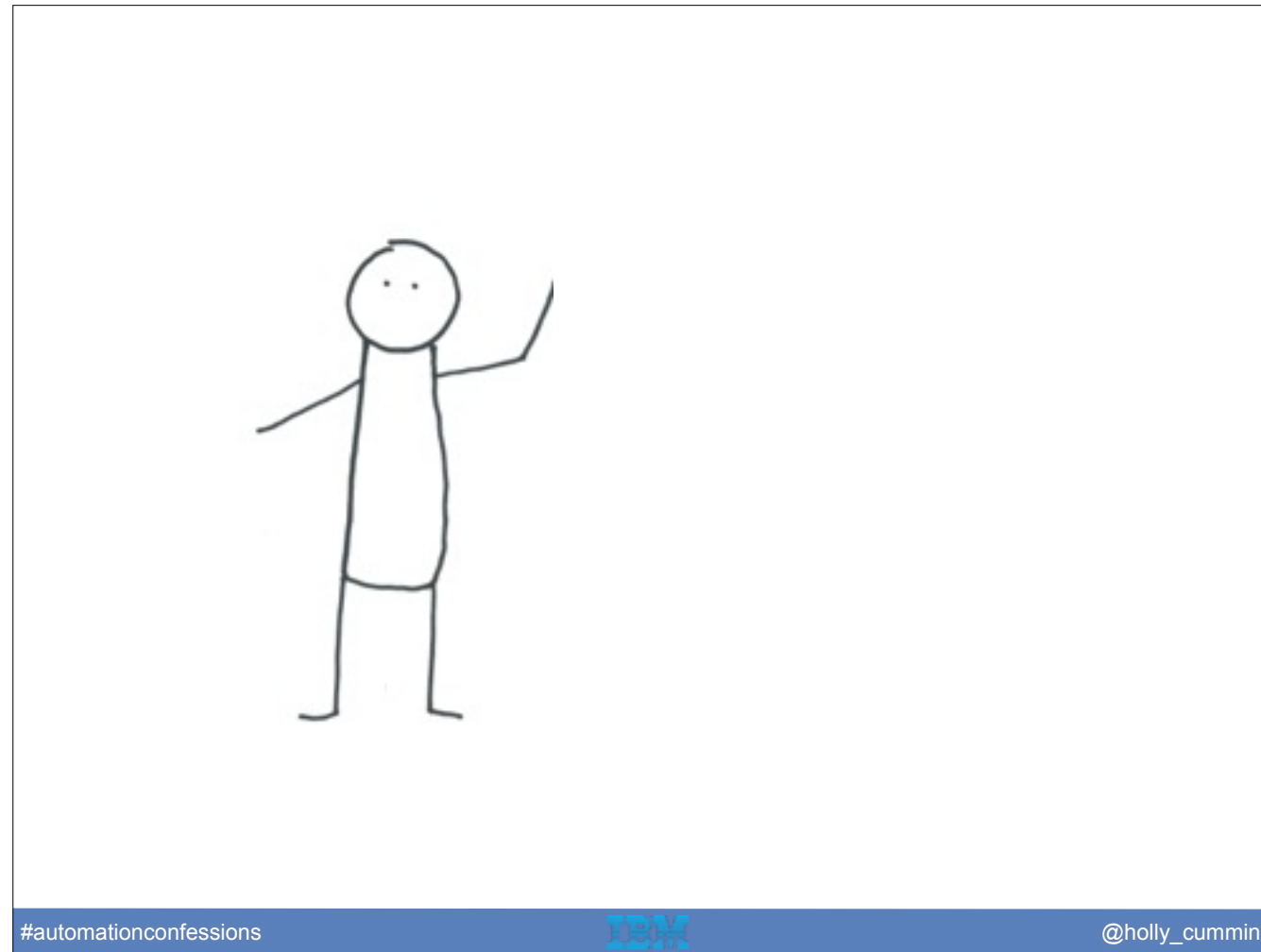


Confessions of an automation addict

Holly Cummins
@holly_cummins



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

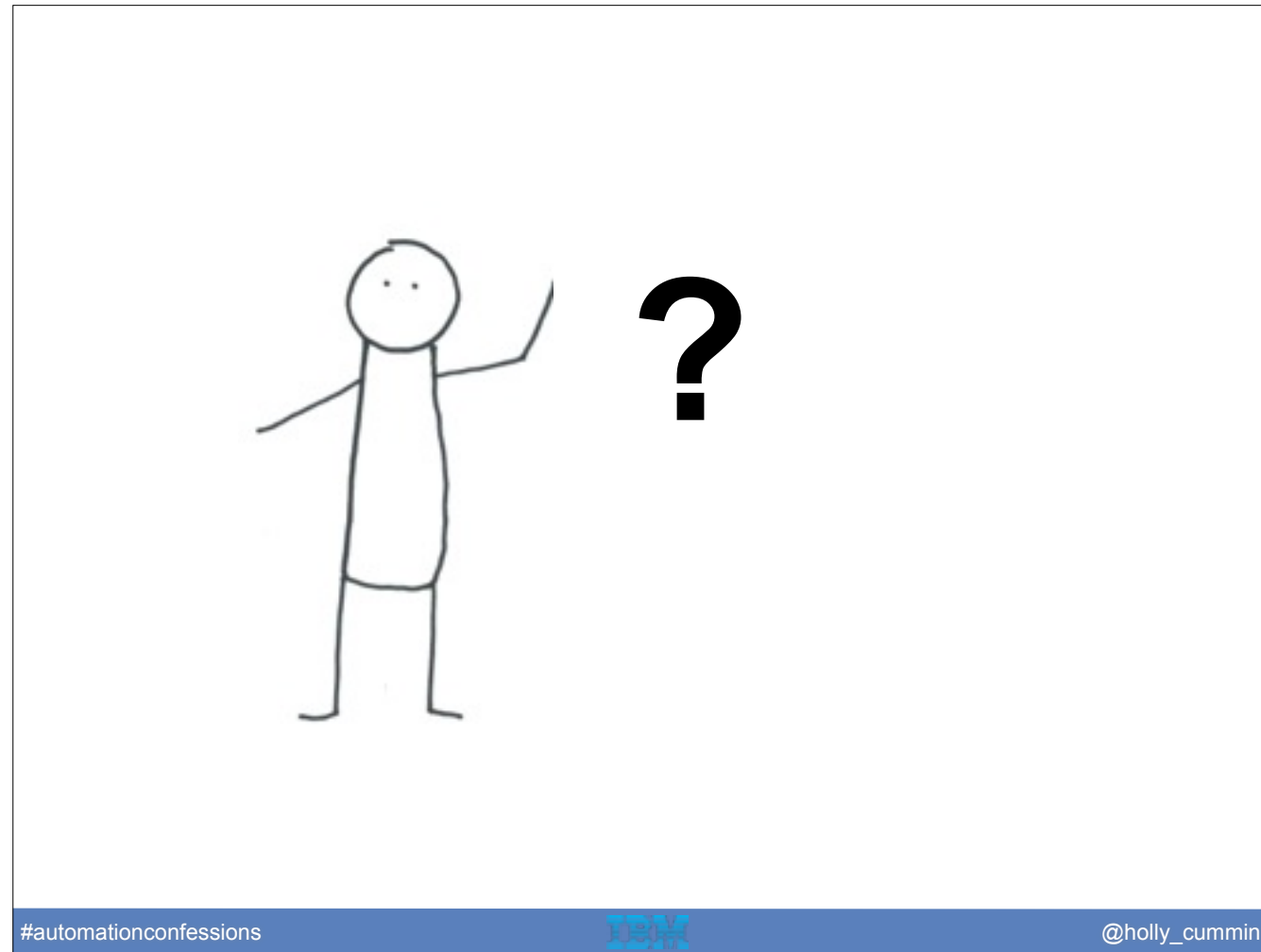
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

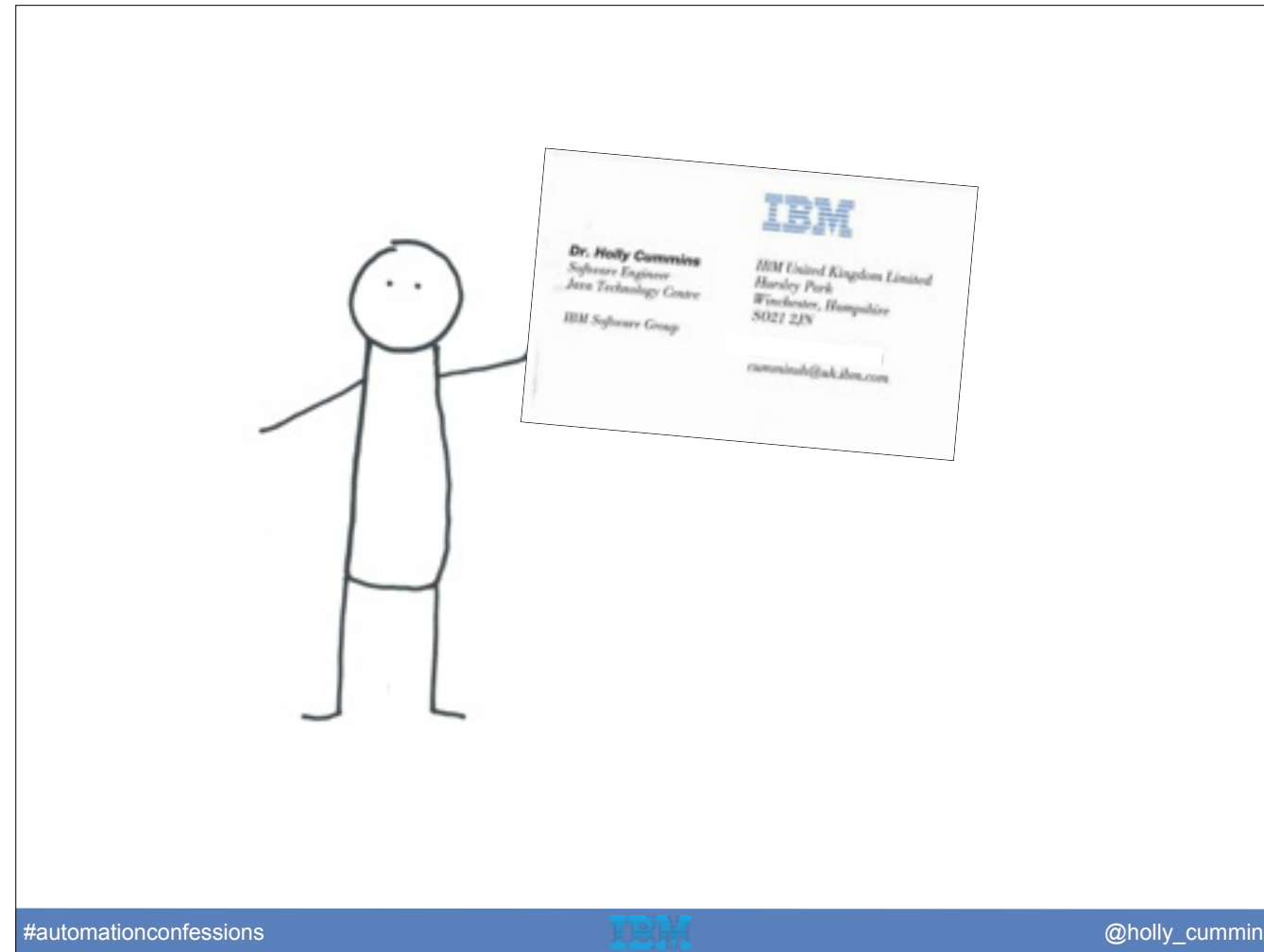
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

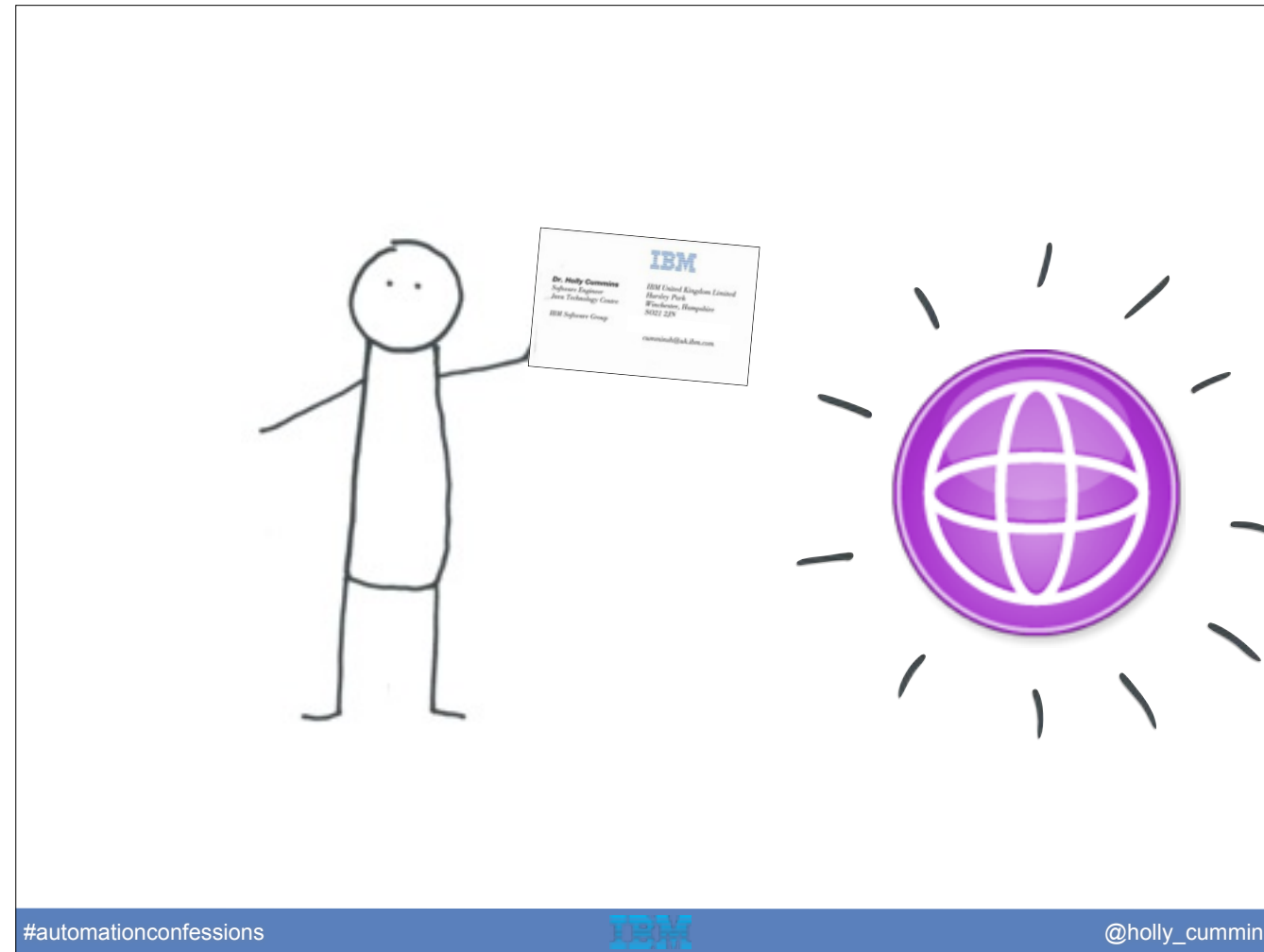
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

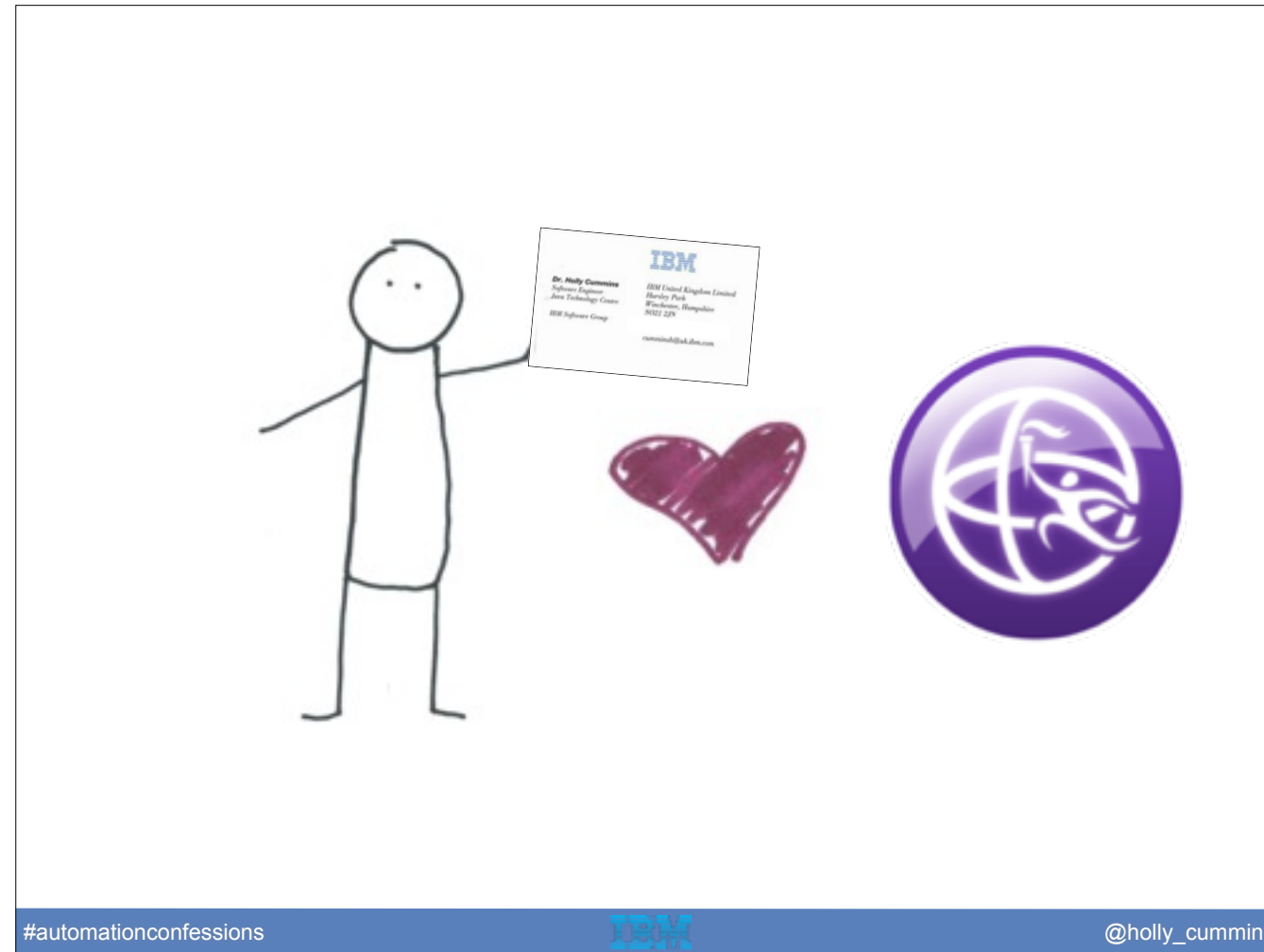
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

