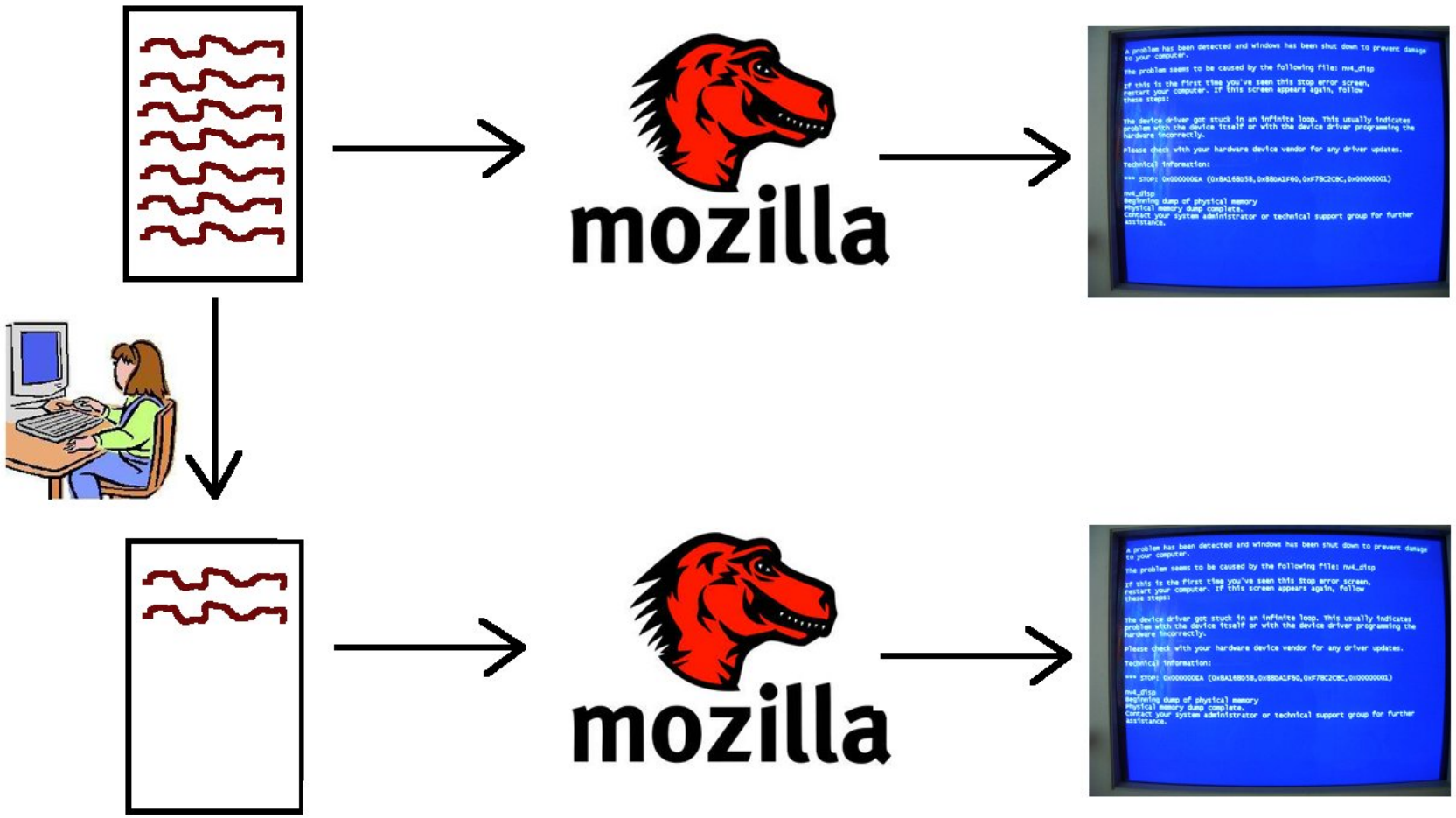
Change components $\Rightarrow$ systems break

