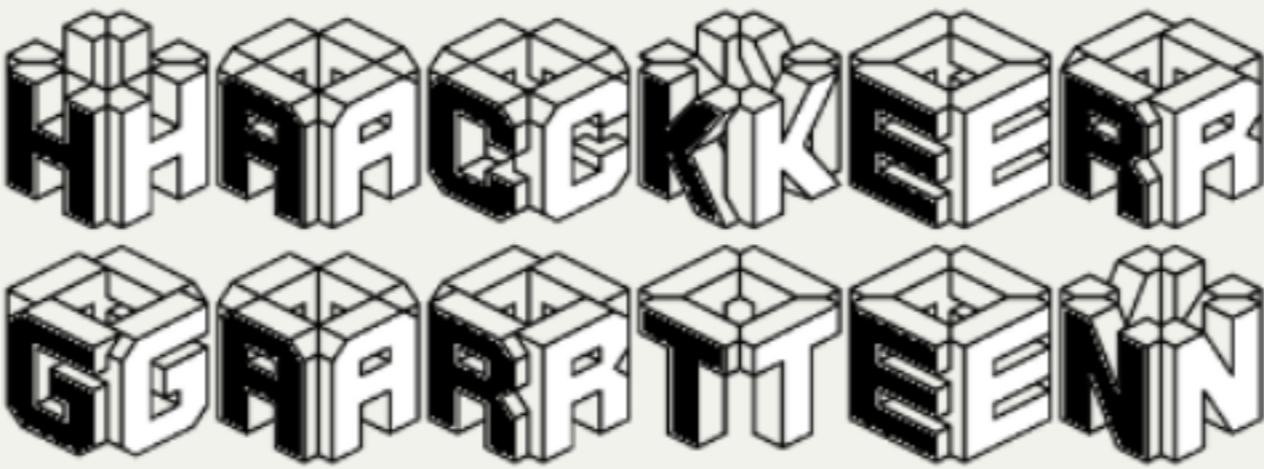


Documentation brought to life: Asciidoctor & Gradle

Markus Schlichting



Markus Schlichting

 **Software Engineer** 
[delivering end-user happiness] **Basel, Switzerland** **Hackergarten Basel** **markus.schlichting@canoo.com** **@madmas**

Overview

Overview

Current situation

Overview

Current situation

AsciiDoc(tor)

Overview

Current situation

AsciiDoc(tor)

Gradle

Overview

Current situation

AsciiDoc(tor)

Gradle

Combined powers (Demo!)

Overview

Current situation

AsciiDoc(tor)

Gradle

Combined powers (Demo!)

Living documentation (Demo!)

Overview

Current situation

AsciiDoc(tor)