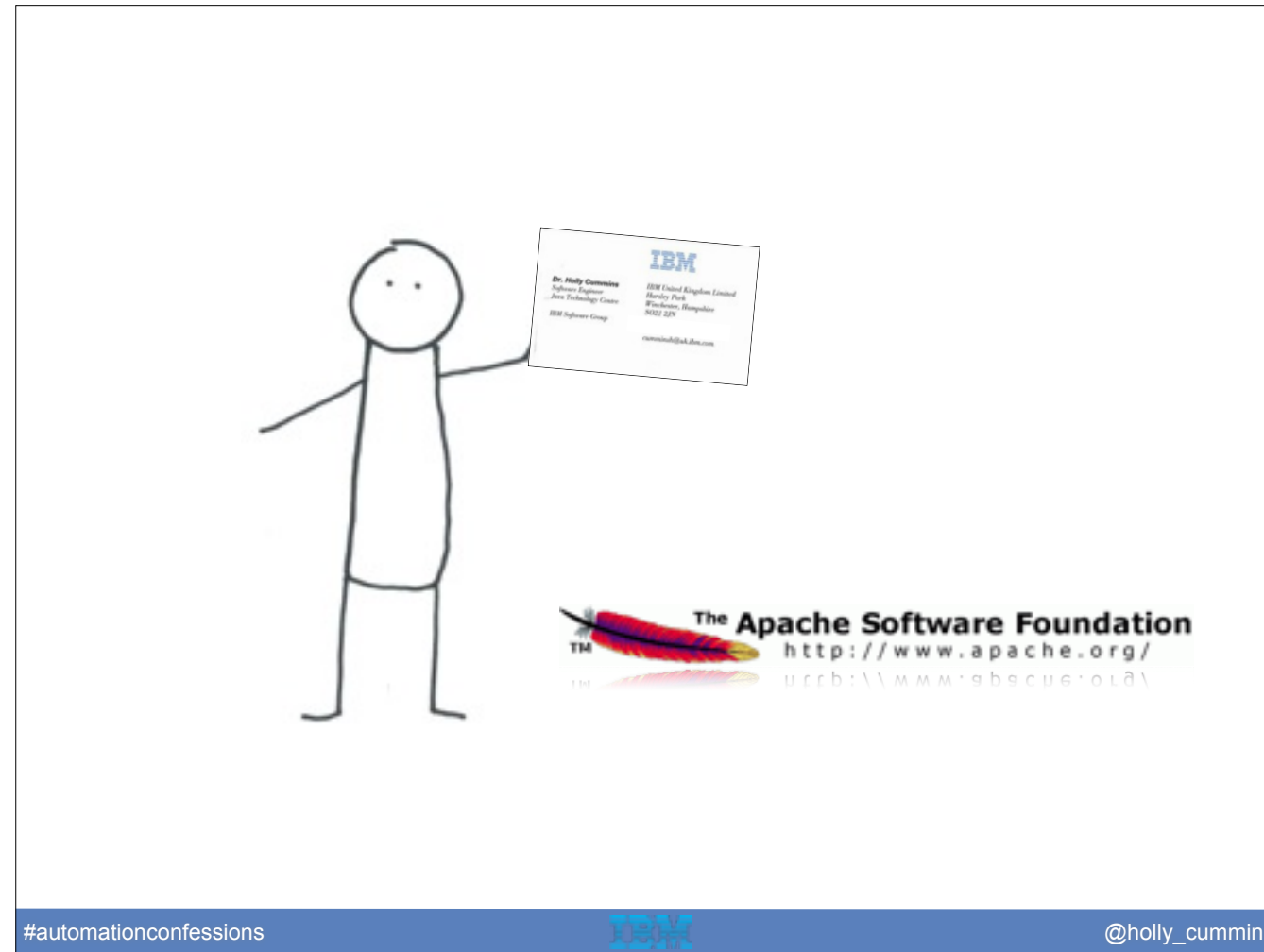
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

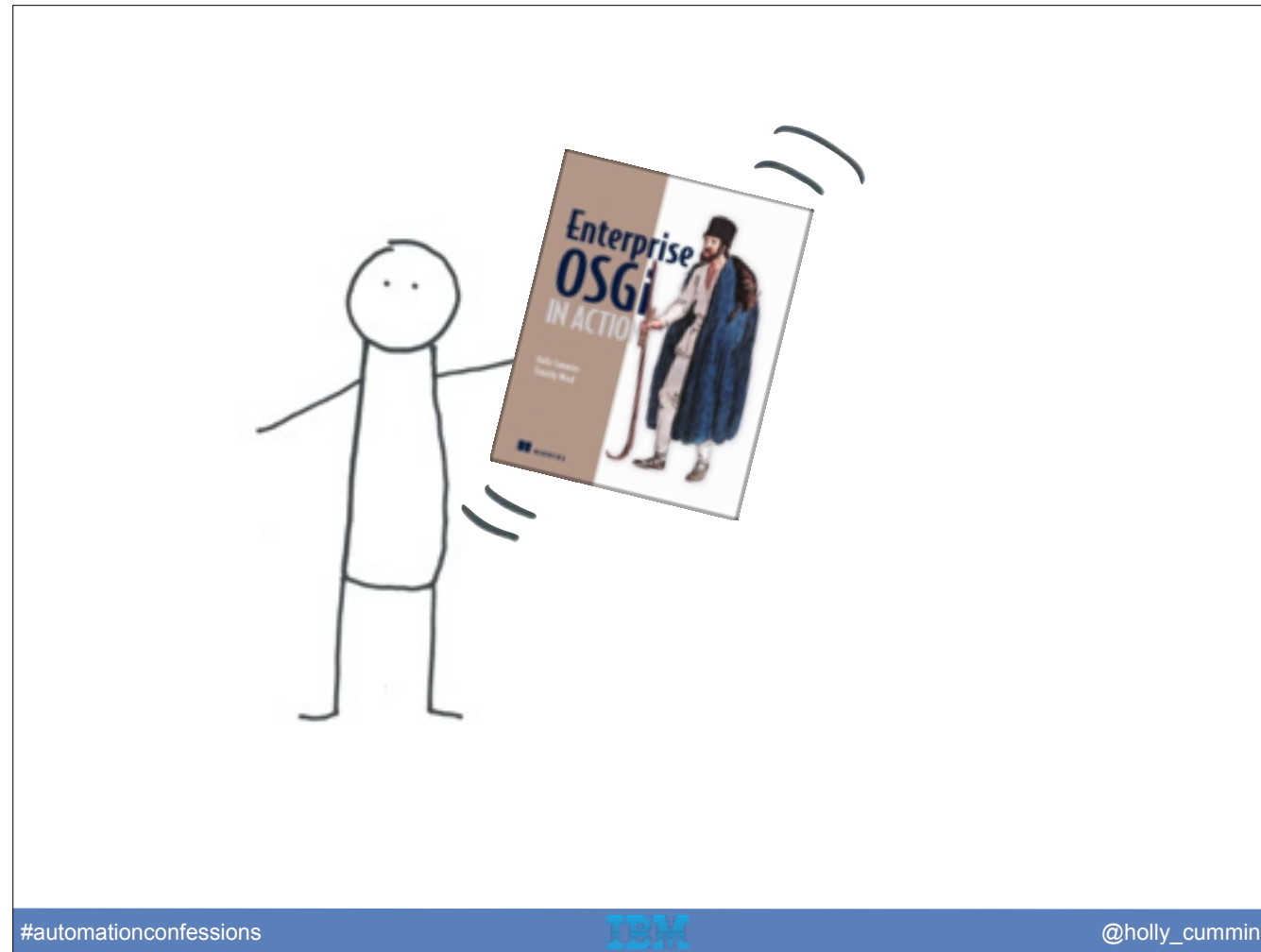
I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



Let me start by telling you a bit about myself, and the context in which I've been trying to automate.

I work for IBM, as a developer in the Hursley labs.

A great thing about working for a company like IBM is you can move around and work on lots of different things. Right now I'm working on WebSphere. At the moment I'm working on the lightweight version of WebSphere Liberty Profile. We've got this unofficial 'i heart liberty' logo that we've done, because we really are incredibly proud of that server. It's so lightweight, so easy to configure, it starts in a flash, it's got a really small download size and footprint ..

Bluemix

I'm also an Apache Committer, on the Apache Aries project, which is an implementation of enterprise OSGi. It takes some of the JEE specs, like JPA, and makes them work in an OSGi world.

If you'd like to know about enterprise OSGi, I have written a book about it. :)



I love automation. I really love automation.

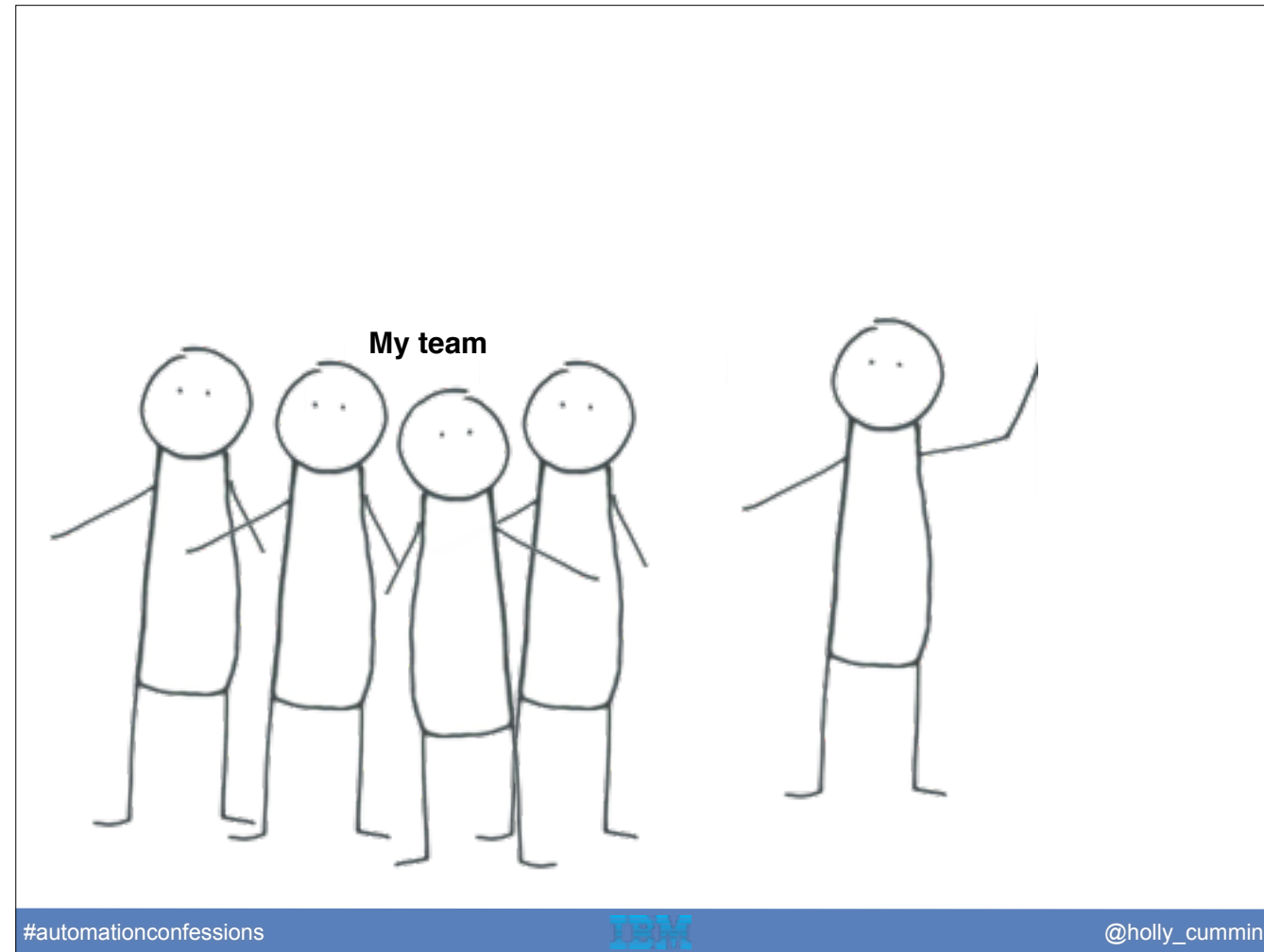
Examples -

1993, dialup, to university network, had a command replay thing on my mac, handled logging in to university dialup, if it was a bit slow to pick up would just display my password. I was a bit surprised first time a friend told me my password.

started at ibm, CMVC gui, I wrote a whole library of command line scripts

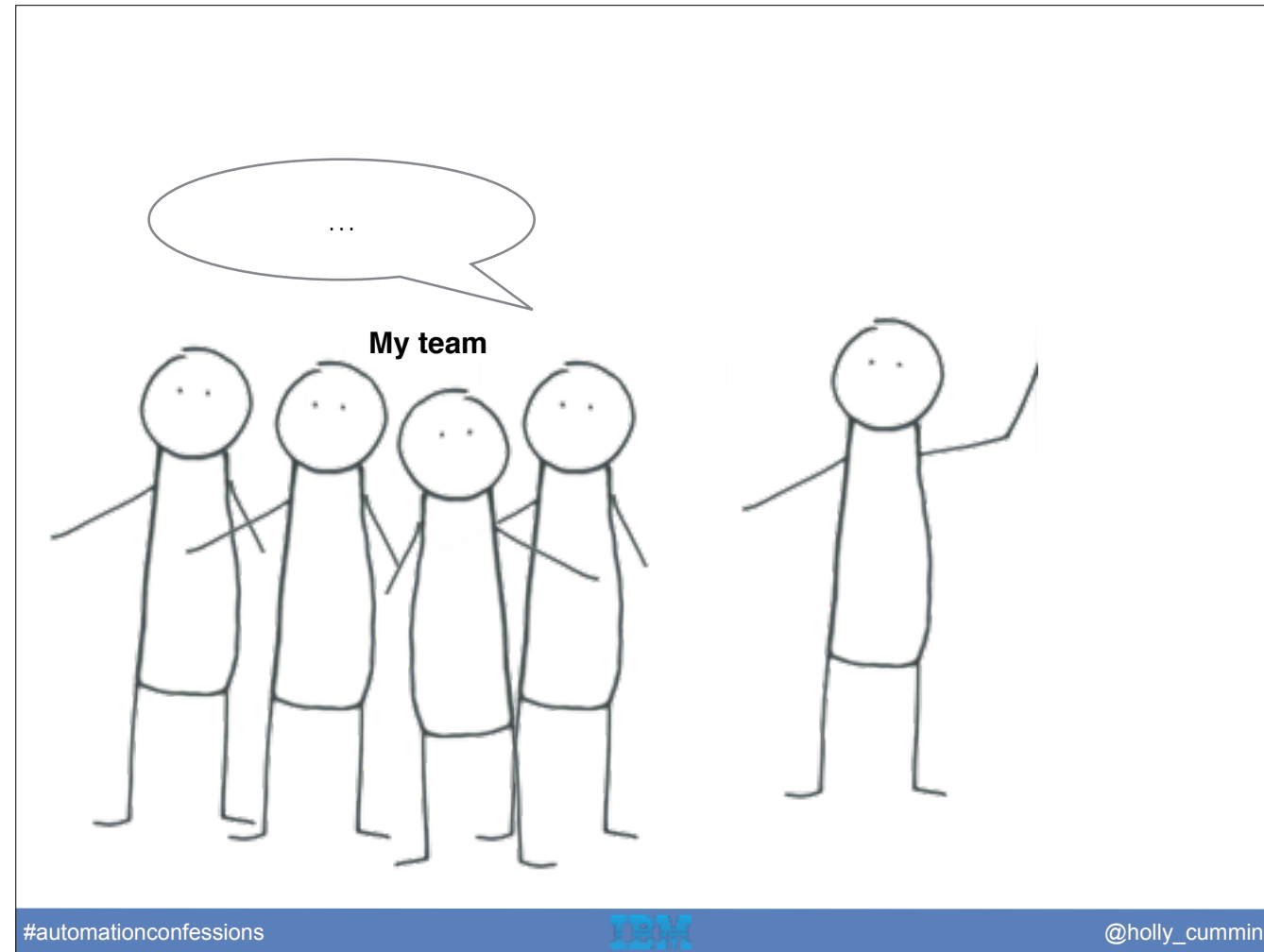
populating data for tests, wrote thing to inject them with data by reflection - this was before mocks - and kept breaking colleagues by leaving stuff which tested their edge cases in a shared db. I thought they should be grateful I was discovering problems in their code for them, but ...

And then there was the time I needed to write formal design docs, and people wanted to consume them in Word, and I wanted to have my objects automatically populated from Java, so I wrote a tool to convert Java to nicely formatted class diagrams in SVG and LaTeX, and then to word. There are products that do less than what I'd written ...



We all agree automation is good, but some do it more automatically than others.

I've never seen a thing I didn't want to automate. I'm always the one in our scrums who says "can that be automated?" whenever anyone else explains what they're doing.



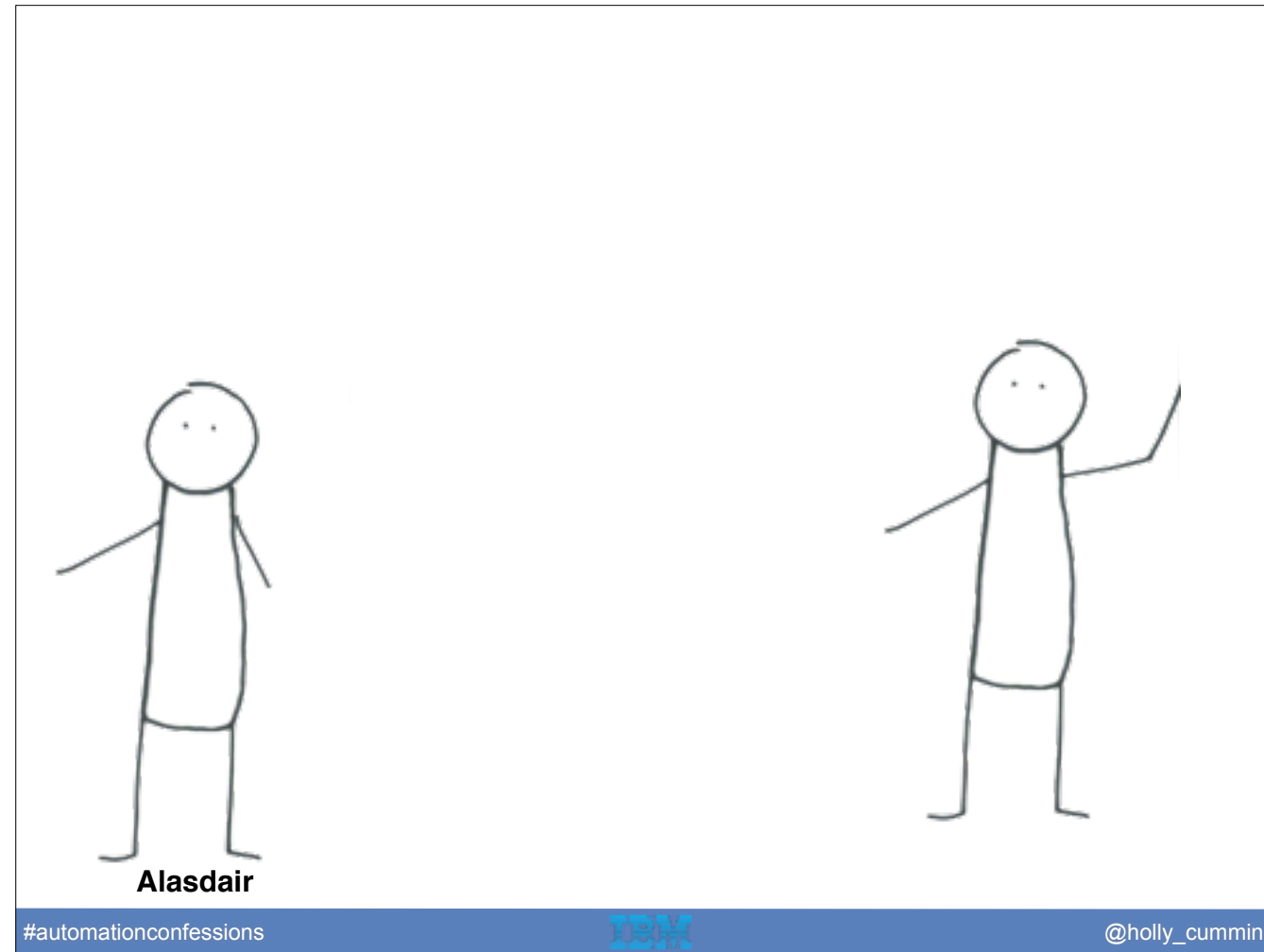
We all agree automation is good, but some do it more automatically than others.