Updating in the 21st Century

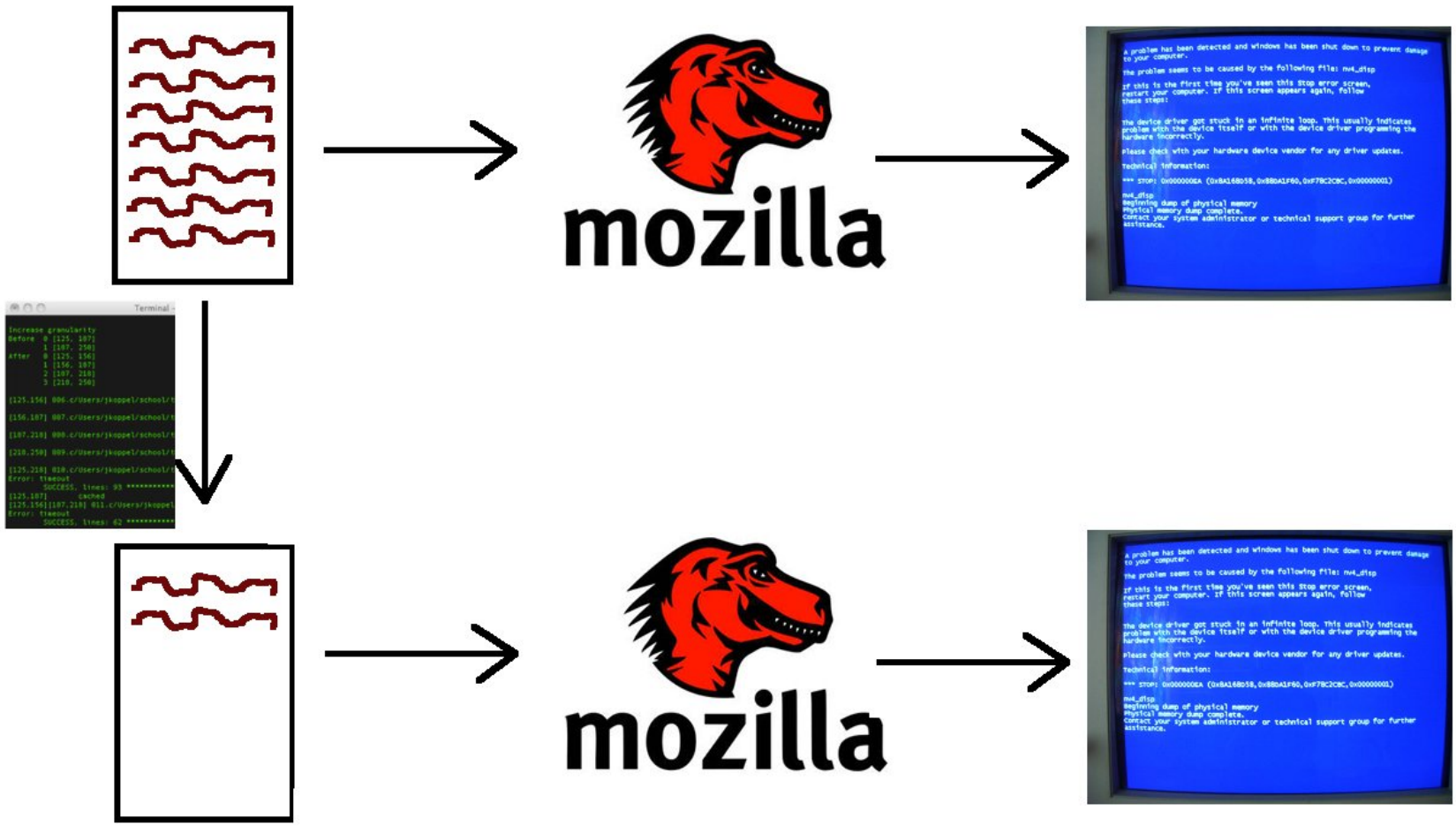
Change components ⇒ systems break

Not anymore!