Gradle

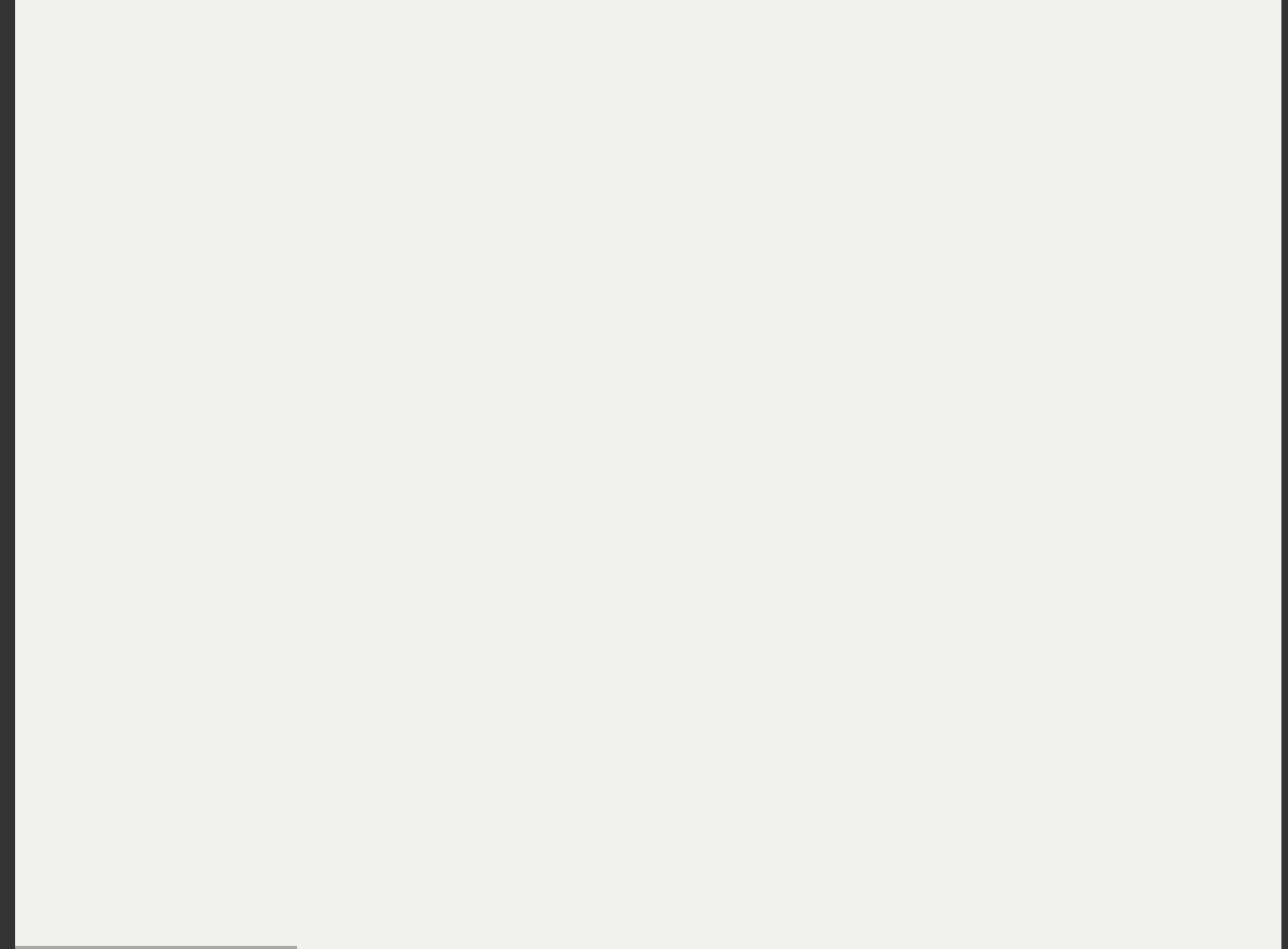
Combined powers (Demo!)

Living documentation (Demo!)

Additional boosts

We write documentation
with systems that drive us
crazy





AsciiDoc

AsciiDoc

```
= AsciiDoc is Writing Zen
Doc Writer <doc.writer@example.com>
:icons: font

_Zen_ in the *art* of writing `plain text` with
http://asciidoc.org[AsciiDoc].  
  
[TIP]
Use http://asciidoctor.org[Asciidoctor] for the best AsciiDoc
experience.footnote:[Not to mention the best looking output!]
Then icon:twitter[role=aqua] about it!  
  
== Sample Section  
  
[square]
* item 1
* item 2  
  
[source,ruby]
-----
puts "Hello, World!"  
-----
```

AsciiDoc source

AsciiDoc is Writing Zen

Doc Writer – doc.writer@example.com

Zen in the **art** of writing **plain text** with [AsciiDoc](#).



Use [Asciidoctor](#) for the best AsciiDoc experience.^[1] Then about it!

Sample Section

- item 1
- item 2

```
puts "Hello, World!"
```

1. Not to mention the best looking output!

Rendered HTML

AsciiDoc

```
= AsciiDoc is Writing Zen
Doc Writer <doc.writer@example.com>
:icons: font

_Zen_ in the *art* of writing `plain text` with
http://asciidoc.org[AsciiDoc].  
  
[TIP]
Use http://asciidoctor.org[Asciidoctor] for the best AsciiDoc
experience.footnote:[Not to mention the best looking output!]
Then icon:twitter[role=aqua] about it!  
  
== Sample Section  
  
[square]
* item 1
* item 2  
  
[source,ruby]
-----
puts "Hello, World!"  
-----
```

AsciiDoc source

AsciiDoc is Writing Zen

Doc Writer – doc.writer@example.com

Zen in the art of writing plain text with [AsciiDoc](#).



Use [Asciidoctor](#) for the best AsciiDoc experience.^[1] Then about it!

Sample Section

- item 1
- item 2

```
puts "Hello, World!"
```

1. Not to mention the best looking output!

Rendered HTML

“

Use AsciiDoc for document markup. It's actually readable by humans, easier to parse and more flexible than XML.