I've never seen a thing I didn't want to automate. I'm always the one in our scrums who says "can that be automated?" whenever anyone else explains what they're doing.



We all agree automation is good, but some do it more automatically than others.

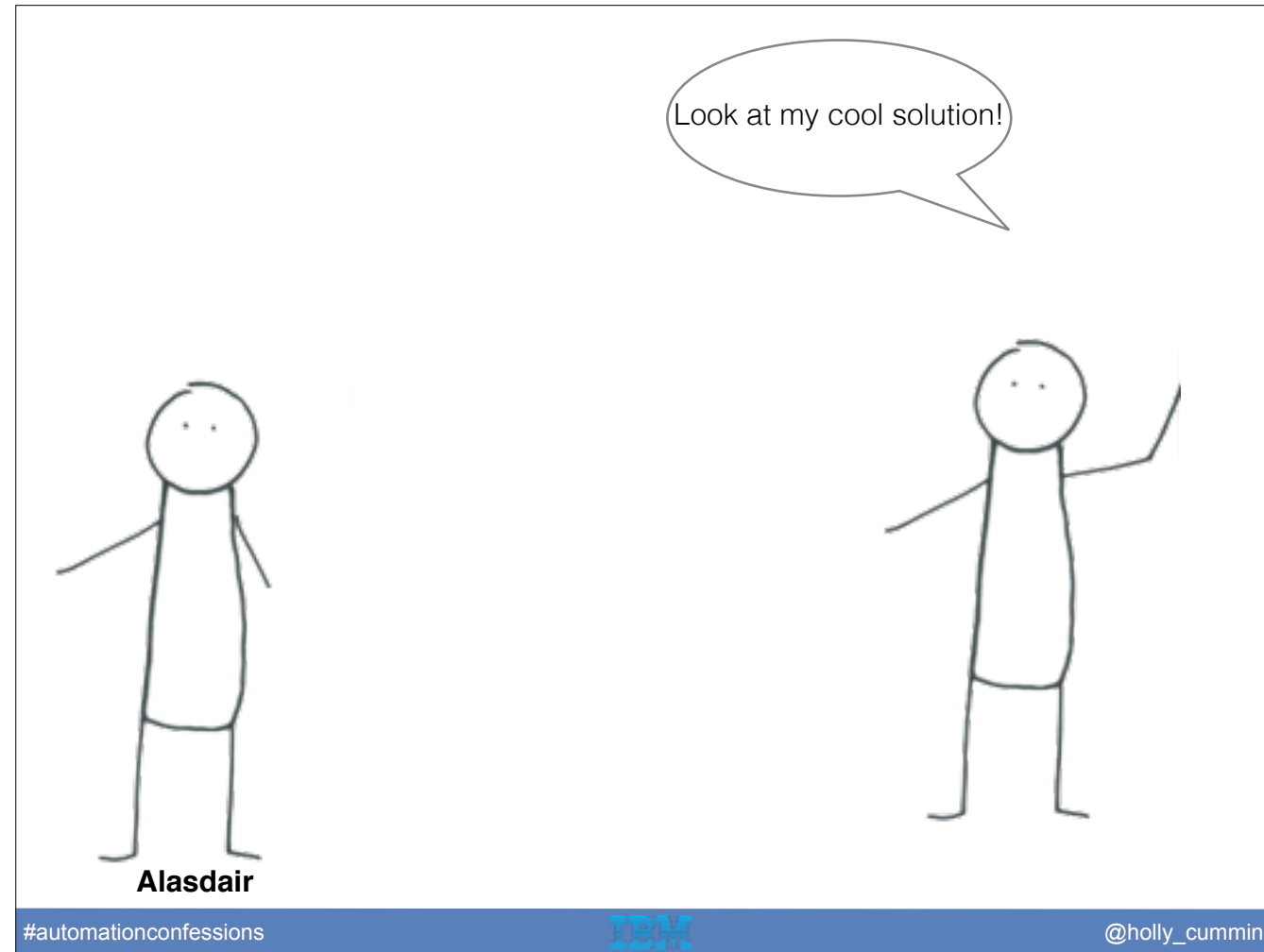
I've never seen a thing I didn't want to automate. I'm always the one in our scrums who says "can that be automated?" whenever anyone else explains what they're doing.



Loving automation has had some advantages, professionally, and also some disadvantages.

Aries reconsume - our customers expect iron-clad serviceability, so that means being able to rebuild several years down the line. We can't guarantee an external server will still be there so we need copies of source we ship. download tagged level of source tree, zip up, upload to filesystem, put link in to our builds, munge build labels, then build.

So I wrote a tool to mirror from subversion to our source control system, build, and upload. So to do a reconsume just snapshot source control and update one reference. In stabilisation mode we could just fork a stream and turn off mirroring.



Loving automation has had some advantages, professionally, and also some disadvantages.

Aries reconsume - our customers expect iron-clad serviceability, so that means being able to rebuild several years down the line. We can't guarantee an external server will still be there so we need copies of source we ship. download tagged level of source tree, zip up, upload to filesystem, put link in to our builds, munge build labels, then build.

So I wrote a tool to mirror from subversion to our source control system, build, and upload. So to do a reconsume just snapshot source control and update one reference. In stabilisation mode we could just fork a stream and turn off mirroring.



Loving automation has had some advantages, professionally, and also some disadvantages.

Aries reconsume - our customers expect iron-clad serviceability, so that means being able to rebuild several years down the line. We can't guarantee an external server will still be there so we need copies of source we ship. download tagged level of source tree, zip up, upload to filesystem, put link in to our builds, munge build labels, then build.

So I wrote a tool to mirror from subversion to our source control system, build, and upload. So to do a reconsume just snapshot source control and update one reference. In stabilisation mode we could just fork a stream and turn off mirroring.



Loving automation has had some advantages, professionally, and also some disadvantages.

Aries reconsume - our customers expect iron-clad serviceability, so that means being able to rebuild several years down the line. We can't guarantee an external server will still be there so we need copies of source we ship. download tagged level of source tree, zip up, upload to filesystem, put link in to our builds, munge build labels, then build.

So I wrote a tool to mirror from subversion to our source control system, build, and upload. So to do a reconsume just snapshot source control and update one reference. In stabilisation mode we could just fork a stream and turn off mirroring.



Loving automation has had some advantages, professionally, and also some disadvantages.

Aries reconsume - our customers expect iron-clad serviceability, so that means being able to rebuild several years down the line. We can't guarantee an external server will still be there so we need copies of source we ship. download tagged level of source tree, zip up, upload to filesystem, put link in to our builds, munge build labels, then build.

So I wrote a tool to mirror from subversion to our source control system, build, and upload. So to do a reconsume just snapshot source control and update one reference. In stabilisation mode we could just fork a stream and turn off mirroring.

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

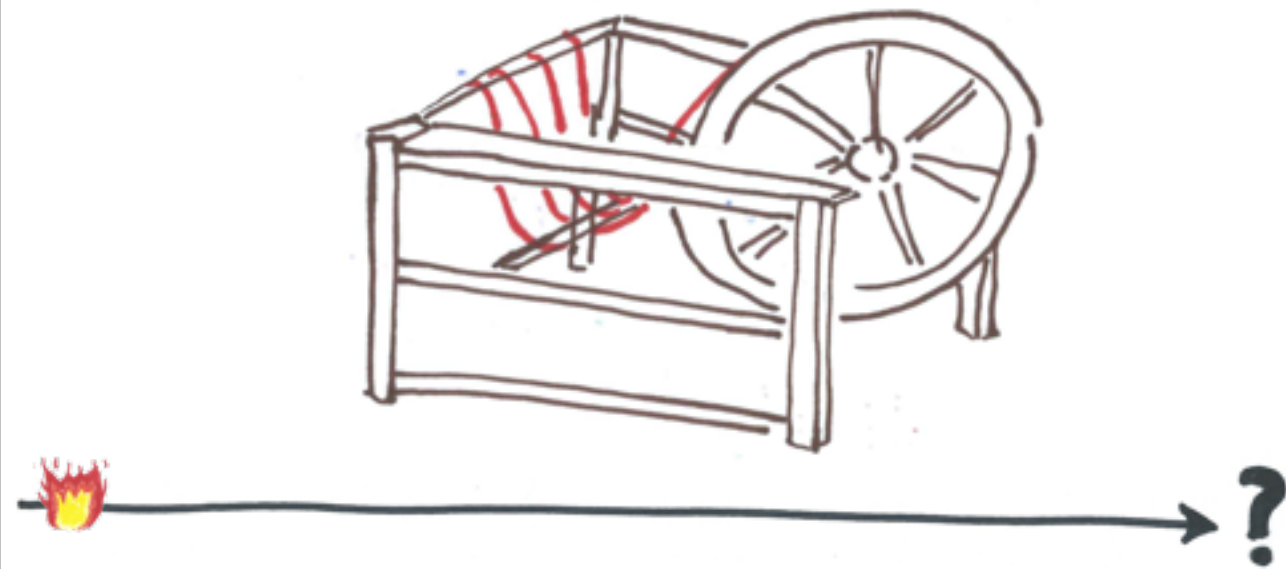
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

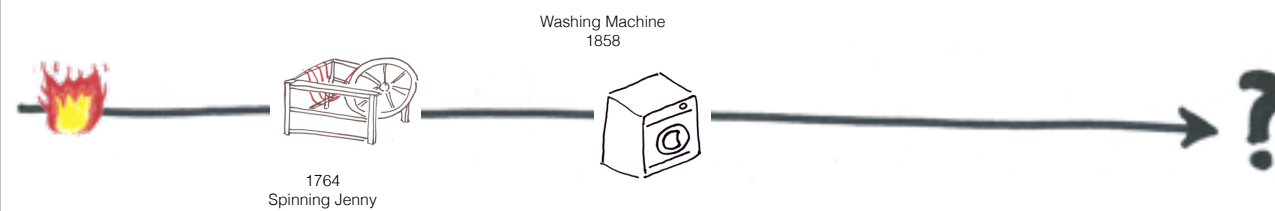
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

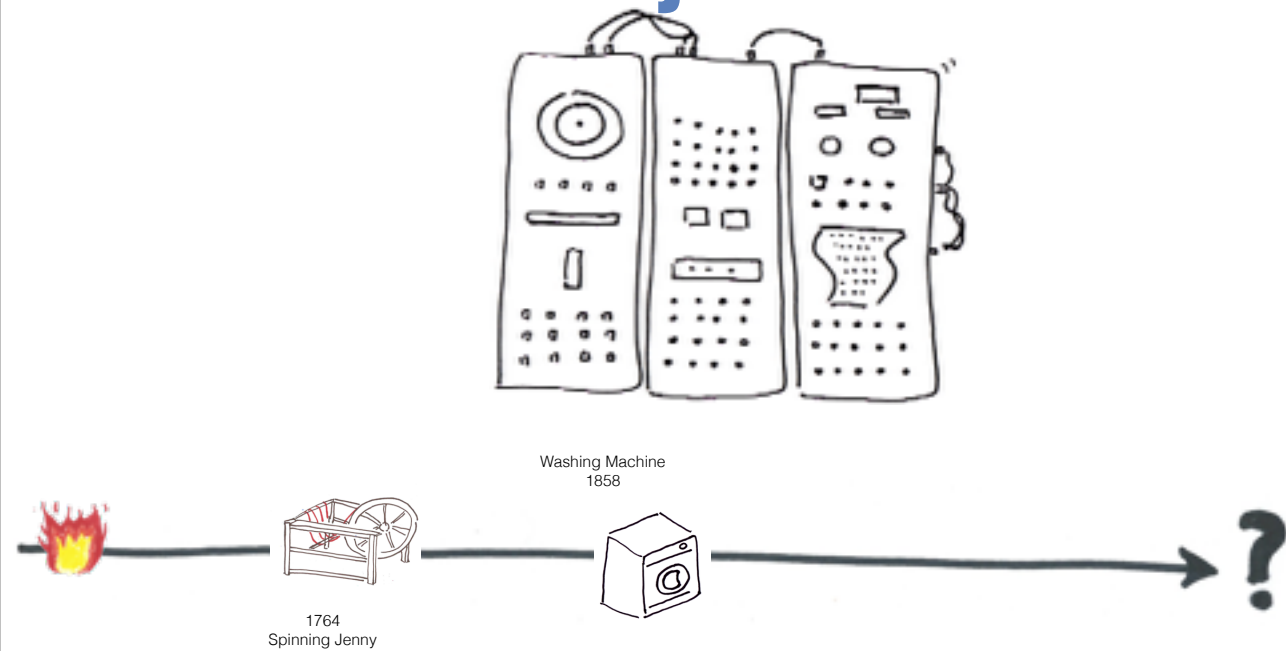
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

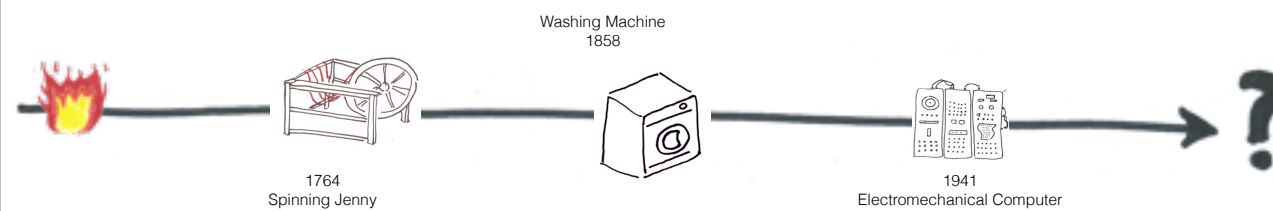
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

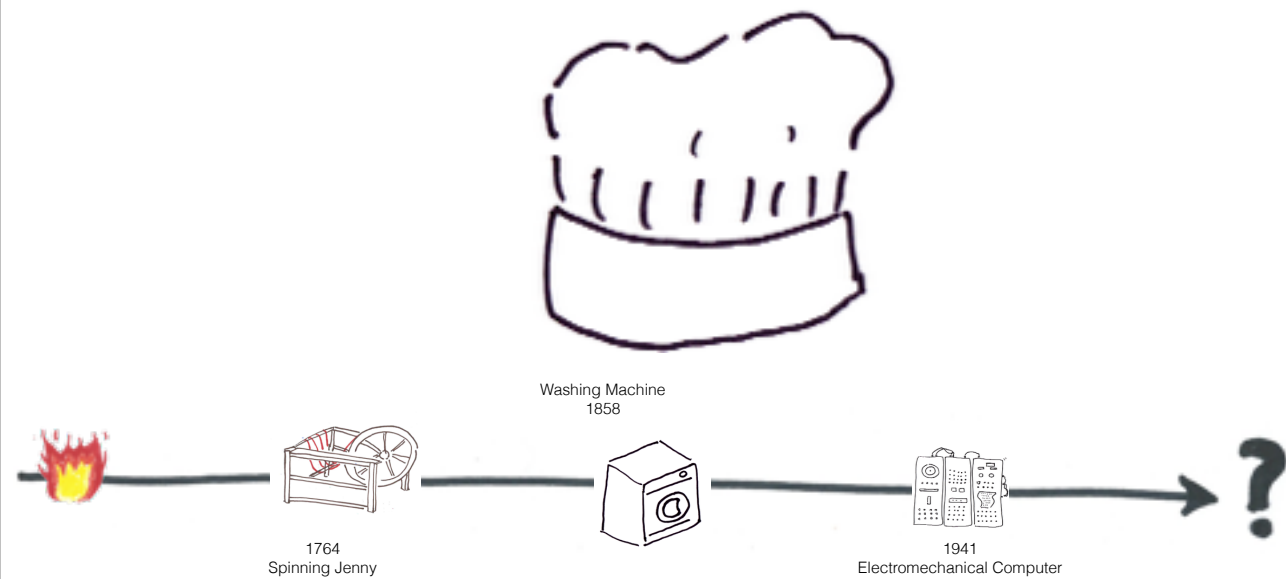
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

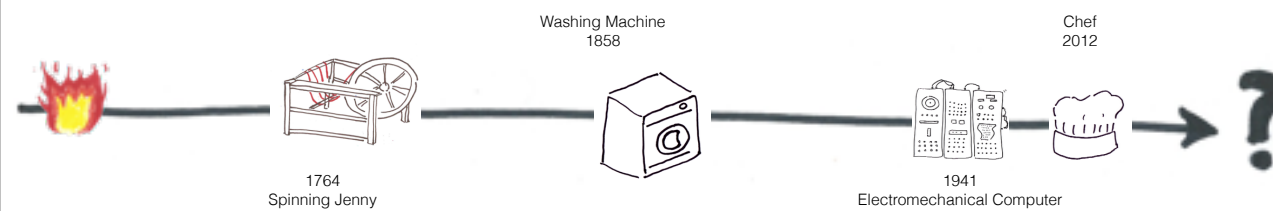
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

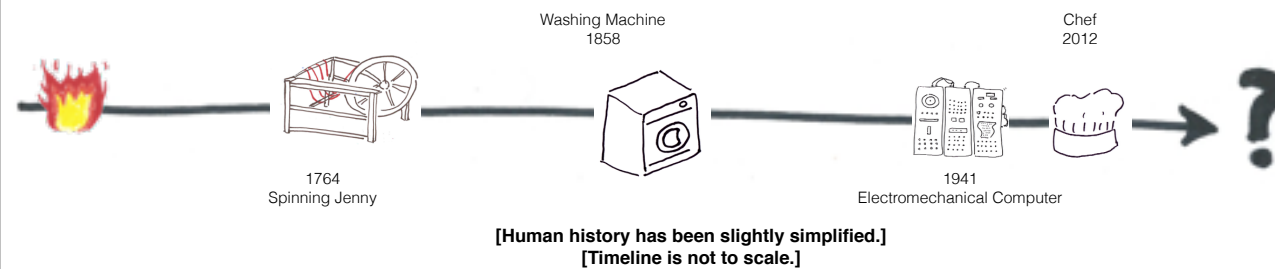
... because we need to do more and more

Devops trend

Chef

stuff like cf push

It's not just me.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

... because we need to do more and more

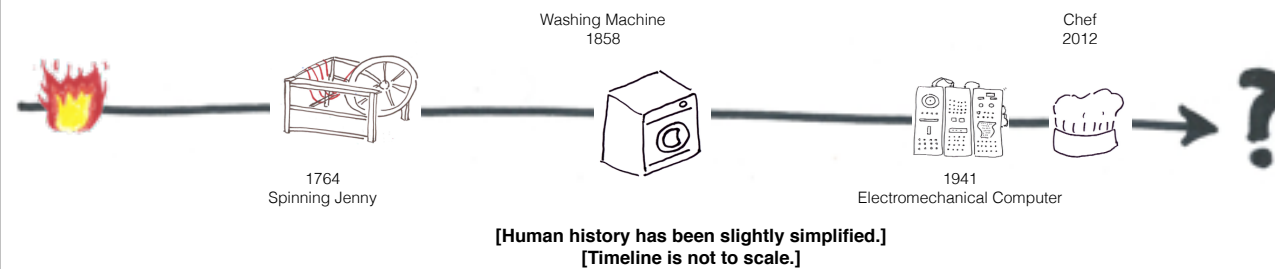
Devops trend

Chef

stuff like cf push

It's not just me.

More and more is automated.



#automationconfessions



@holly_cummins

This is a journey we've been on, as a species, for a long time.

We're automating more and more ...

... because we need to do more and more

Devops trend

Chef

stuff like cf push

So are we living the dream?

#automationconfessions



@holly_cummins

Technology giveth, and technology taketh away. We've got machines doing things for us that couldn't even be comprehended in the 1700s. And yet, we still have to do tedious tasks. I sometimes suspect that the technology allows us to generate tedious tasks at a much greater level than was possible in the 1700s.

TODO Is there a cute cute for that, like to err is human to really foul things up takes a computer, or computers allow us to generate dull tasks at a faster rate than ever before?

So are we living the dream?

Well. Not quite.

