

Patterns

- You find patterns by looking at what people do and observing things that work.
- Each pattern is a three-part rule, which expresses a relation between a specific context, a problem, and a solution.
- Each pattern describes a problem which occurs over and over again, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.

Pattern Languages

- Each pattern is incomplete, and needs the context of the others, to make sense.
- An isolated pattern does not possess a generative quality, this quality emerges from an entire system of patterns, interdependent at many levels.
- If these patterns really do belong together, then we should be able to come up with a story that flows through the patterns.

Software Craftsmanship

- Knuth's art
- The Pragmatic Programmer's subtitle
- Pete McBreen's imperative
- Graham's hackers
- A growing sentiment...
 - Guido van Rossum: If there was no art in it, it wouldn't be any fun, and then I wouldn't still be doing it after 30 years.
 - Richard Stallman: I would describe programming as a craft, which is a kind of art, but not a fine art. Craft means making useful objects with perhaps decorative touches. Fine art means making things purely for their beauty.

Apprenticeship Patterns

(beta version)

- **Accurate Self-Assessment**
 - **Be The Worst**
 - Kindred Spirits
 - Rubbing Elbows
 - Sweep The Floor
- **The Long Road**
 - **Draw Your Own Map**
 - **For Love Not Money**
 - **Ignore Your Title**
 - **Nurture Your Passion**
 - **Resist The Promotion**
- **Construct Your Curriculum**
 - Primary Sources
 - Read Constantly
 - Study The Classics
 - The Right Book At The Right Time
- **Perpetual Learning**
 - Breakable Toys
 - Confront Your Ignorance
 - Expand Your Bandwidth
 - Expose Your Ignorance
 - Learn How You Learn
 - **Practice, Practice, Practice**
 - Record What You Learn
 - Share What You Learn
 - Use The Source

Thank You!

The Latest Releases

<http://redsqirrel.com/dave/work/a2j>

I'm blogging this...

<http://redsqirrel.com/cgi-bin/dave>

Media I quoted, referenced, and paraphrased:

- *A Pattern Language*, Christopher Alexander et al
- "Art and Computer Programming" from ONLamp.com, John Littler
- *Organizational Patterns of Agile Software Development*, Jim Coplien
- *Patterns of Enterprise Application Architecture*, Martin Fowler
- *Software Craftsmanship*, Pete McBreen
- *The Pragmatic Programmer*, Andy Hunt & Dave Thomas
- *The Timeless Way of Building*, Christopher Alexander