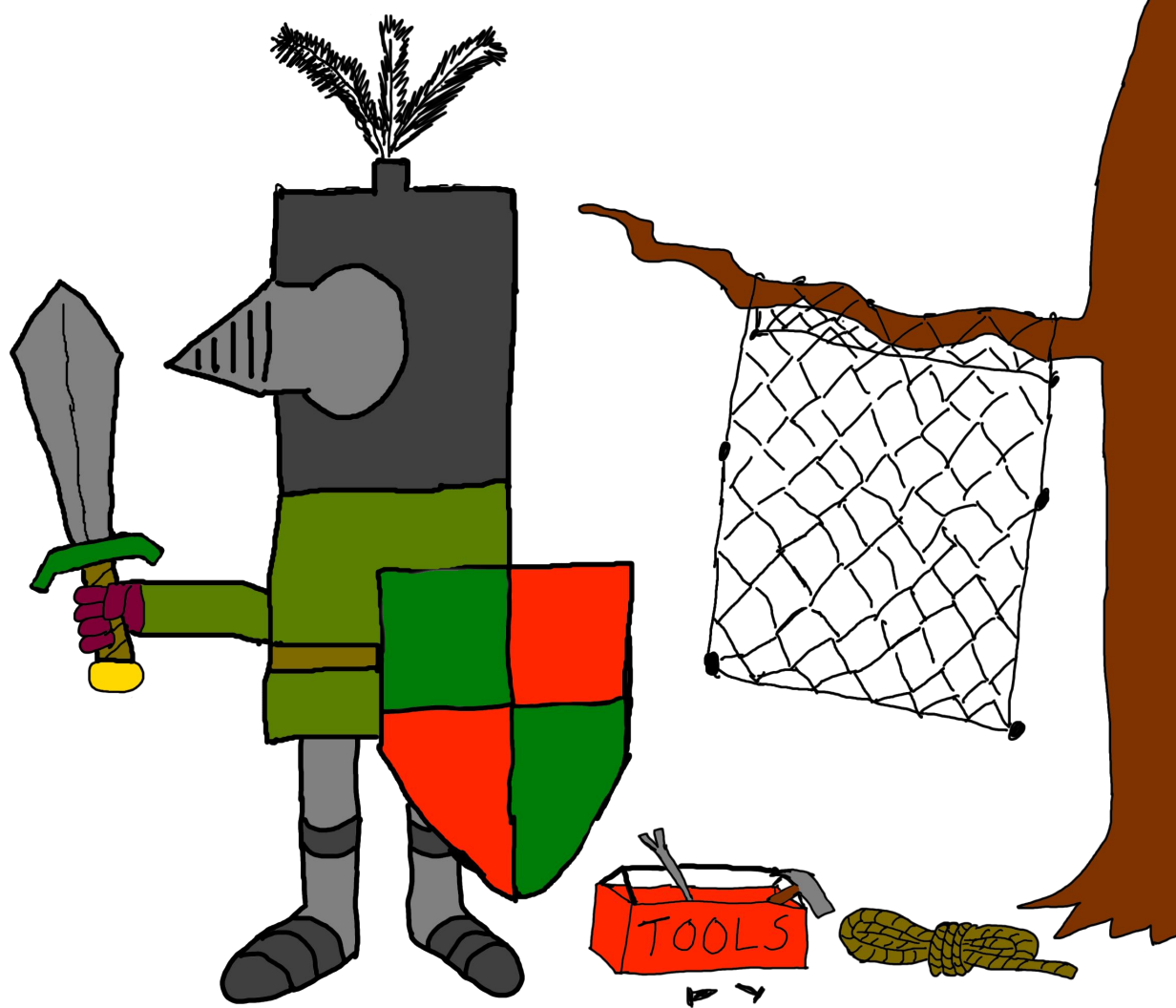# Hic Svnt Dracones!

Beat the beasts in the code

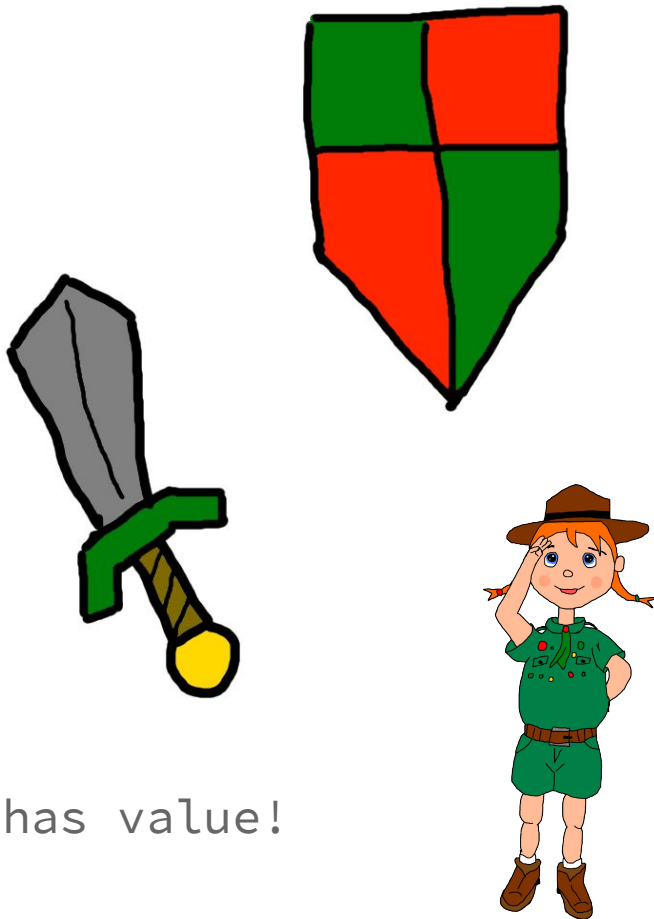# The Craftsman

Our Hero

METHODOLOGY

Chain the Beasts

- Shield
    - Capture the What not the How
    - Pinning and characterization tests
    - Create test outside in
- Sword
    - Break dependencies to other parts
    - Isolate really bad code
- Refactoring
    - Inside out
- Boy Scouting

Learn from the legacy code, it still has value!

Tools

- xUnit frameworks
- Mocking frameworks
- The Compiler
- Your favorite IDE
- Dependency Breaking techniques
- Mikado Method
- Ringmjuren a.k.a. Strangler Application Pattern

Smells of Trolls and Dragons

- Comments
- Talking to strangers, feature envy
- Duplication of knowledge
- Skylines
- Fortifications

Warning

- Don't refactor for the sake of refactoring
- Don't overdo the refactoring
- Don't take too large steps
- Don't refactor close to releases

Let's Charge!

References

- Working Effectively with Legacy Code – Michael Feathers
- Beyond Legacy Code – David Scott Bernstein
- Clean Code – Robert "Uncle Bob" Martin
- The Clean Coder – Robert "Uncle Bob" Martin
- Refactoring – Martin Fowler
- The Mikado Method – Ola Ellnestam, Daniel Brolund
- The Software Craftsman – Sandro Mancuso
- Test Driven Development – Kent Beck
- http://marketing.intracto.com/paying-technical-debt-how-to-rescue-legacy-code-through-refactoring – Jeroen Moons
- etc.