#automationconfessions



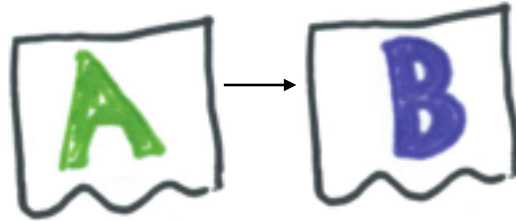
@holly_cummins

Technology giveth, and technology taketh away. We've got machines doing things for us that couldn't even be comprehended in the 1700s. And yet, we still have to do tedious tasks. I sometimes suspect that the technology allows us to generate tedious tasks at a much greater level than was possible in the 1700s.

TODO Is there a cute cute for that, like to err is human to really foul things up takes a computer, or computers allow us to generate dull tasks at a faster rate than ever before?

example

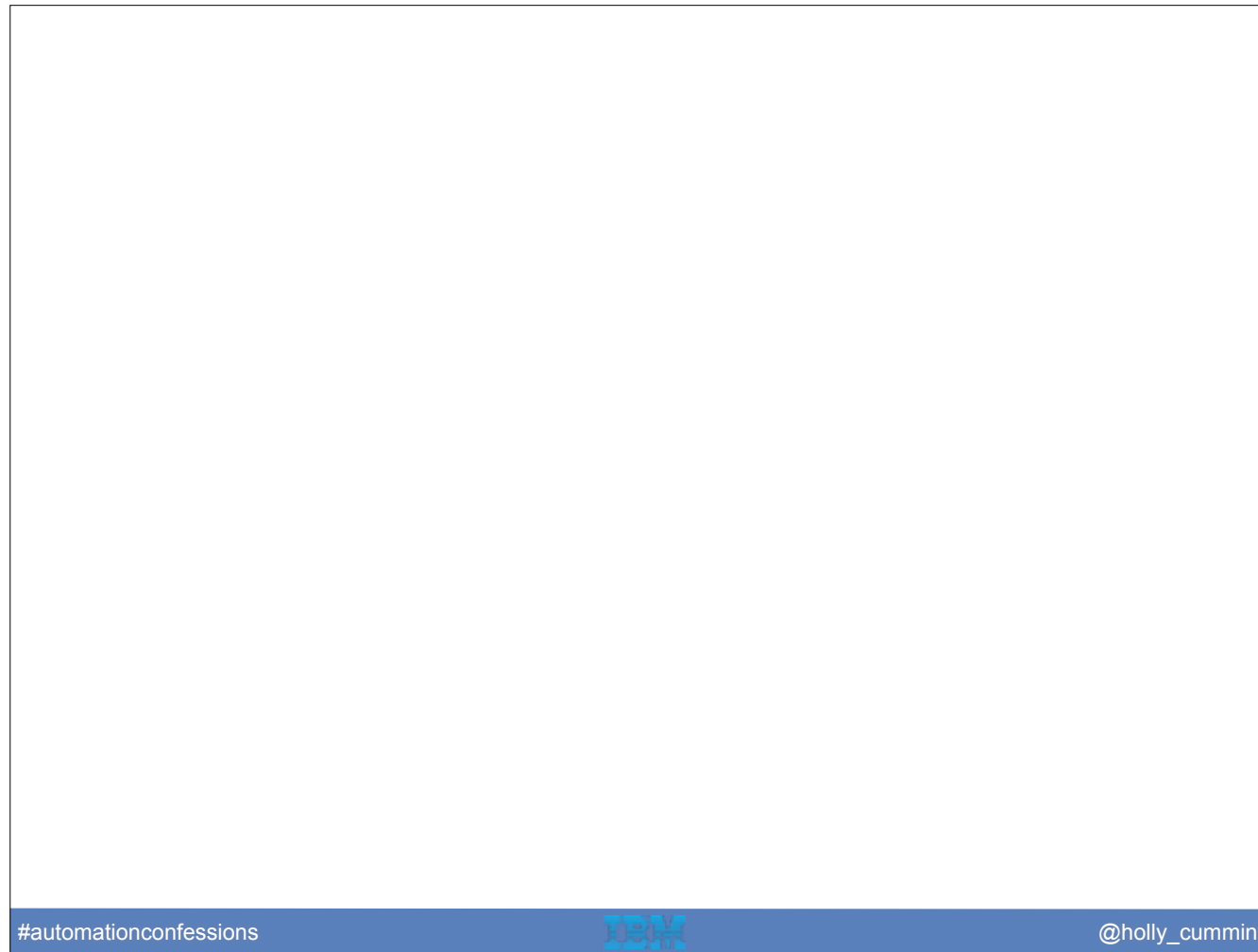
Fixing a classpath



#automationconfessions

@holly_cummins

We're an ant and Eclipse shop. We've got some clever stuff in our build which works out the right order in which to build projects from the Eclipse classpaths, and automatically adds the Eclipse .classpath to the ant classpath. This ensures we don't have to write any dependency down twice, and updating a dependency is just a question of updating one .classpath.



So far, so easy. But then we discovered we had 26 different copies of httpunit in the source tree.

Noticed it and thought "we *have* to fix this before it gets any worse," but it wasn't the kind of job that you mention in recruitment brochures.

Job of correcting each class path was pretty tedious.

It's probably the sort of thing that would have benefited from automation with a sed script.

New twist: one line perl command, never wrote it down, can't find it with google

Fixing 26 classpaths

#automationconfessions



@holly_cummins

So far, so easy. But then we discovered we had 26 different copies of httpunit in the source tree.

Noticed it and thought “we *have* to fix this before it gets any worse,” but it wasn’t the kind of job that you mention in recruitment brochures.

Job of correcting each class path was pretty tedious.

It’s probably the sort of thing that would have benefited from automation with a sed script.

New twist: one line perl command, never wrote it down, can’t find it with google



So far, so easy. But then we discovered we had 26 different copies of httpunit in the source tree.

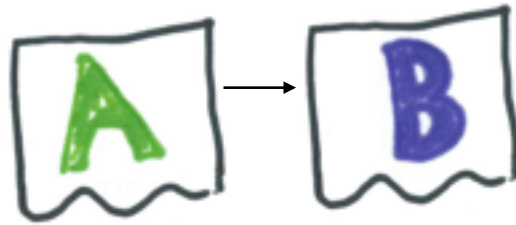
Noticed it and thought "we *have* to fix this before it gets any worse," but it wasn't the kind of job that you mention in recruitment brochures.

Job of correcting each class path was pretty tedious.

It's probably the sort of thing that would have benefited from automation with a sed script.

New twist: one line perl command, never wrote it down, can't find it with google

Epilogue



```
perl -pi -w -e 's/SEARCH_FOR/REPLACE_WITH/g;' *.txt
```

Why automate?



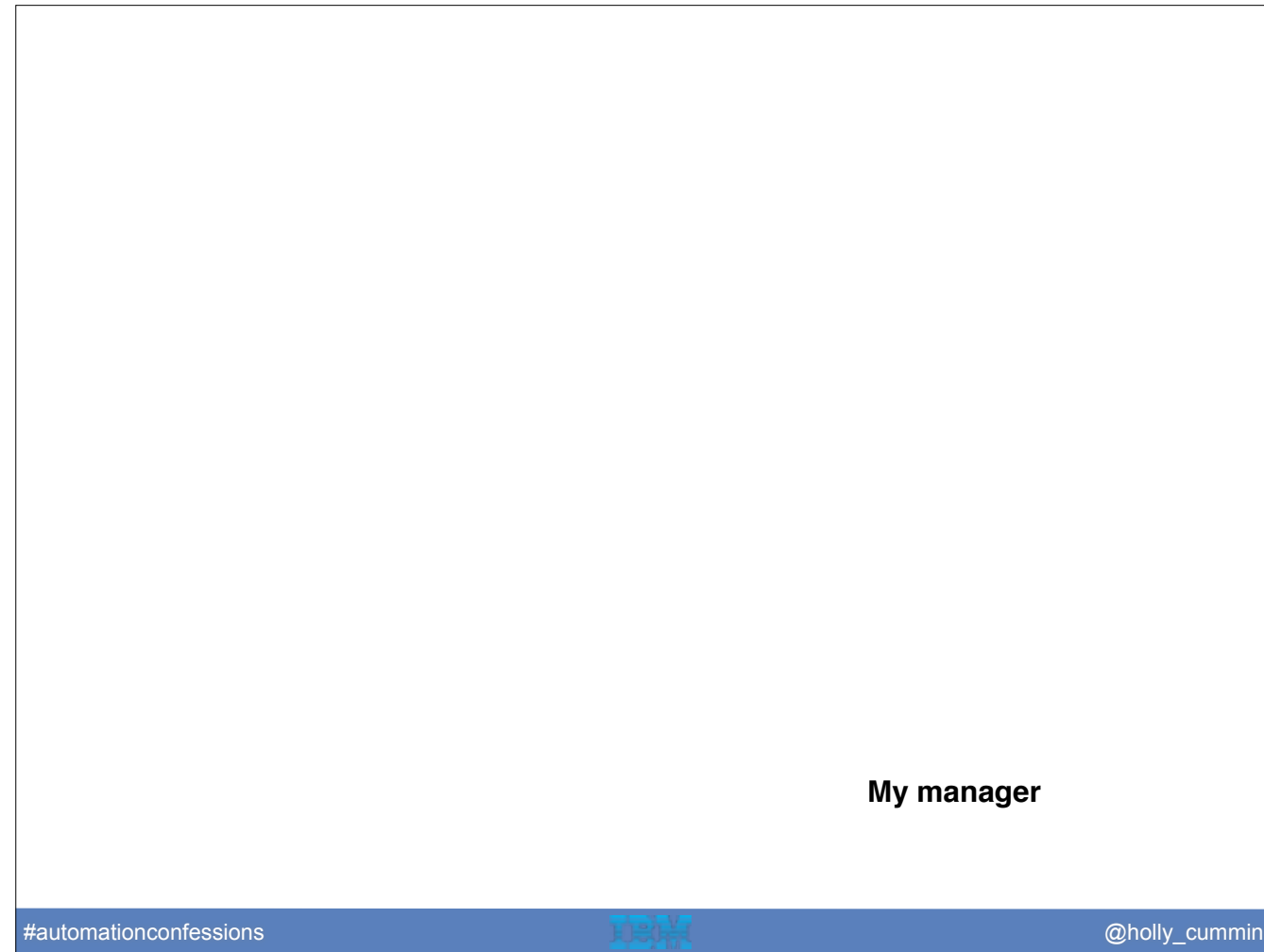
Automate because we are *lazy*

#automationconfessions



@holly_cummins

I think this story brings us nicely to the reason we automate. We automate because we don't want to be doing that. We automate because we are *lazy*.



My manager

#automationconfessions



@holly_cummins

So, here's a career tip.

Telling my boss I'm lazy doesn't tend to give good end of year feedback.

The word you want is either efficiency or sustainability.



My manager

#automationconfessions

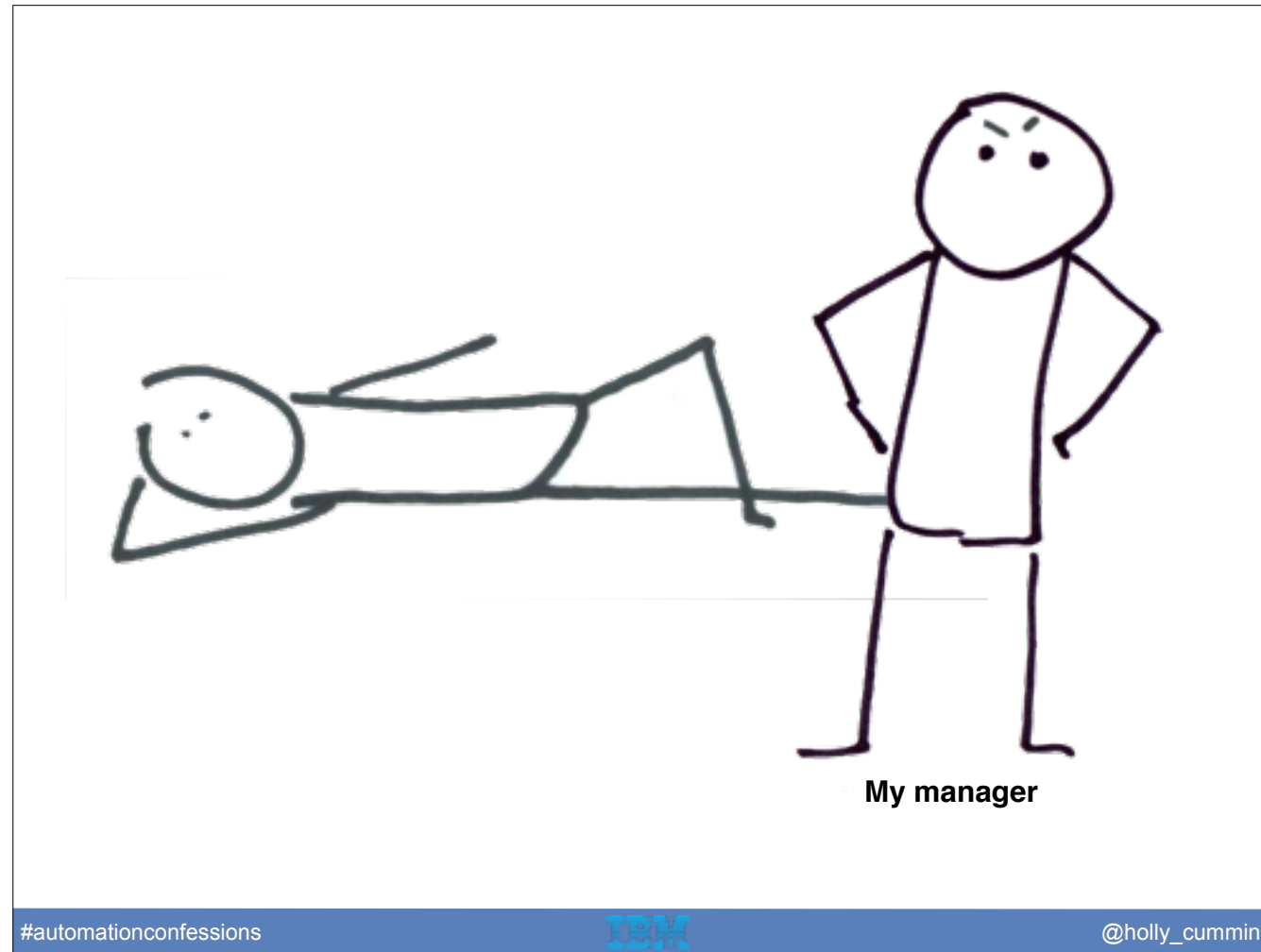


@holly_cummins

So, here's a career tip.

Telling my boss I'm lazy doesn't tend to give good end of year feedback.

The word you want is either efficiency or sustainability.



So, here's a career tip.

Telling my boss I'm lazy doesn't tend to give good end of year feedback.

The word you want is either efficiency or sustainability.

“Sustainable”



#automationconfessions



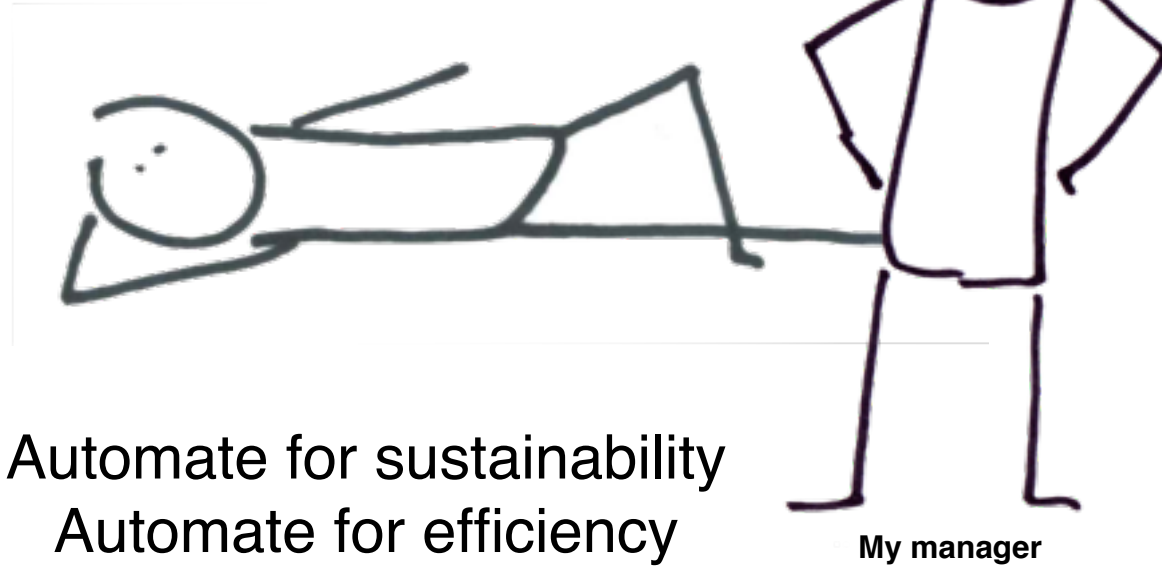
@holly_cummins

So, here's a career tip.

Telling my boss I'm lazy doesn't tend to give good end of year feedback.

The word you want is either efficiency or sustainability.

“Sustainable”



#automationconfessions

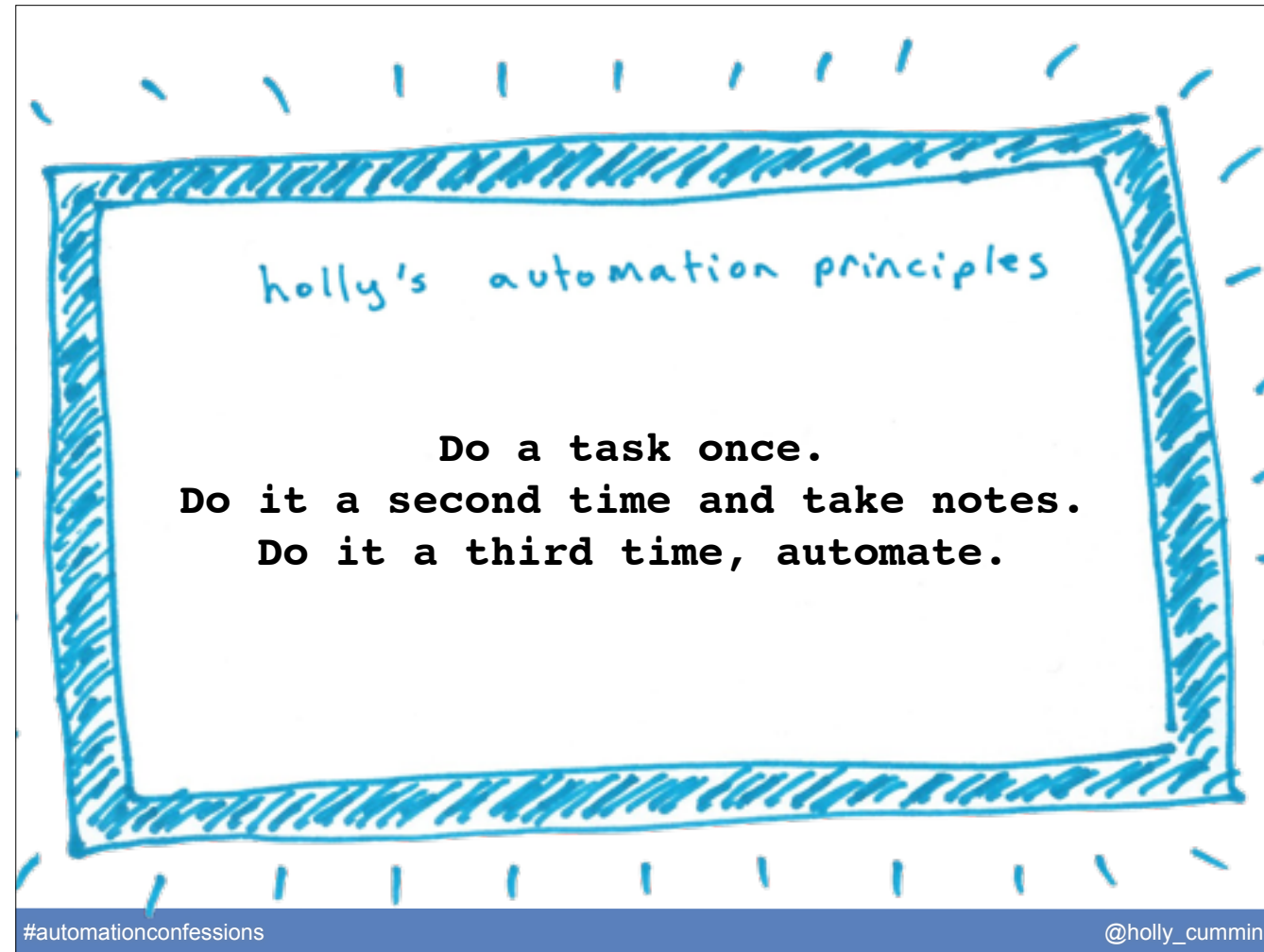


@holly_cummins

So, here's a career tip.