— L. Torvalds

AsciiDoctor

AsciiDoctor

modern impl. of AsciiDoc with Ruby

AsciiDoctor

modern impl. of AsciiDoc with Ruby

easily useable on JVM using JRuby

AsciiDoctor

modern impl. of AsciiDoc with Ruby

easily useable on JVM using JRuby

supports extensions

AsciiDoctor

modern impl. of AsciiDoc with Ruby

easily useable on JVM using JRuby

supports extensions

many output formats

PDF, eBook

HTML5

everything else (→ DocBook)

AsciiDoctor

modern impl. of AsciiDoc with Ruby

easily useable on JVM using JRuby

supports extensions

many output formats

PDF, eBook

HTML5

everything else (→ DocBook)

maintainable Tables with Ascii or CSV

AsciiDoctor

modern impl. of AsciiDoc with Ruby

easily useable on JVM using JRuby

supports extensions

many output formats

PDF, eBook

HTML5

everything else (→ DocBook)

maintainable Tables with Ascii or CSV

Admoticons

Where

Documentation

Writing & publishing books

Presentations (deck.js, reveal.js, DZSlides)

the upcoming build system
for the JVM ecosystem (&
beyond)



GRADLE: SAMPLE

```
apply plugin: 'java'
apply plugin: 'application'

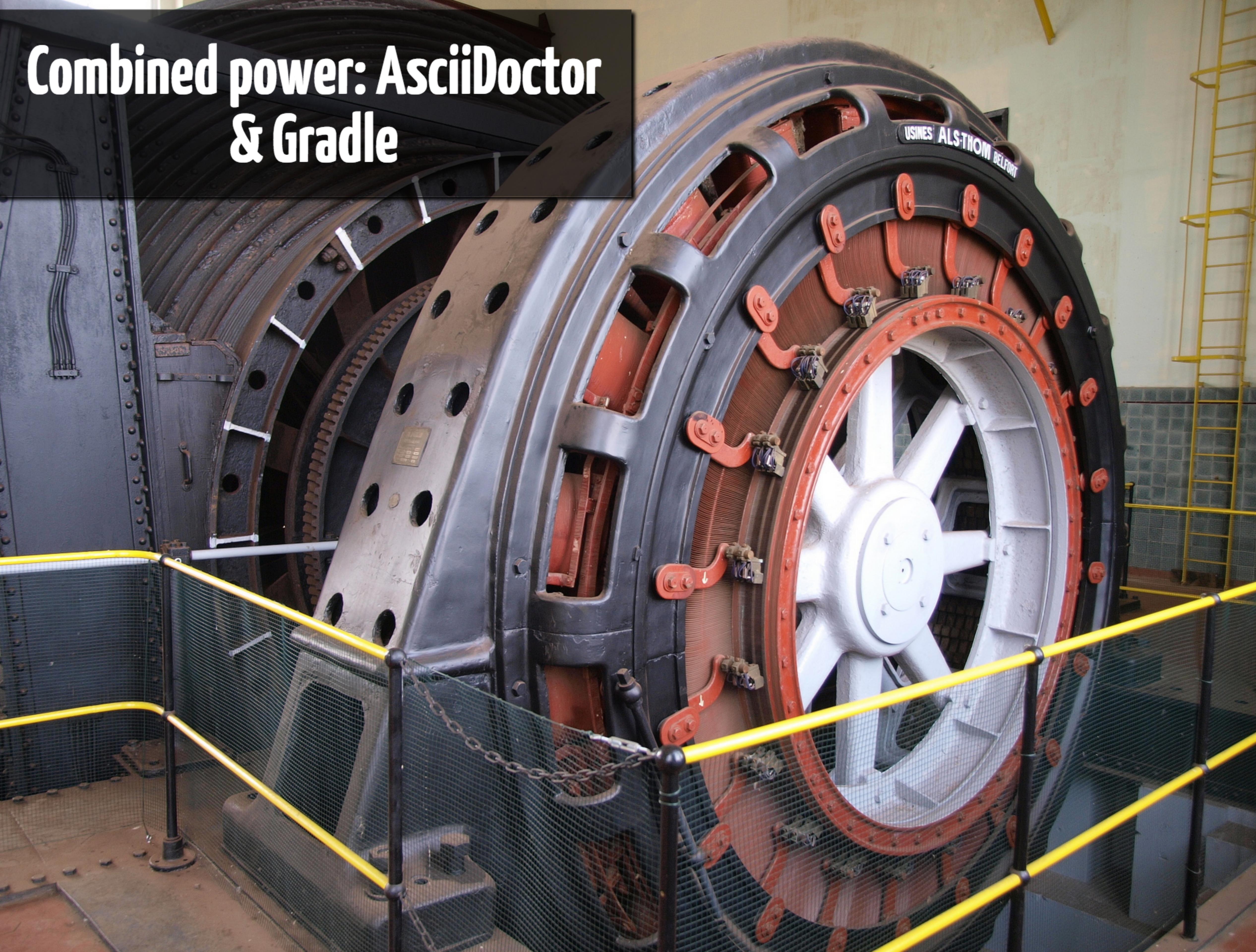
mainClassName = 'sample.Foo'

version = '0.0.0-SNAPSHOT'
group = 'com.canoo'

repositories {
    jcenter()
}

dependencies {
    compile 'commons-lang:commons-lang:2.6'
    testCompile 'junit:junit:4.11'
}
```