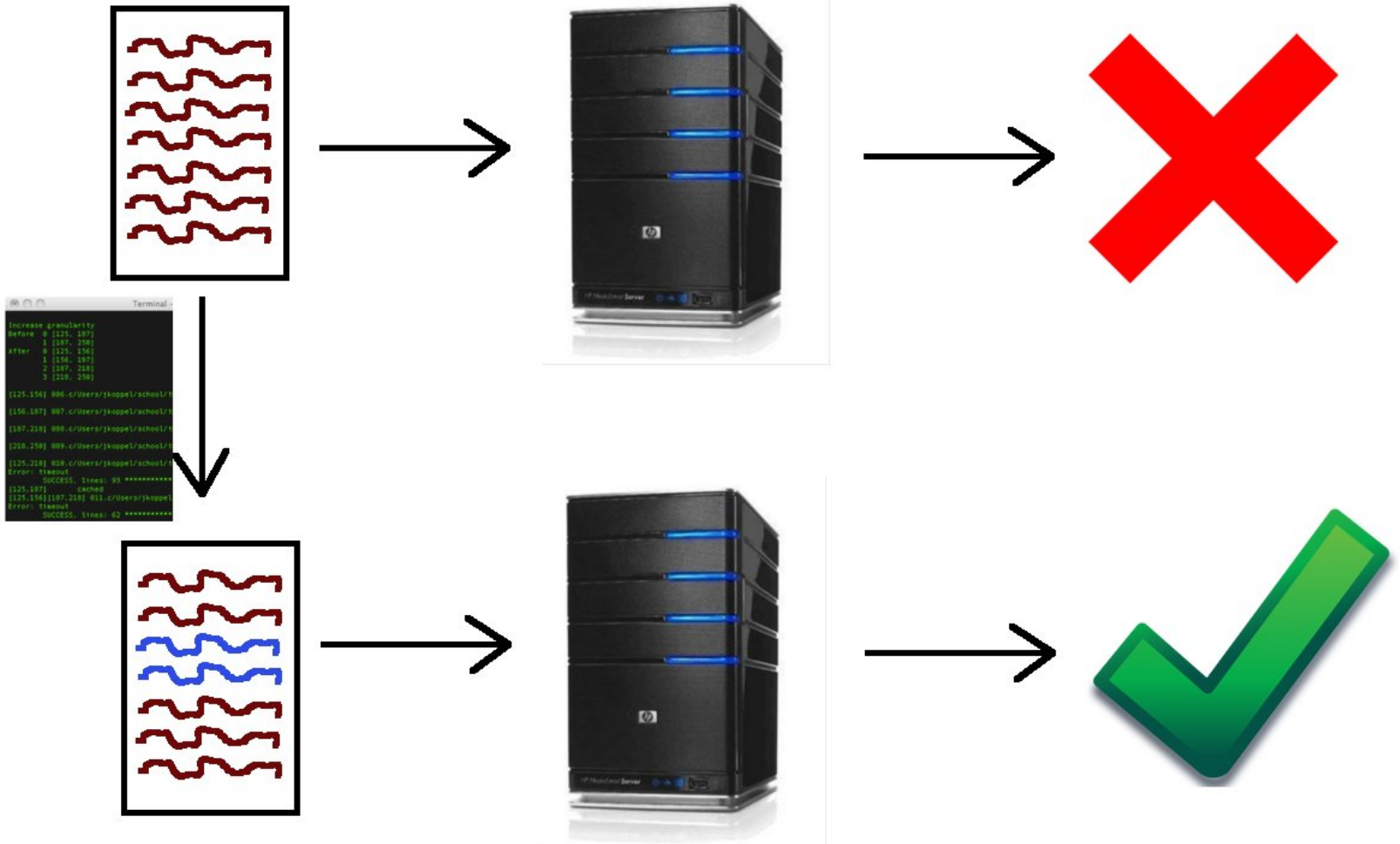
From Manual Testing to Delta Debugging



From Manual Testing to Delta Debugging

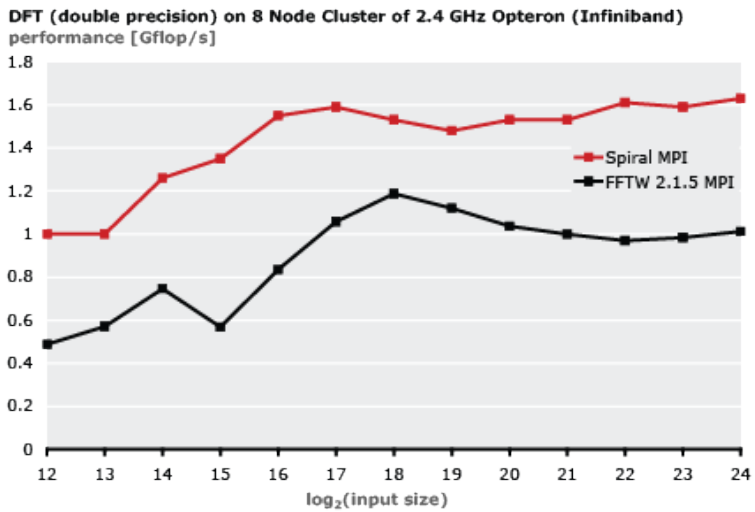


From Delta Debugging to Automated Program Repair



Program Synthesis

Fastest FFT is computer-generated

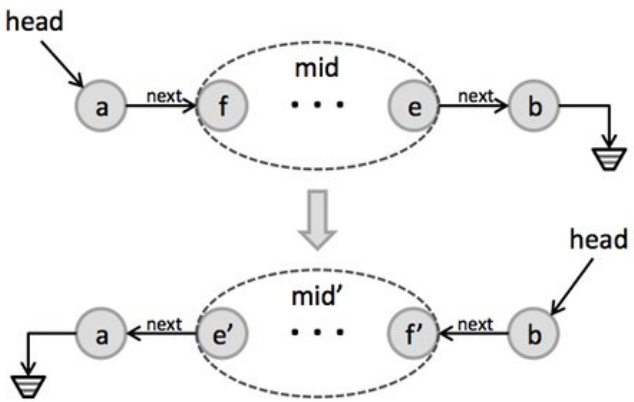


Sketching matrix transpose

```
int[16] trans_sse(int[16] M) implements trans {
  int[16] S = 0, T = 0;
  repeat (??) S[??::4] = shufps(M[??::4], M[??::4], ??);
  repeat (??) T[??::4] = shufps(S[??::4], S[??::4], ??);
  return T;
}
```

```
int[16] trans_sse(int[16] M) implements trans { // synthesized coc
  S[4::4]   = shufps(M[6::4],   M[2::4],  11001000b);
  S[0::4]   = shufps(M[11::4],  M[6::4],  10010110b);
  S[12::4]  = shufps(M[0::4],   M[2::4],  10001101b);
  S[8::4]   = shufps(M[8::4],   M[12::4], 11010111b);
  T[4::4]   = shufps(S[11::4],  S[1::4],  10111100b);
  T[12::4]  = shufps(S[3::4],   S[8::4],  11000011b);
  T[8::4]   = shufps(S[4::4],   S[9::4],  11100010b);
  T[0::4]   = shufps(S[12::4],  S[0::4],  10110100b);
}
```

Storyboard Programming




```
Node llReverse(Node head){
  Node temp1 = null, temp2 = null;
  Node temp3 = null;

  temp1 = head;
  while(temp1 != null){
    // unfold temp1;
    head = temp1;