Telling my boss I'm lazy doesn't tend to give good end of year feedback.

The word you want is either efficiency or sustainability.



TODO strike through for trent and holly

Scale

The more of you there are, the more it is worth investing in automation

#automationconfessions



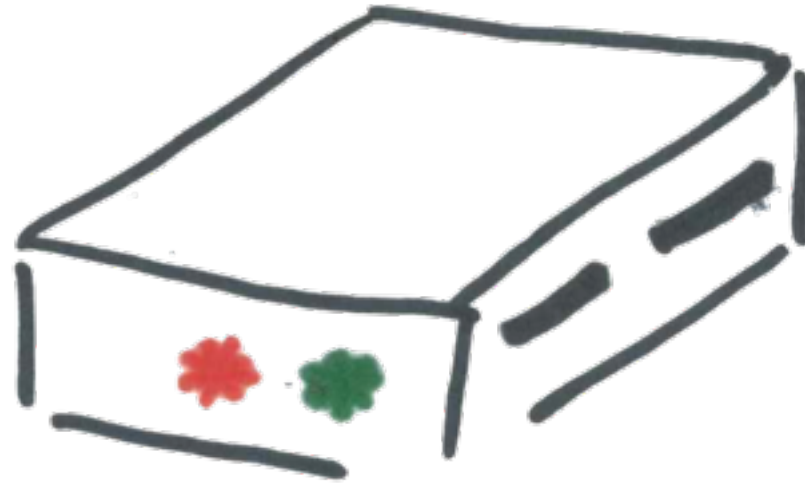
@holly_cummins

When you're operating at scale, it's very easy to trip the three times rule

Going back to the class path story, the reason that was a problem was that 26 different developers had had the idea of checking in that jar - if it had been one or two, would have been easy to fix.

example

Treat your servers like cattle, not like pets.



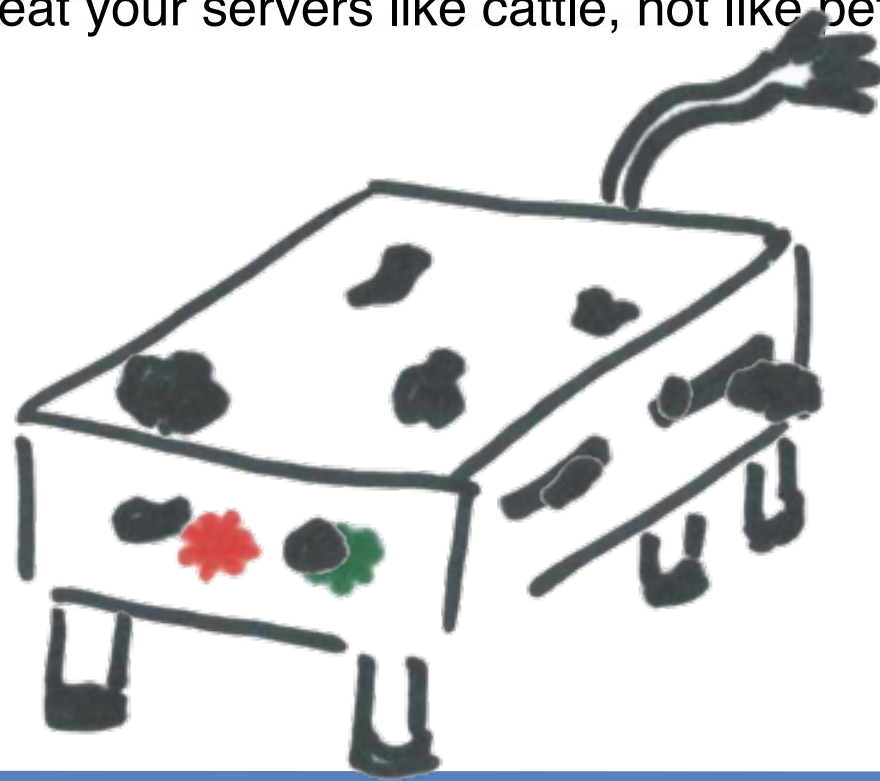
#automationconfessions

@holly_cummins

Treat your
this has some interactions with robustness, too

example

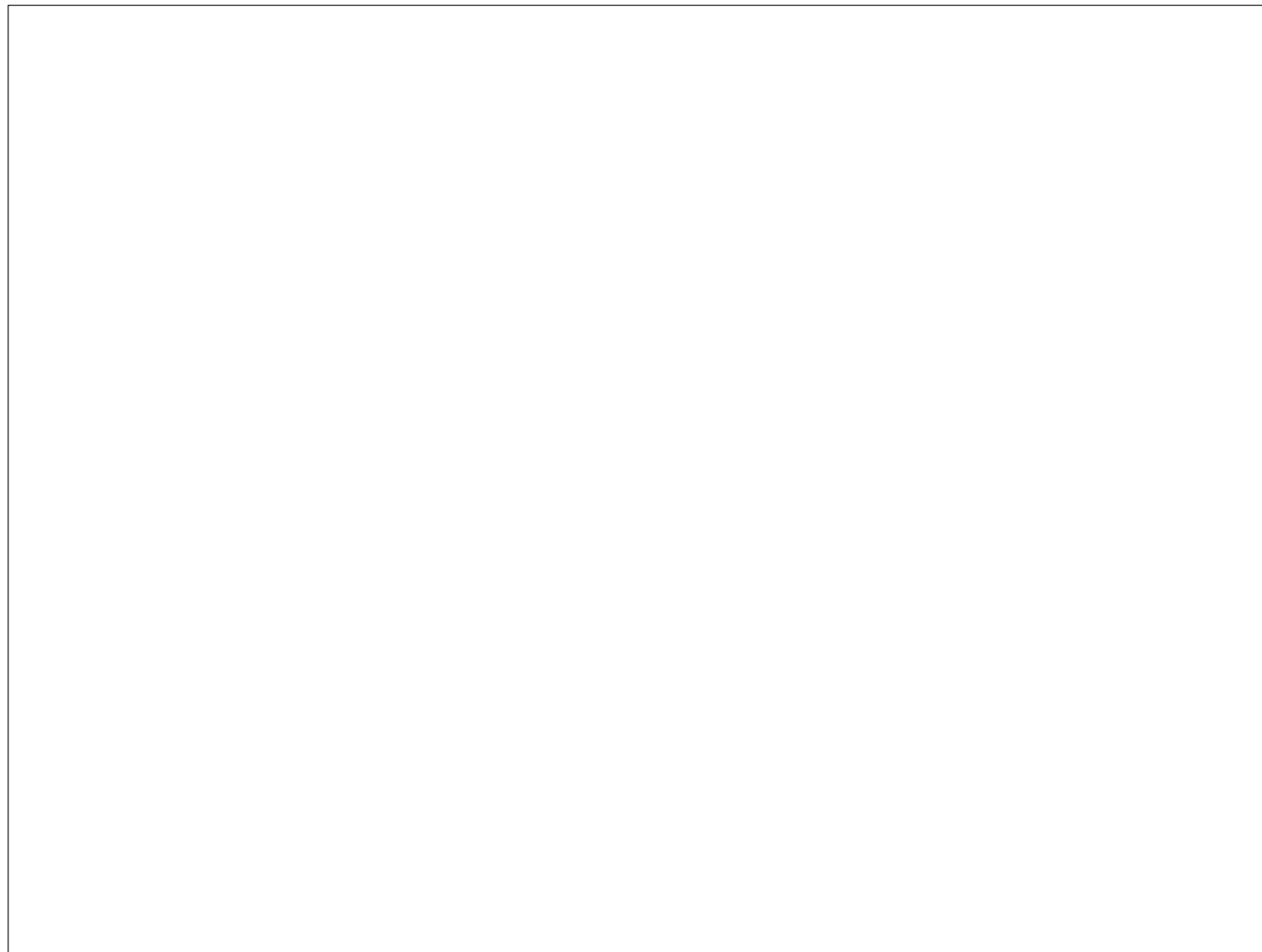
Treat your servers like cattle, not like pets.



#automationconfessions

@holly_cummins

Treat your
this has some interactions with robustness, too



For one server, it doesn't really matter. For a lot of servers, it matters a lot.

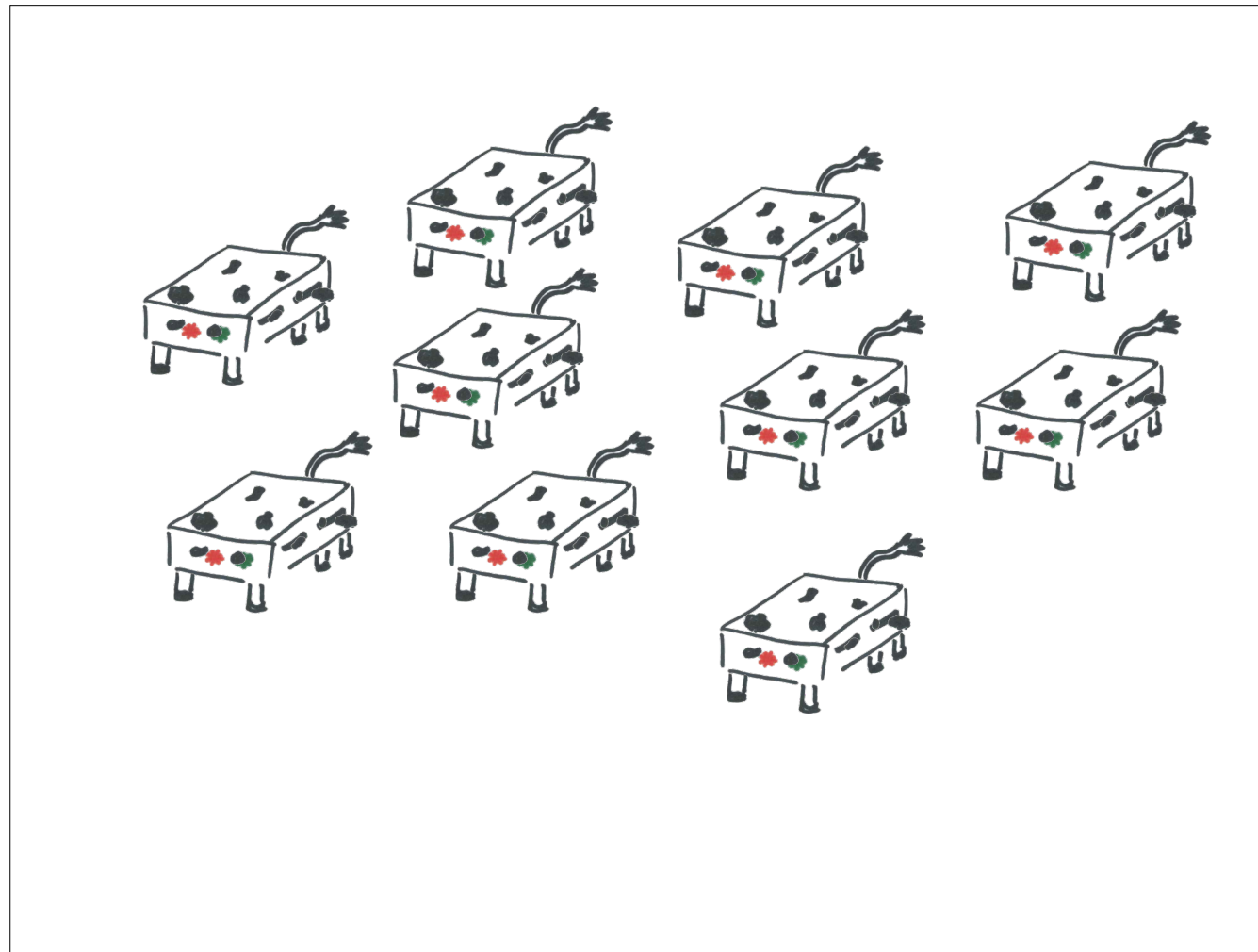
Our build farm is 500 servers, so I'm not going to illustrate that.

It matters when you're setting them up, and it matters more when they go wrong.

If they're not unique or special, you can cope with ones going away - you just get a replacement back.

So it's about scale, but also about managing failure.

There's another benefit to this way of managing servers, which is that you know servers won't end up misconfigured. We have a lot of build servers, and when we set up them up by hand, there's always some niggling thing which is wrong.



For one server, it doesn't really matter. For a lot of servers, it matters a lot.

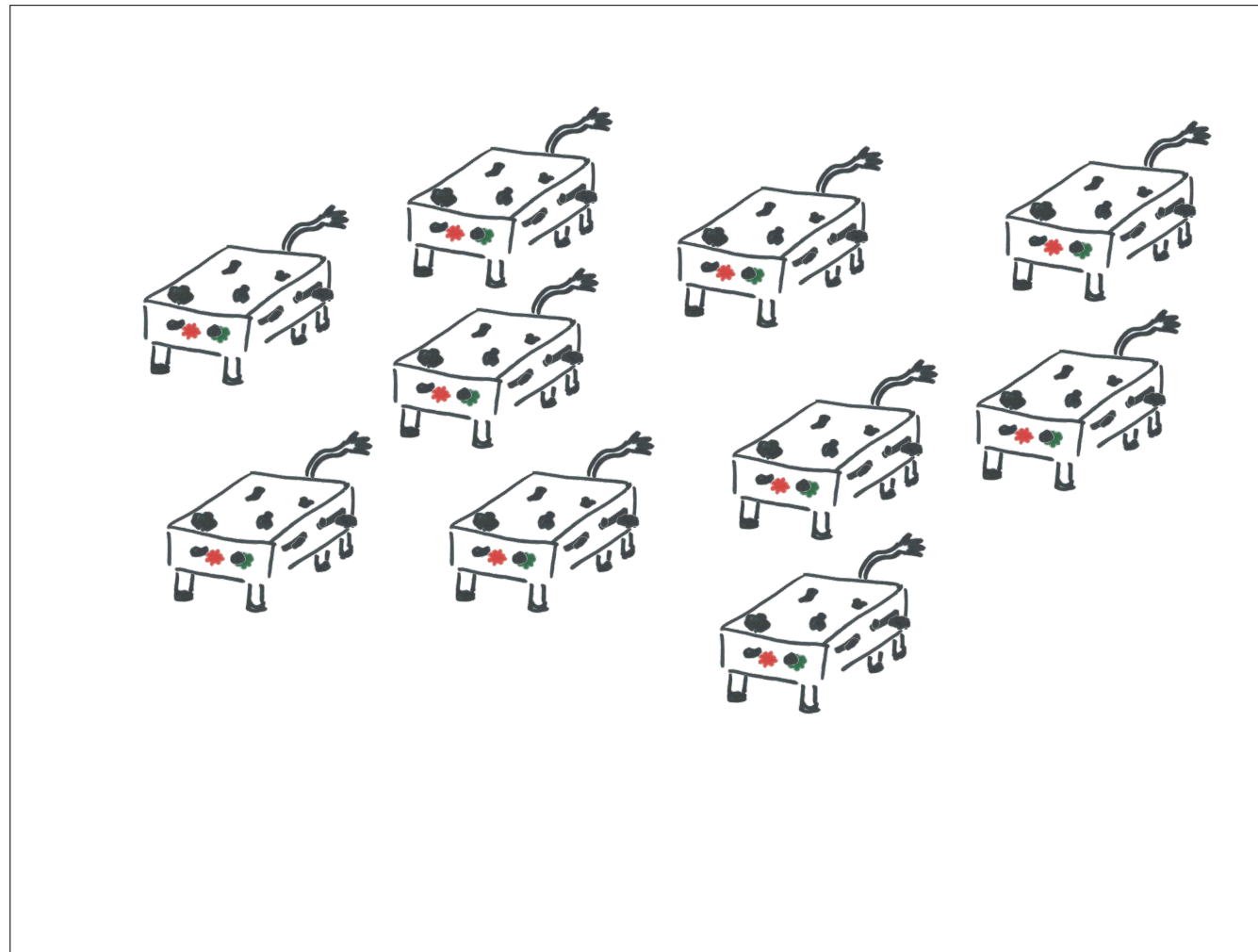
Our build farm is 500 servers, so I'm not going to illustrate that.

It matters when you're setting them up, and it matters more when they go wrong.

If they're not unique or special, you can cope with ones going away - you just get a replacement back.

So it's about scale, but also about managing failure.

There's another benefit to this way of managing servers, which is that you know servers won't end up misconfigured. We have a lot of build servers, and when we set up them up by hand, there's always some niggling thing which is wrong.



For one server, it doesn't really matter. For a lot of servers, it matters a lot.

Our build farm is 500 servers, so I'm not going to illustrate that.

It matters when you're setting them up, and it matters more when they go wrong.

If they're not unique or special, you can cope with ones going away - you just get a replacement back.

So it's about scale, but also about managing failure.

There's another benefit to this way of managing servers, which is that you know servers won't end up misconfigured. We have a lot of build servers, and when we set up them up by hand, there's always some niggling thing which is wrong.

Why Automate?



#automationconfessions



@holly_cummins

Part of the reason we treat our servers like cattle is because we're lazy, but it's also because we want to get it right

Why Automate?



#automationconfessions



@holly_cummins

Part of the reason we treat our servers like cattle is because we're lazy, but it's also because we want to get it right

Why Automate?



Automate
because you
care

#automationconfessions



@holly_cummins

Part of the reason we treat our servers like cattle is because we're lazy, but it's also because we want to get it right



Principle: If you care about it, automate it.

if take one thing away, make it this one

Why Automate?



Why Automate?



Automate
because you
care that it
happens at all

example
Marvin

#automationconfessions

@holly_cummins

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

Marvin

- Build breaks (including test failures) should be reverted from the stream

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

Marvin

- Build breaks (including test failures) should be reverted from the stream
- No one wanted to be the one who did it

#automationconfessions

@holly_cummins

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

Marvin

- Build breaks (including test failures) should be reverted from the stream
- No one wanted to be the one who did it
- It didn't happen

#automationconfessions

@holly_cummins

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

Marvin

- Build breaks (including test failures) should be reverted from the stream
- No one wanted to be the one who did it
- It didn't happen
- Solution: Automate!

#automationconfessions

@holly_cummins

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

Marvin

- Build breaks (including test failures) should be reverted from the stream
- No one wanted to be the one who did it
- It didn't happen
- Solution: Automate!