Combined power: AsciiDoctor & Gradle



Workflow: daily business

Workflow: daily business

Gradle watch task

Workflow: daily business

Gradle watch task

Version Number integration

Workflow: daily business

Gradle watch task

Version Number integration

Tested snippets in documentation

Workflow: git adds superpowers!

Workflow: git adds superpowers!

Split and structure your documentation → apply DRY

Workflow: git adds superpowers!

Split and structure your documentation → apply DRY

pull / merge requests! → easier envolvement with progress in development

Workflow: git adds superpowers!

Split and structure your documentation → apply DRY

pull / merge requests! → easier envolvement with progress in development

plays well with feature branches

Workflow: git adds superpowers!

Split and structure your documentation → apply DRY

pull / merge requests! → easier envolvement with progress in development

plays well with feature branches

Integrate with releases and generate the correct documentation

Creation

Preview while edit:

IntelliJ IDEA Plugin

Eclipse: jbosstools-asciidocor

Firefox plugin

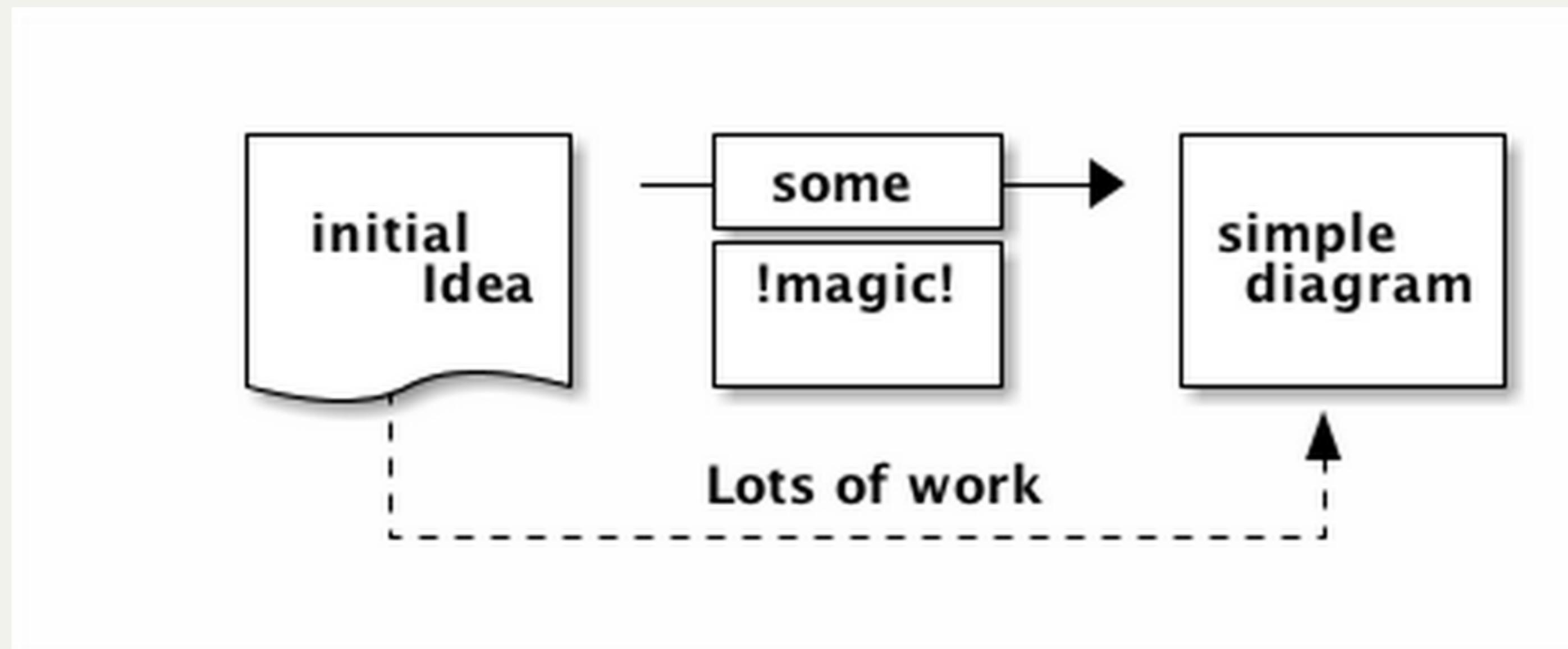
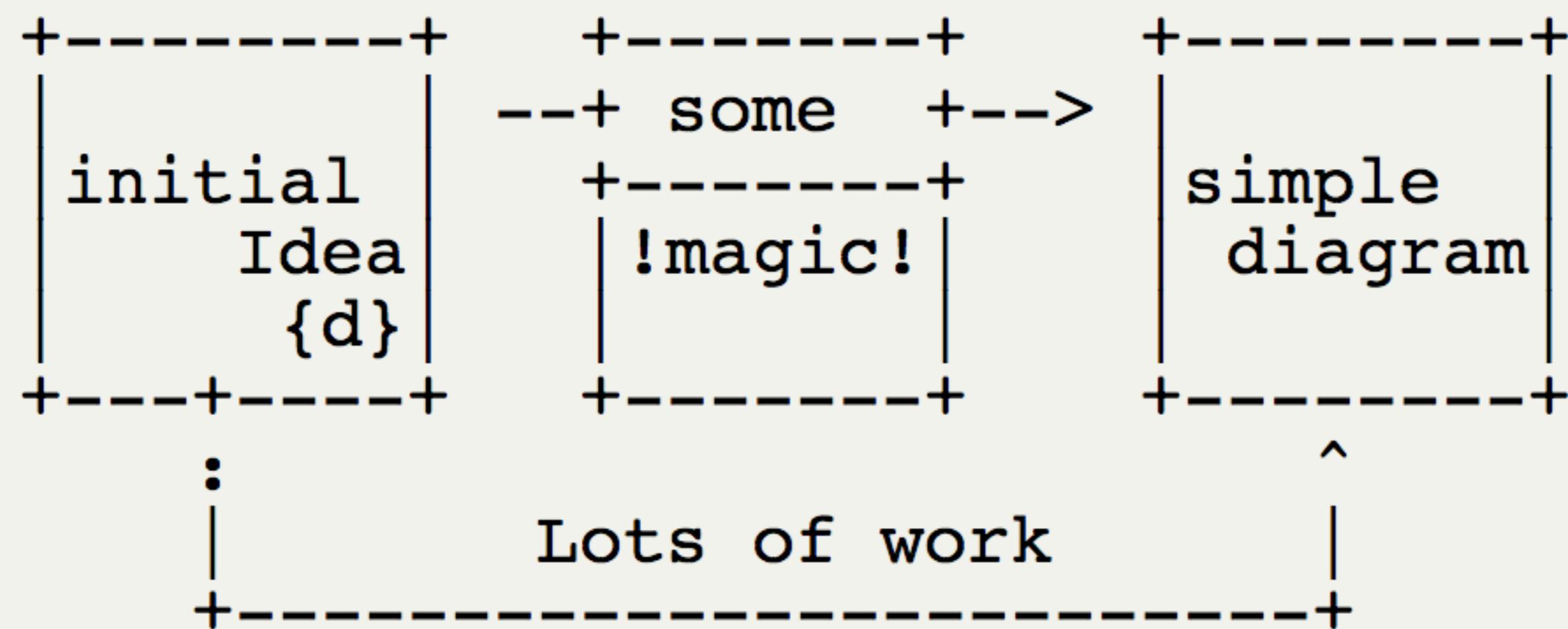
Chrome plugin

Specification by Example & Living Documentation

DEMO TIME!

Additional power: Diagrams

- PlantUML
- Dita
- Shaape
- BlockDiag, SeqDiag, ActDiag, NwDiag
- GraphViz DOT



Get started

Asciidoctor Gradle Lazybones Template:

```
lazybones create asciidoctor-gradle <yourProjectName>
```

Thank you!

Questions ?

 **Markus Schlichting** 
[delivering end-user happiness]

 markus.schlichting@canoo.com

 [@madmas](#)

Resources

[asciidoc-gradle-examples](#)

[Discover the zen of writing AsciiDoc](#)

[AsciiDoc with pleasure](#)

[When WebSockets meet AsciiDoctor](#)

[Asciidoctor Tips from Devoxx](#)

[MrHaki's Blog with regular posts on AsciiDoctor](#)

[Peter Ledbrook: 'Practical Groovy'](#)

fin!