    temp1 = temp1.next;
    head.next = head;
    head.next = temp3;
    temp3 = head;
    // fold head;
} }
```

Conclusion

- Software engineering is linear?

Conclusion

- ~~Software engineering is linear?~~
- **Software engineering is exponential**, just like everything else

References

- A. Solar-Lezama. Program Synthesis by Sketching. PhD thesis, University of California, Berkeley, 2008.
- A. Zeller and R. Hildebrandt. Simplifying and isolating failure-inducing input. IEEE Transactions on Software Engineering, 28(2):183200, Feb. 2002.
- D. Dig. Automated upgrading of component-based applications. PhD thesis, University of Illinois at Urbana-Champaign, 2007.
- M. Püschel, B. Singer, J. Xiong, J. M. F. Moura, J. Johnson, D. Padua, M. Veloso, and R. W. Johnson, SPIRAL: A generator for platform-adapted libraries of signal processing algorithms, Int. J. High Perform. Comput. Appl., vol. 18, no. 1, pp. 2145, 2004.
- R. Singh and A. Solar-Lezama. Synthesizing data structure manipulations from storyboards. In SIGSOFT FSE, pages 289-299, 2011.
- Westley Weimer, Stephanie Forrest, Claire Le Goues, and ThanhVu Nguyen. Automatic program repair with evolutionary computation. Communications of the ACM, 53(5):109116, June 2010.