#automationconfessions

@holly_cummins

Scale is thing here, too, because if you have of the order of a hundred or so developers delivering code, the odds of a mistake are higher than if there are two of you

example

End of iteration demos



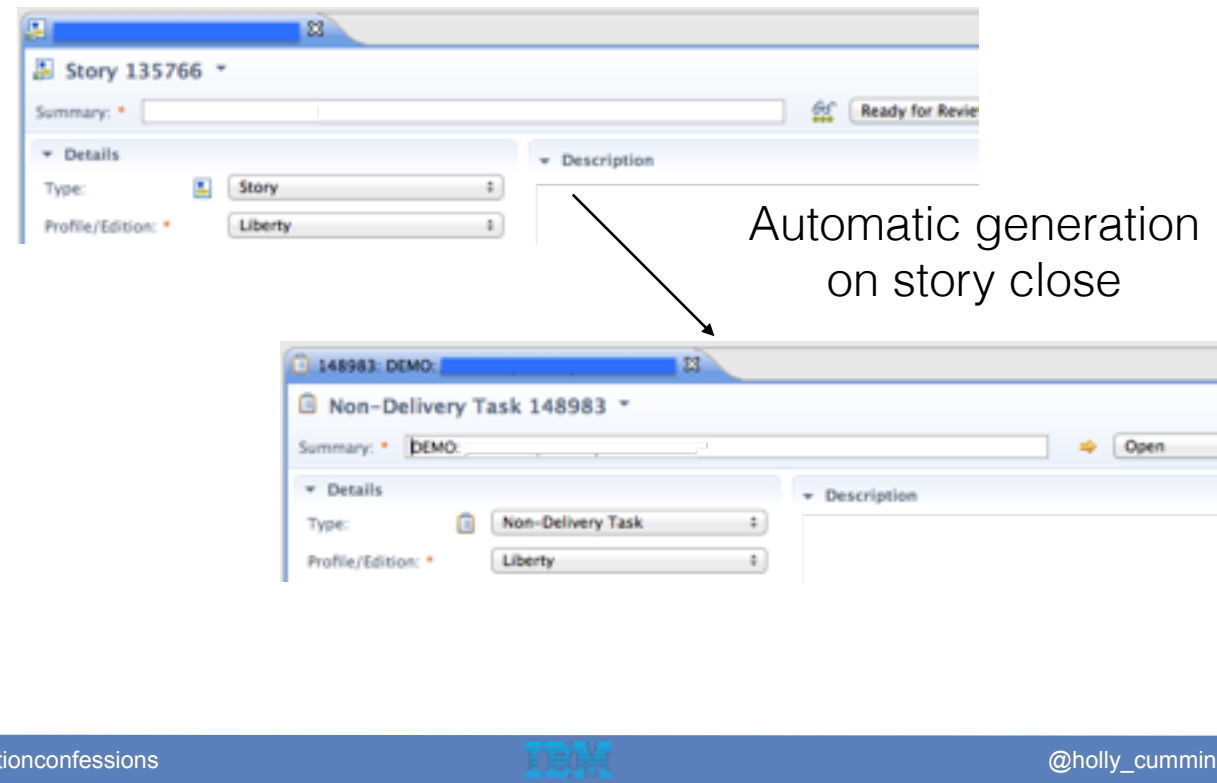
#automationconfessions

@holly_cummins

Example - end of iteration demos, some people loved doing them and some people ignored them. At the kind of scale we're working on, we were relying on people to volunteer to do them, and so a lot of stories were missing being demoed.

This picture, by the way, is showing a whiteboard with someone not doing an end of iteration demo, but that's kind of hard to draw.

End of iteration demos



So we put in an automation, and changed RTC to automatically spawn a work item for an end of iteration demo when we close a story, got way more stories.

This is an example of a really simple automation - the only thing that's happening is a work item.



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion
Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

Automating to improve
the outcome rather than
to save effort:



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion

Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

Automating to improve the outcome rather than to save effort:

- Write the Rational Team Concert plugin (one-off)



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion
Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

Automating to improve the outcome rather than to save effort:

- Write the Rational Team Concert plugin (one-off)
- Close the demo task (every story)



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion
Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

Automating to improve the outcome rather than to save effort:

- Write the Rational Team Concert plugin (one-off)
- Close the demo task (every story)
- Actually do the demo (every story)



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion
Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

Automating to improve the outcome rather than to save effort:

- Write the Rational Team Concert plugin (one-off)
- Close the demo task (every story)
- Actually do the demo (every story)
- But ... we get guaranteed demos now.



Clearly automating to improve outcome rather than save effort. It was actually a lot more effort, and it's more ongoing effort, too.

Not all success stories here, our iterations should exit with no must defects but doesn't ever happen because it's only an assertion
Defect management

We shouldn't exit an iteration with any must-fix defects.

Priority inflation, scaling, we do exit, because nothing in our process forces that step

We actually know the product is shippable, so it's a mismatch

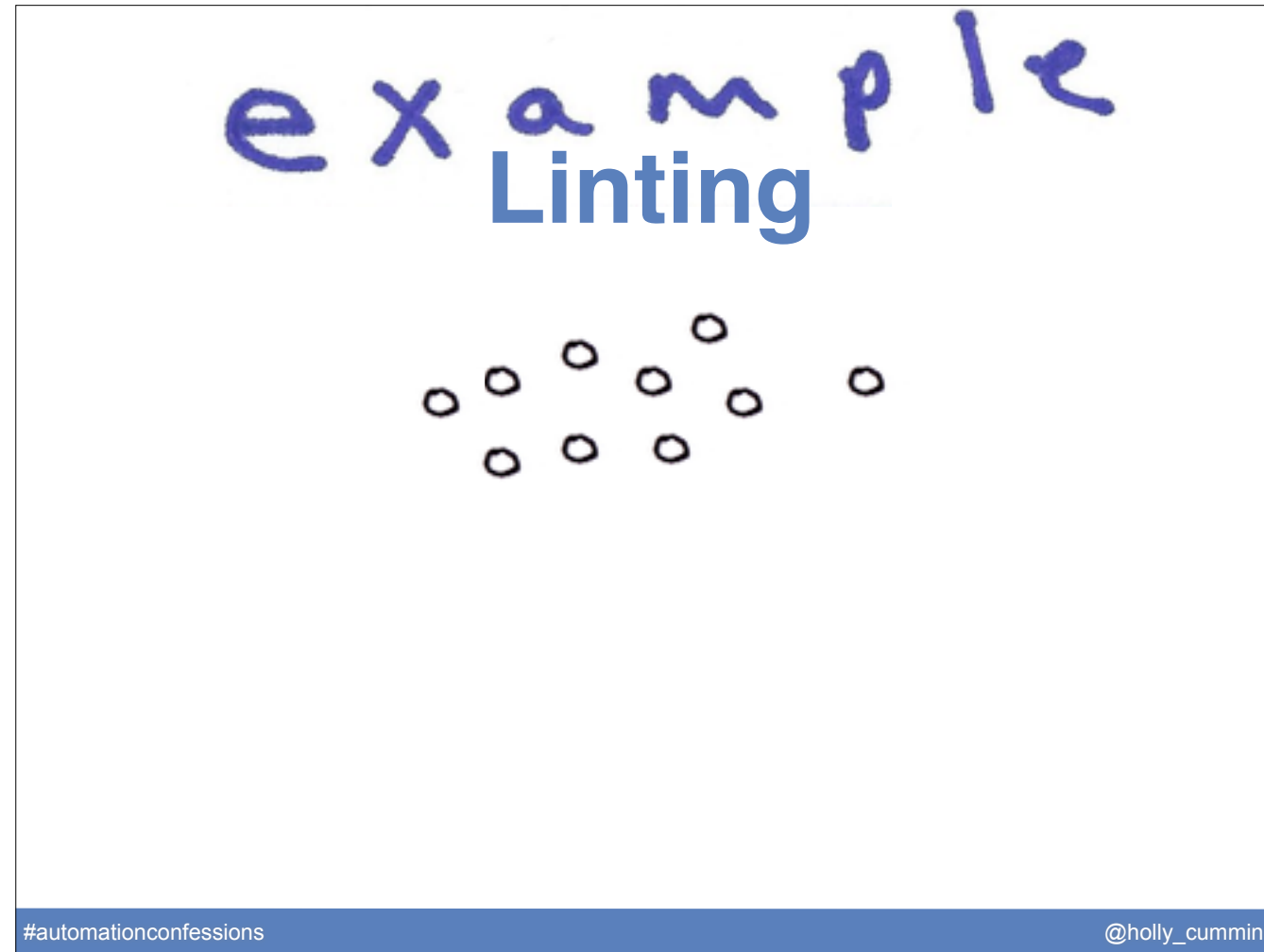
Why Automate?



Why Automate?



Automate
because you
care that it's
done right



If a find bugs failure gets delivered, marvin automatically backs it out

Findbugs, js linting

Other examples:

Suspending changesets

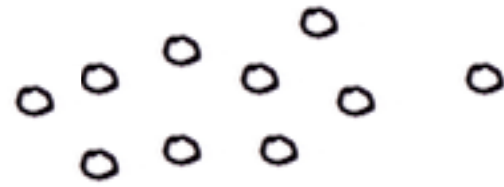
Semantic versioning

Release deployments can be unexpectedly complex, easy to forget one step

TODO where does this fit?

ie our build monitoring - A skilled person can do it better than a machine. An unskilled person will do it worse than a machine.

example Linting



- A computer can notice problems that a human can't (like misspelled variable names)

#automationconfessions

@holly_cummins

If a find bugs failure gets delivered, marvin automatically backs it out

Findbugs, js linting

Other examples:

Suspending changesets

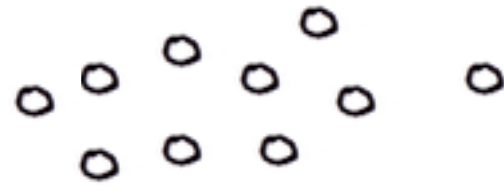
Semantic versioning

Release deployments can be unexpectedly complex, easy to forget one step

TODO where does this fit?

ie our build monitoring - A skilled person can do it better than a machine. An unskilled person will do it worse than a machine.

example Linting



- A computer can notice problems that a human can't (like misspelled variable names)
- We love findbugs

#automationconfessions

@holly_cummins

If a find bugs failure gets delivered, marvin automatically backs it out

Findbugs, js linting

Other examples:

Suspending changesets

Semantic versioning

Release deployments can be unexpectedly complex, easy to forget one step

TODO where does this fit?

ie our build monitoring - A skilled person can do it better than a machine. An unskilled person will do it worse than a machine.

example Deployment

#automationconfessions

@holly_cummins

Suspending changesets

Release deployments can be unexpectedly complex, easy to forget one step

example Deployment

- Should be easy, right?

#automationconfessions

@holly_cummins

Suspending changesets

Release deployments can be unexpectedly complex, easy to forget one step

example Deployment

- Should be easy, right?
- Always one step that gets forgotten ...

#automationconfessions

@holly_cummins

Suspending changesets

Release deployments can be unexpectedly complex, easy to forget one step

Why automate?



This is another one that I don't put in my performance appraisals.

I didn't want to leave it out, though, because it's really true, and this talk is all about confessions.

It's maybe not a *good* reason to automate, but it does happen.

If you're thinking about how interesting the problem is, rather than the long term picture, that's maybe a red flag

It's maybe not the best use of your time and your employer's time

Why automate?



This is another one that I don't put in my performance appraisals.

I didn't want to leave it out, though, because it's really true, and this talk is all about confessions.

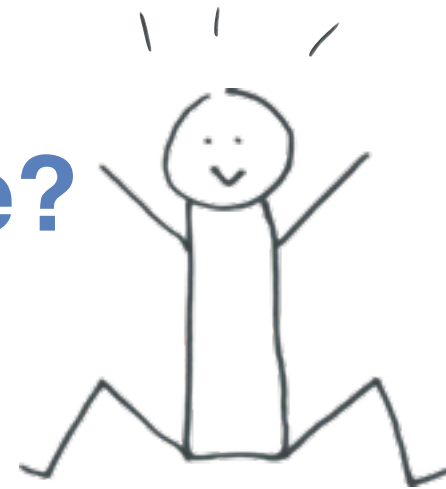
It's maybe not a *good* reason to automate, but it does happen.

If you're thinking about how interesting the problem is, rather than the long term picture, that's maybe a red flag

It's maybe not the best use of your time and your employer's time

Why automate?

Automate
because it's
fun



This is another one that I don't put in my performance appraisals.

I didn't want to leave it out, though, because it's really true, and this talk is all about confessions.

It's maybe not a *good* reason to automate, but it does happen.

If you're thinking about how interesting the problem is, rather than the long term picture, that's maybe a red flag

It's maybe not the best use of your time and your employer's time

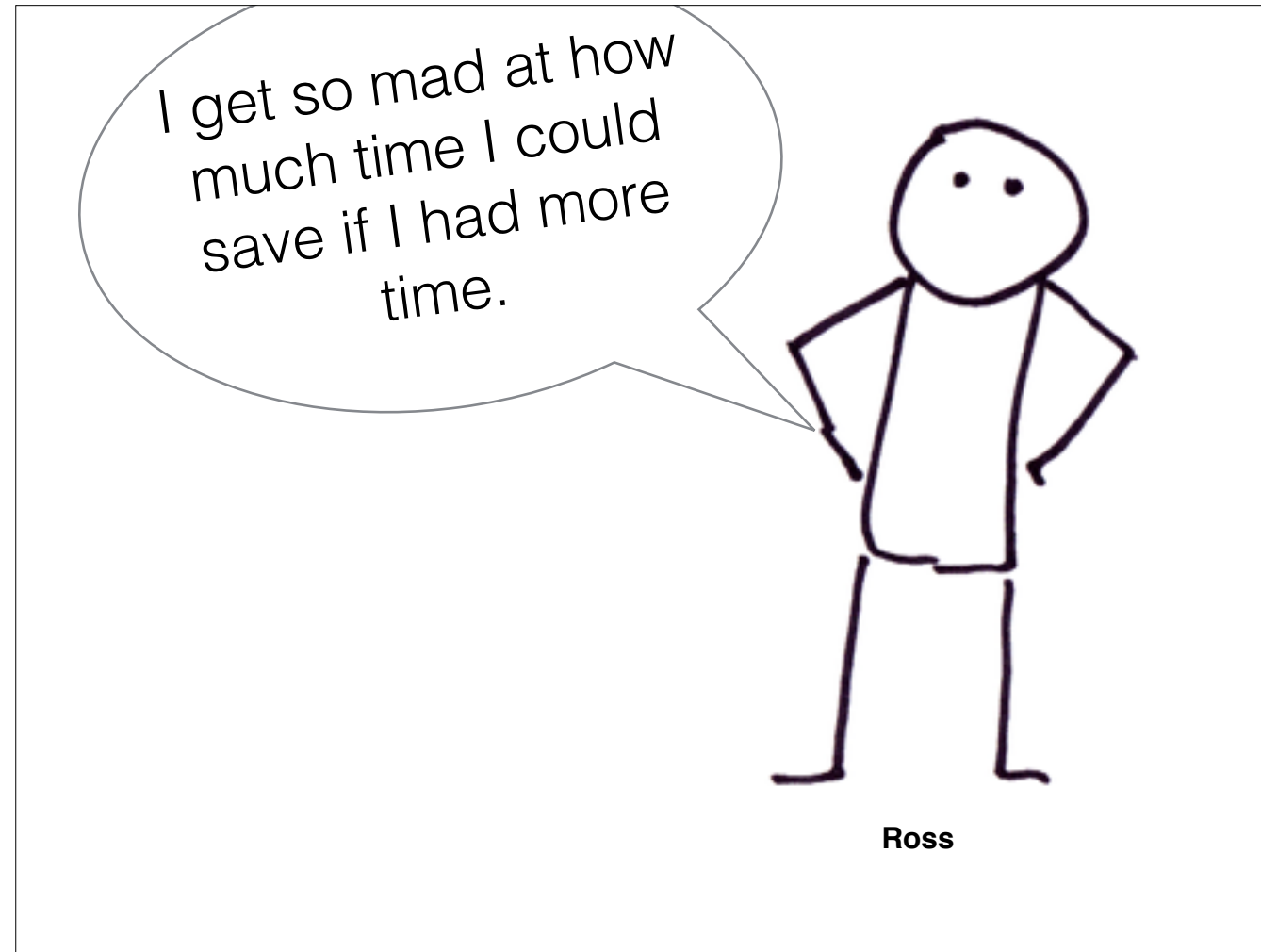
Why *not* automate?



Ross

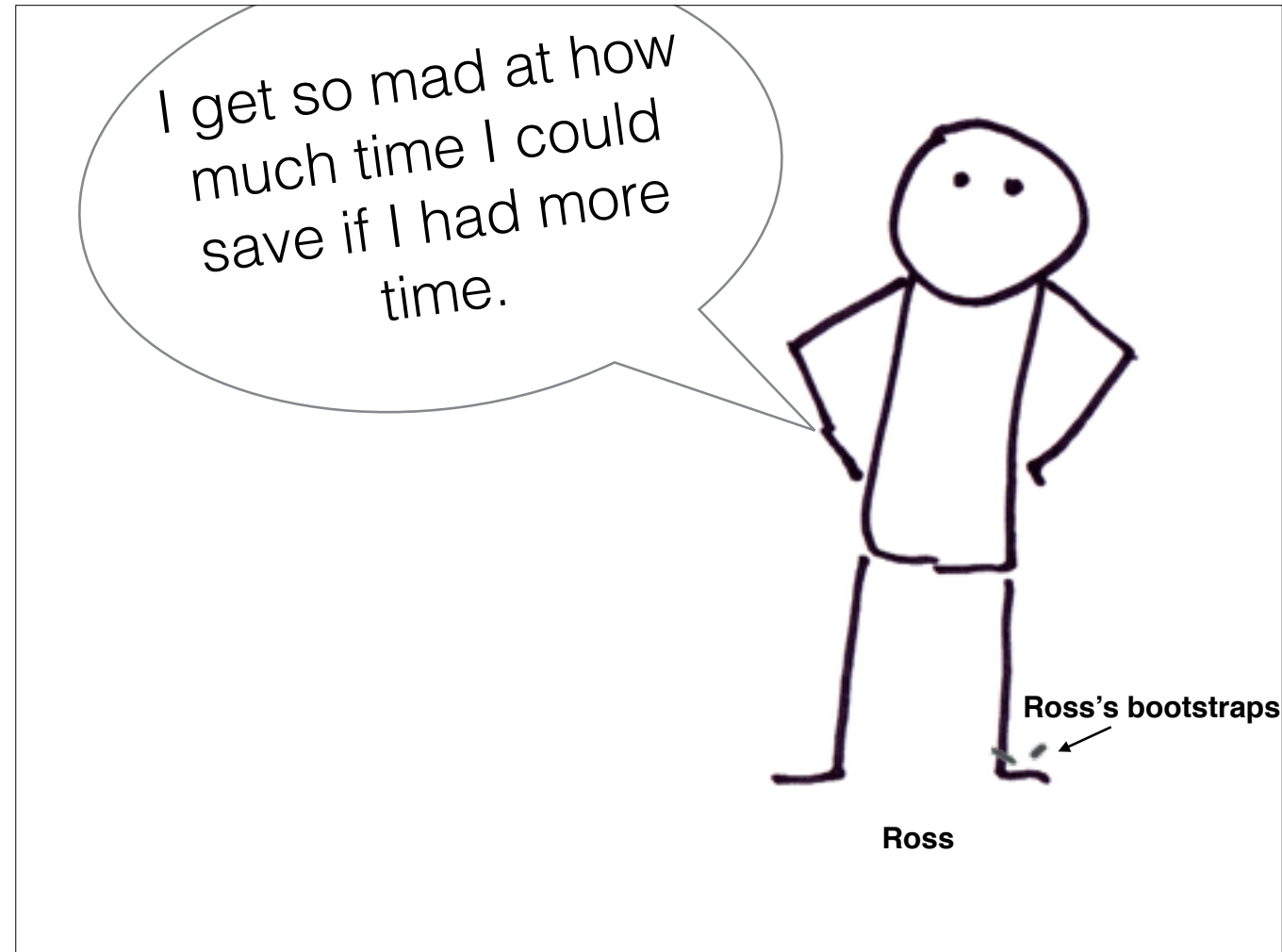
Bootstrapping problem, sometimes we can't make that initial investment.

Sometimes what we think is a bootstrapping problem is actually a more fundamental problem.



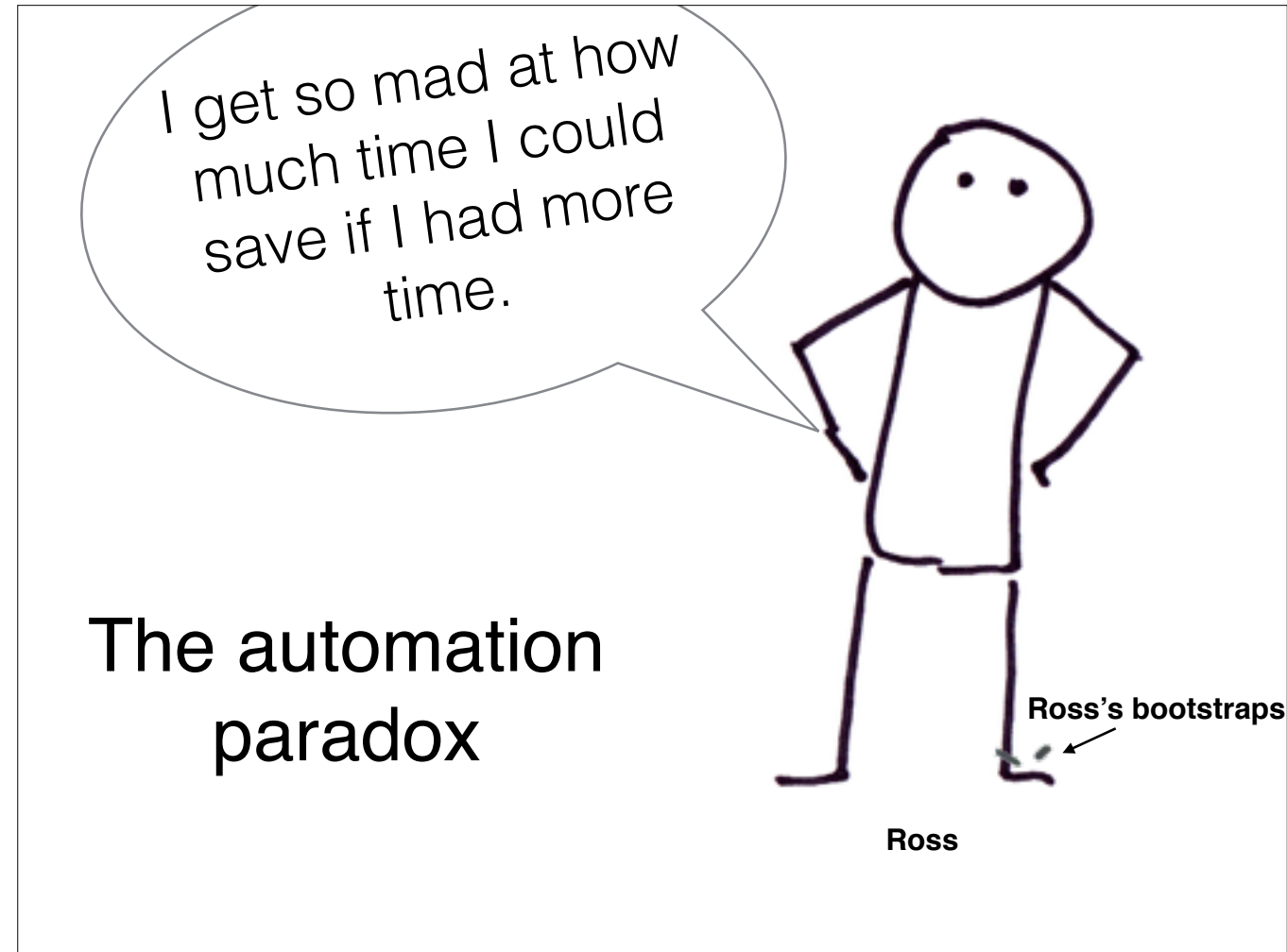
Bootstrapping problem, sometimes we can't make that initial investment.

Sometimes what we think is a bootstrapping problem is actually a more fundamental problem.



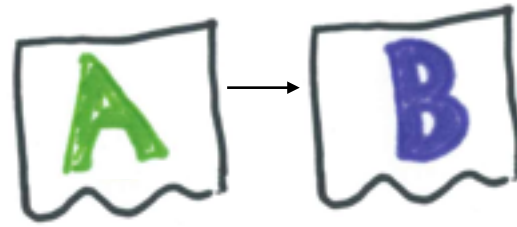
Bootstrapping problem, sometimes we can't make that initial investment.

Sometimes what we think is a bootstrapping problem is actually a more fundamental problem.



Bootstrapping problem, sometimes we can't make that initial investment.

Sometimes what we think is a bootstrapping problem is actually a more fundamental problem.

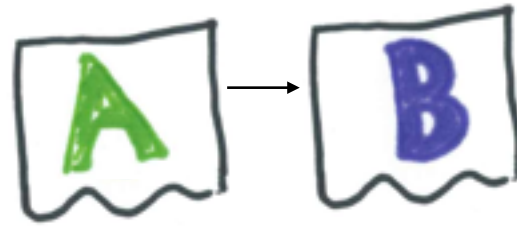


#automationconfessions



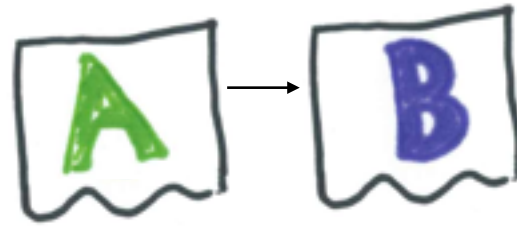
@holly_cummins

I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...



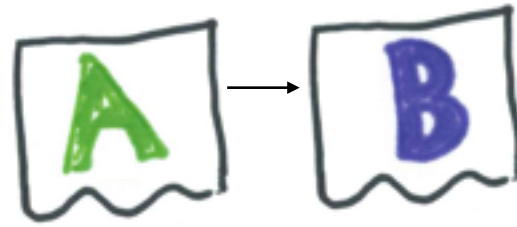
find something | exec something {} + ; ?!

I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...



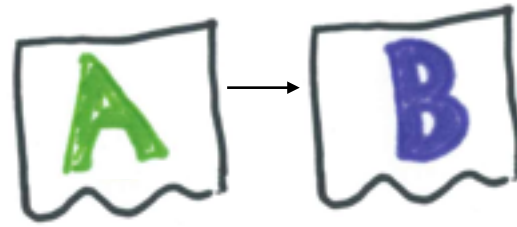
```
find something | exec something {} + ; ?!  
for ... then sed something; mv $i.bak $i; ... ?
```

I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...



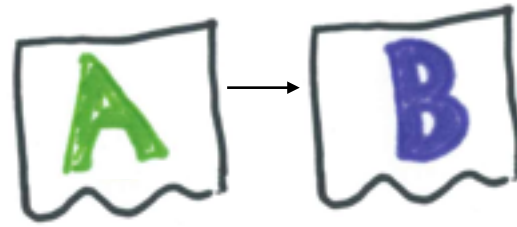
```
find something | exec something {} + ; ?!  
for ... then sed something; mv $i.bak $i; ... ?  
sedder.sh and dosed.sh and ... ?
```

I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...



```
find something | exec something {} + ; ?!  
for ... then sed something; mv $i.bak $i; ... ?  
sedder.sh and dosed.sh and ... ?  
perl ... um .. something *.txt?
```

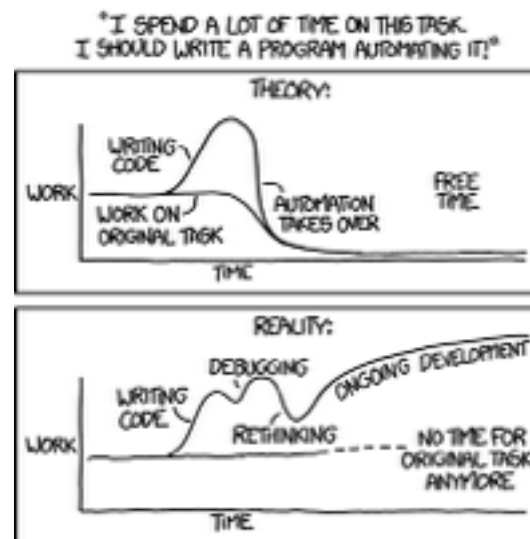
I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...



```
find something | exec something {} + ; ?!  
for ... then sed something; mv $i.bak $i; ... ?  
sedder.sh and dosed.sh and ... ?  
perl ... um .. something *.txt?
```

I've given this talk a few times, and every time I think about giving our graduate that search job I feel guilty. BUT ...
I've invested so much time in writing script libraries to manage sed, in googling for scripts to search replace ...

Why *not* automate



thanks (for everything!), xkcd.com

I saw this on a colleague's wall, and I just felt this sinking feeling as I recognised so many projects I'd worked on.



TODO make robot be juggling
Automation can be a good friend



TODO make robot be juggling
Automation can be a good friend

The Reality



The Reality



The Reality

Think about
support.



The Reality

Think about
support.
Think about
robustness.

example Rosie



#automationconfessions

@holly_cummins

Move to CD makes it way simpler externally, more complex internally

Right way, right?

TODO example logo

Key points:

Scale of team doing deliveries, some basic process things like "did you do a sandbox build"? "are you using the right kind of work item"? get lost. Tool to replace unsustainable - there's that word again! - process of manual approvals, and guide team through process.



Rosie ate
the world.

#automationconfessions



@holly_cummins

Rosie ate the world.

Total time sink, one of us, then two of us, then three of us, all our other development stopped.

A bit like don't fight a land war in asia. Don't do automations which involve javascript.

Or at least, be very cautious of automations with user interfaces.



Rosie ate
the world.

(Well, my corner of it.)

#automationconfessions



@holly_cummins

Rosie ate the world.

Total time sink, one of us, then two of us, then three of us, all our other development stopped.

A bit like don't fight a land war in asia. Don't do automations which involve javascript.

Or at least, be very cautious of automations with user interfaces.

Questions to ask *before* automating

Questions to ask *before* automating

- Who can maintain this automation?

Questions to ask *before* automating

- Who can maintain this automation?
- Who can manage this automation?

Questions to ask *before* automating

- Who can maintain this automation?
- Who can manage this automation?
- Will it be robust? What if things change?

Questions to ask *before* automating

- Who can maintain this automation?
- Who can manage this automation?
- Will it be robust? What if things change?
- How much will it cost?

Checkpoint during automation

#automationconfessions



@holly_cummins

Checkpoint during automation

- How do I define success?

Checkpoint during automation

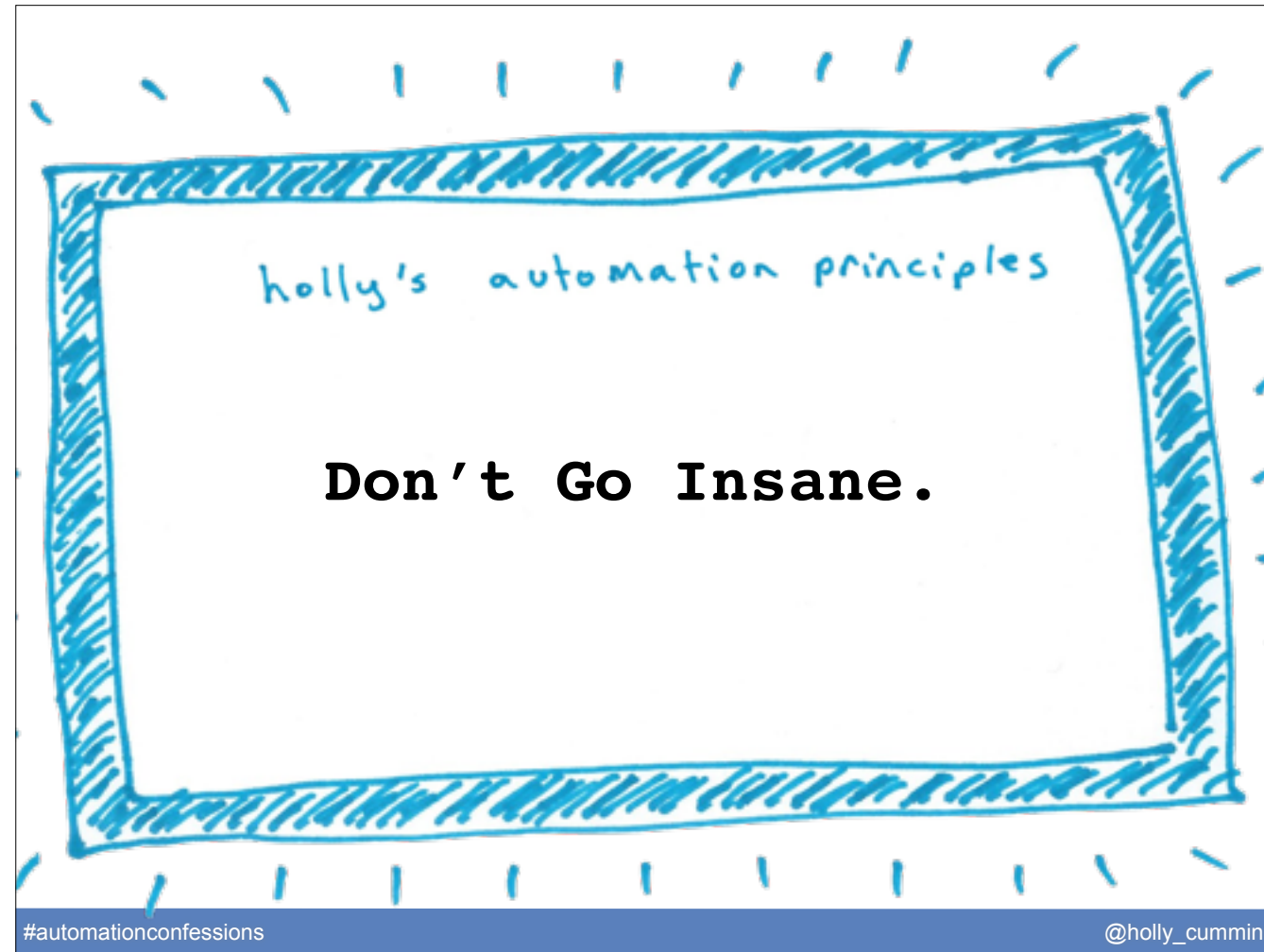
- How do I define success?
- Have I got completion criteria?

Checkpoint during automation

- How do I define success?
- Have I got completion criteria?
- Is this still worth it?

Checkpoint during automation

- How do I define success?
- Have I got completion criteria?
- Is this still worth it?
- Am I still saving time?



Keep your sense of proportion.

Before - and during automating, checkpoint regularly to make sure the cost is still acceptable.

Remember the bus

#automationconfessions



@holly_cummins

Example - my automation to reconsume Aries, rest of team scared to touch it and it breaks every now and then

Remember the bus

A clever automation isn't so clever if only one person understands it...

#automationconfessions

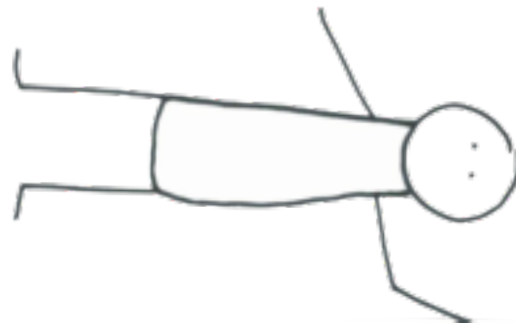


@holly_cummins

Example - my automation to reconsume Aries, rest of team scared to touch it and it breaks every now and then

Remember the bus

A clever automation isn't so clever if only one person understands it...
... and they get hit by a bus.

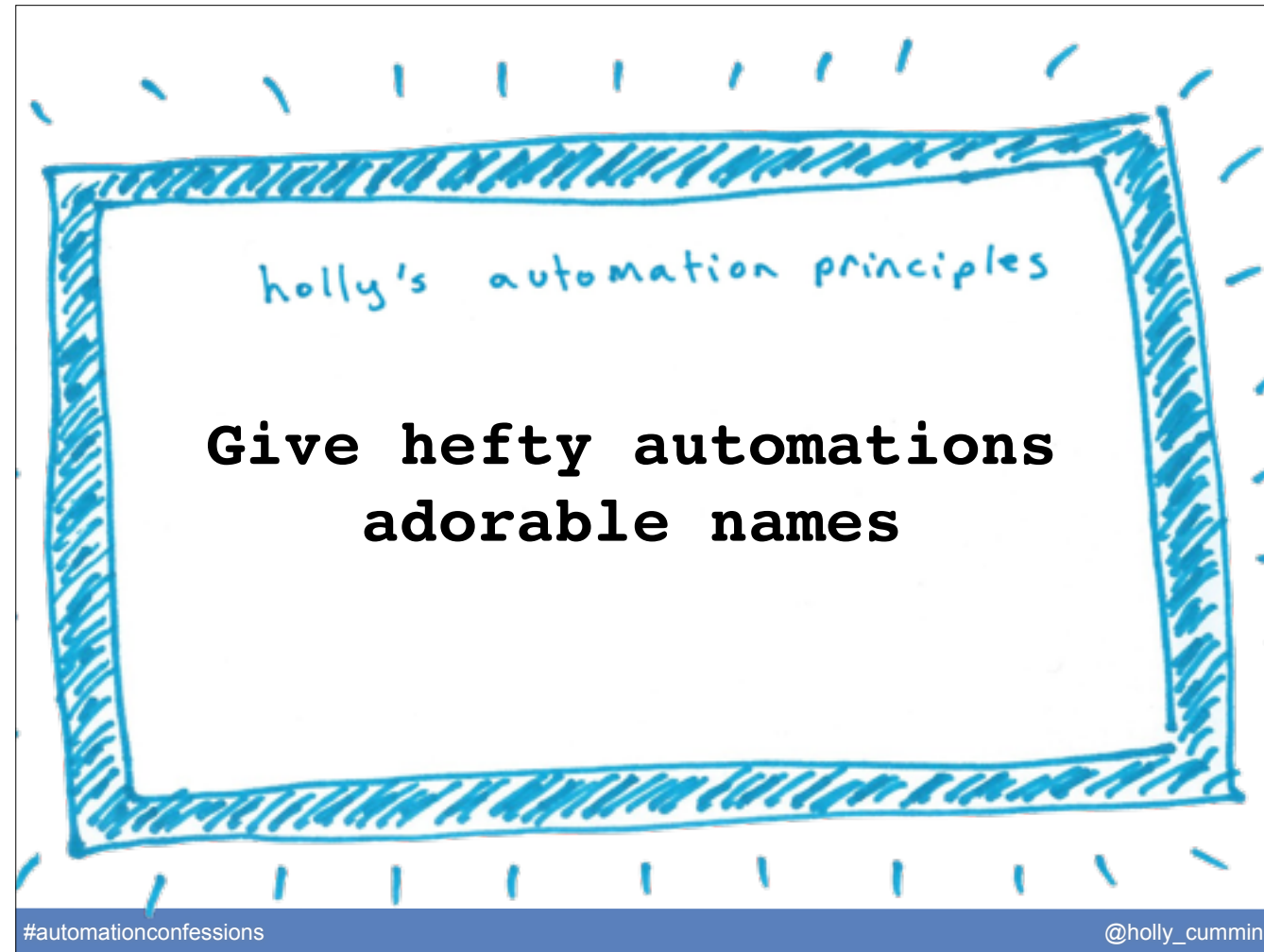


#automationconfessions



@holly_cummins

Example - my automation to reconsume Aries, rest of team scared to touch it and it breaks every now and then

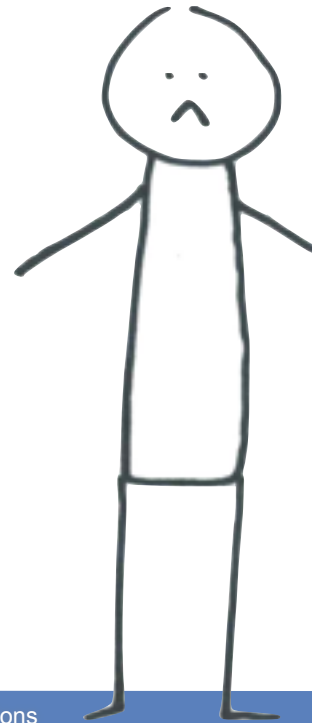


Dan North has Conan the deployer.

Rosie maid from Jetsons

So, no matter how much you checkpoint, some automations will be big. So This one is important. If this thing is going to take over your life, it better have a name, right? So you can say to your boss "I was up until midnight working on Rosie last night ..."

Why *not* automate

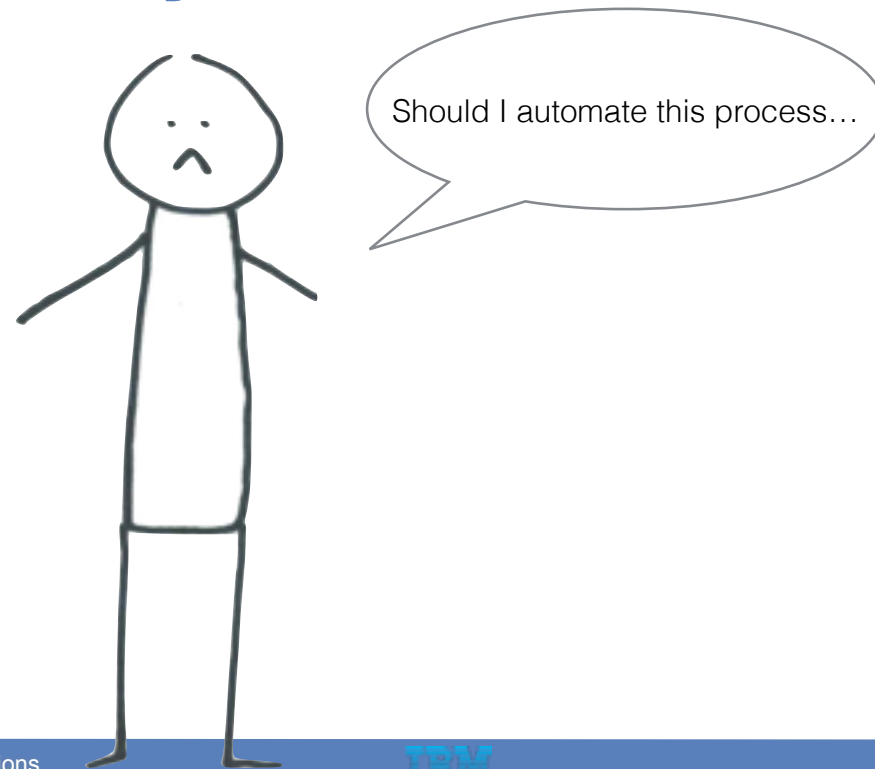


#automationconfessions



@holly_cummins

Why *not* automate

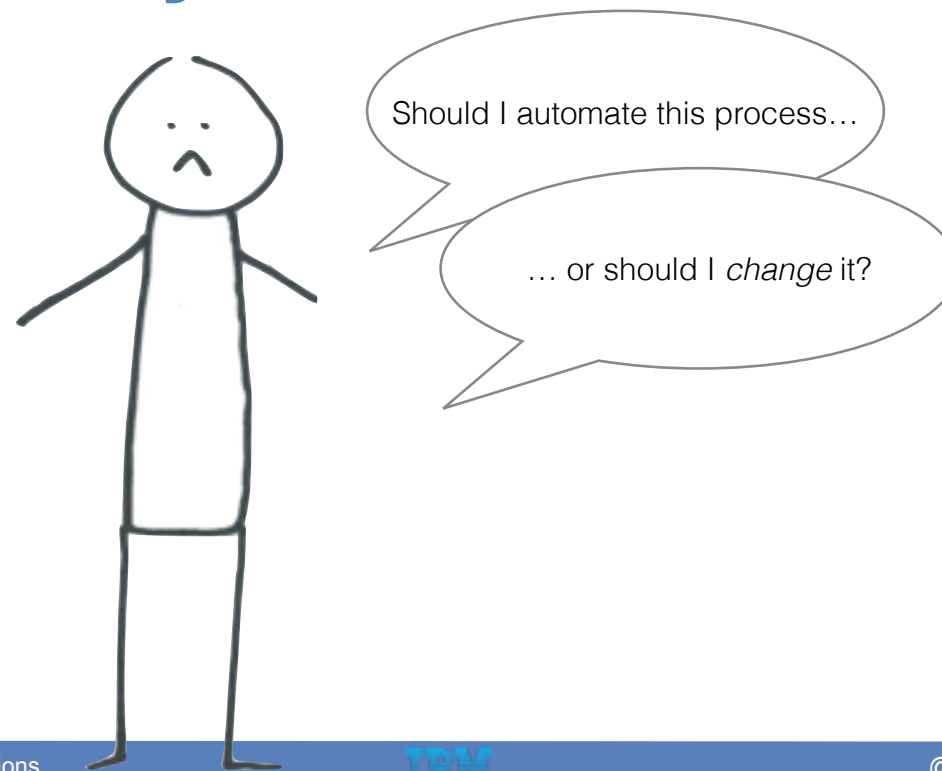


#automationconfessions



@holly_cummins

Why *not* automate



The dumbness problem

TODO can we come up with an example?

Is server setup vs EBC an example?

TODO, want to talk more about Process - Is it always bad?

A structure for how to interact at scale

(credit, JavaPosse somewhere)

The dumbness problem

- It's dumb to automate dumb things

TODO can we come up with an example?

Is server setup vs EBC an example?

TODO, want to talk more about Process - Is it always bad?

A structure for how to interact at scale

(credit, JavaPosse somewhere)

The dumbness problem

- It's dumb to automate dumb things
- Automation allows you to do dumb things at a greater speed

TODO can we come up with an example?

Is server setup vs EBC an example?

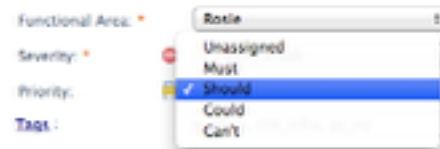
TODO, want to talk more about Process - Is it always bad?

A structure for how to interact at scale

(credit, JavaPosse somewhere)

example

Defect screening

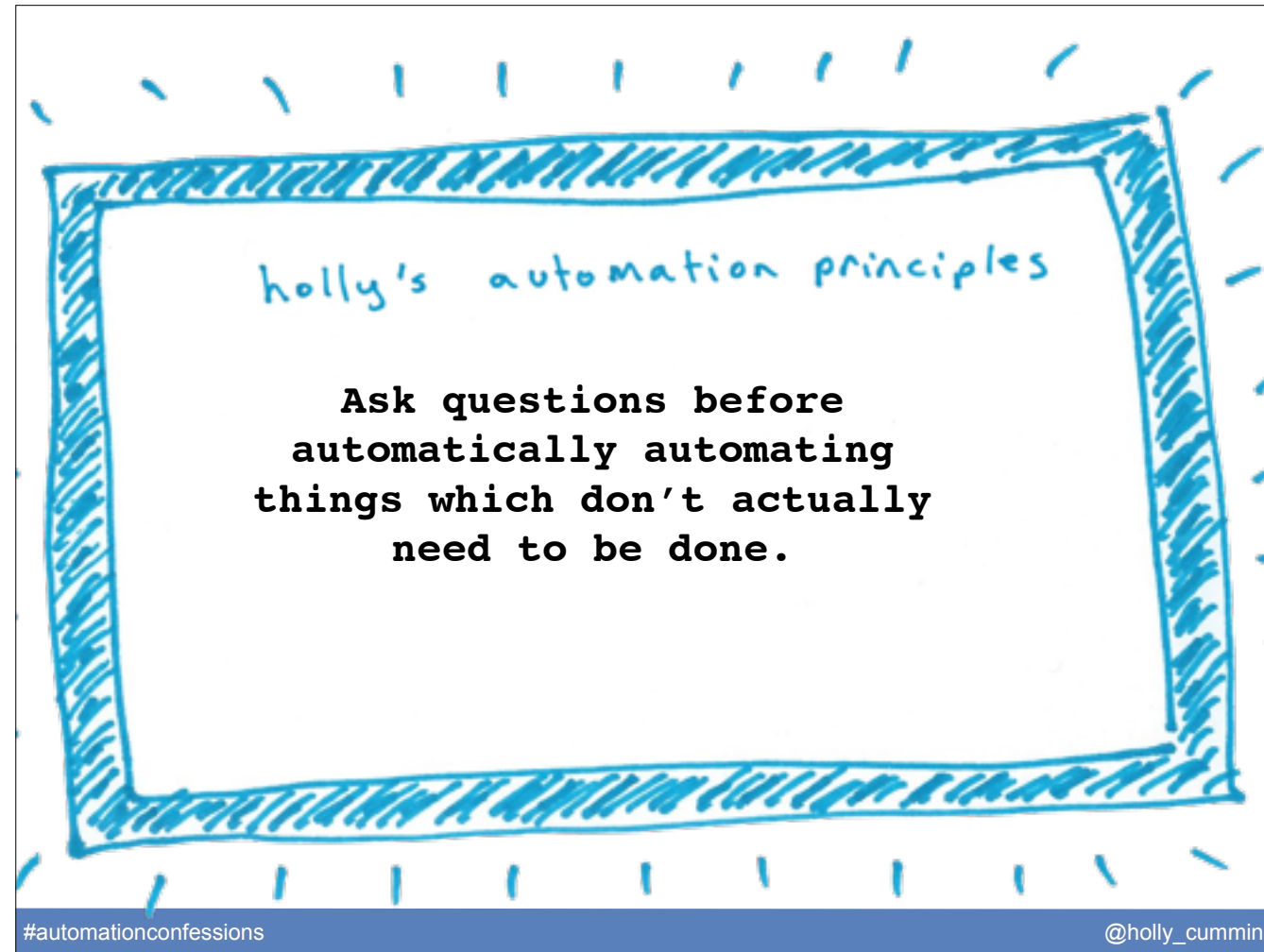


#automationconfessions

@holly_cummins

We screen defects daily to manually set a priority, and takes time, and isn't anyone's favourite activity.

Someone asked if we could automate it, and just automatically set them to should, which is what we usually end up settling on. Technically, we could, but is it the right thing to do? If the priority field doesn't have value, we shouldn't have it.



If a process is so painful the only way to cope with it is to automate it, maybe the process needs to go away.

The other way of phrasing this is "don't automate if you don't care"

Why *not* automate

Lovely video on youtube of robot trying to feed a mannequin but instead repeatedly punching it in the face.

Some jobs are too hard for computers, either because they don't have intuition, or because the rules are too complex to be able to turn into an algorithm.

Some things, like deployment, computers are way better at

Others, computers can be worse at

Why *not* automate



Lovely video on youtube of robot trying to feed a mannequin but instead repeatedly punching it in the face.

Some jobs are too hard for computers, either because they don't have intuition, or because the rules are too complex to be able to turn into an algorithm.

Some things, like deployment, computers are way better at

Others, computers can be worse at

Why *not* automate

Computers
don't always
get it right.



Lovely video on youtube of robot trying to feed a mannequin but instead repeatedly punching it in the face.

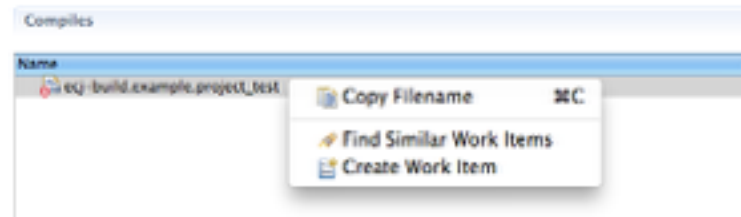
Some jobs are too hard for computers, either because they don't have intuition, or because the rules are too complex to be able to turn into an algorithm.

Some things, like deployment, computers are way better at

Others, computers can be worse at

example

Build monitoring



#automationconfessions

@holly_cummins

TODO example graphic

TODO screenshot of RTC build monitoring, lucene similarity search

What caused a failure?

If we could write a program to know, we could write a program that didn't fail

Need 100% accuracy

Need intuition

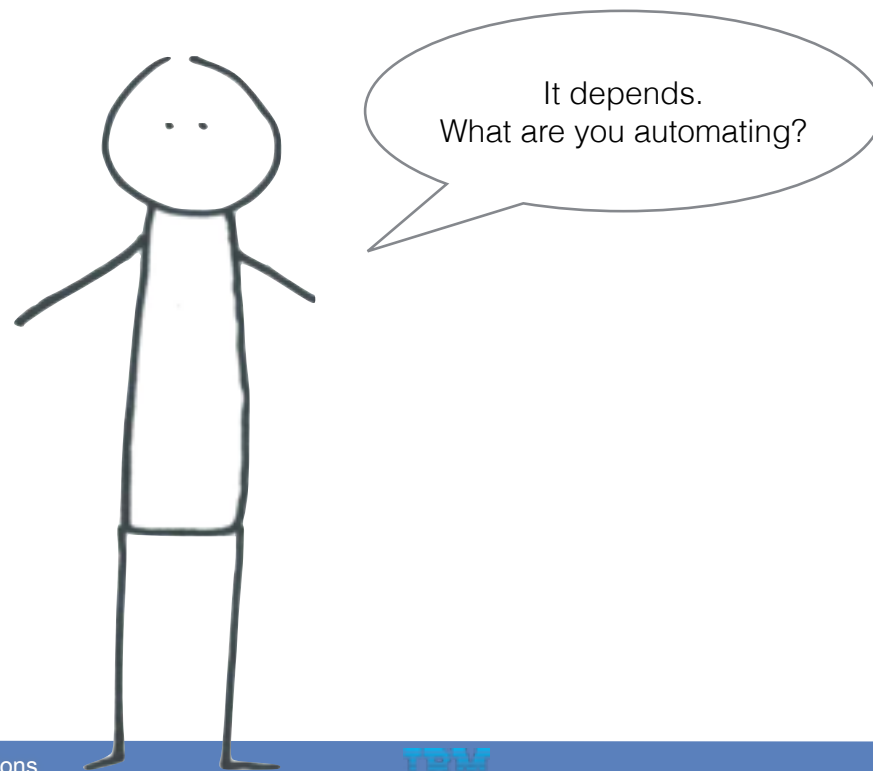
How to automate

#automationconfessions



@holly_cummins

How to automate



#automationconfessions



@holly_cummins

What can we automate?

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

What can we automate?

- Machine configuration

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

What can we automate?

- Machine configuration
- Deployments

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

What can we automate?

- Machine configuration
- Deployments
- Test environments

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

What can we automate?

- Machine configuration
- Deployments
- Test environments
-

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

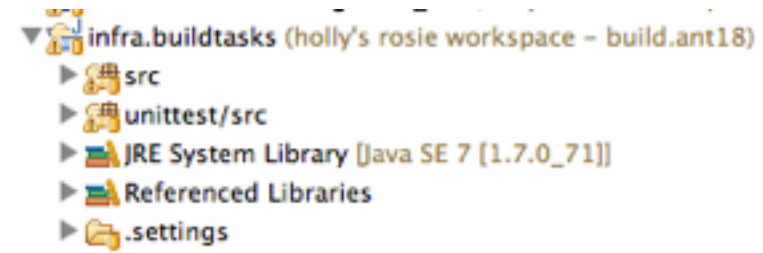
What can we automate?

```
..@Test{  
..public void testField(){  
..{  
.....String value = randomString();  
.....Thing object = new Thing();  
.....object.setField(value);  
.....assertEquals("The getter and setter are busted", value, object.getField());  
..}  
..}
```

IANAL, but need to care, which means a computer can help with 'did you forget this?' but the actual maintenance of the statements should be done by a person

Unit test generation - automatically drive getters and setters

What can we automate?

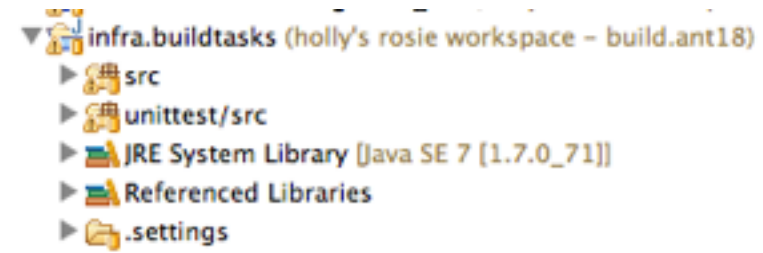


Project settings

Code formatting

...

What can we automate?

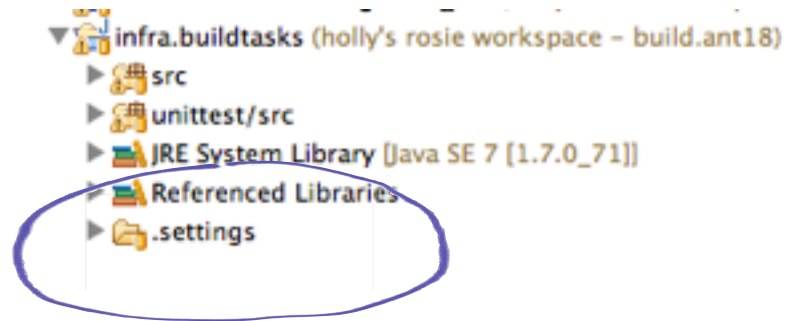


Project settings

Code formatting

...

What can we automate?

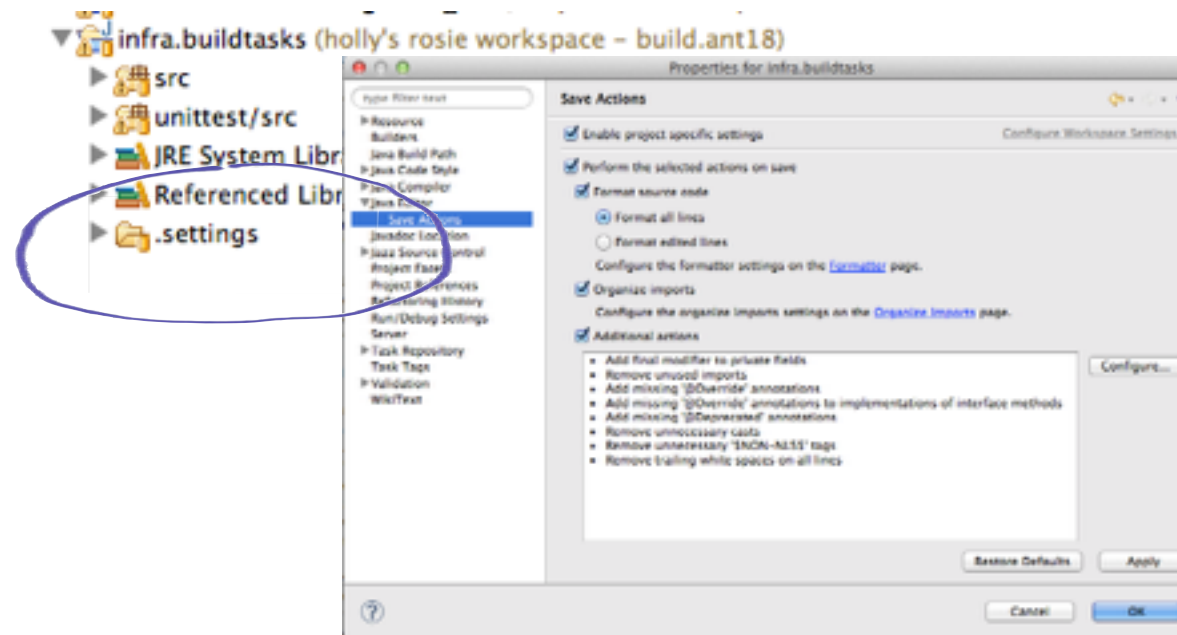


Project settings

Code formatting

...

What can we automate?

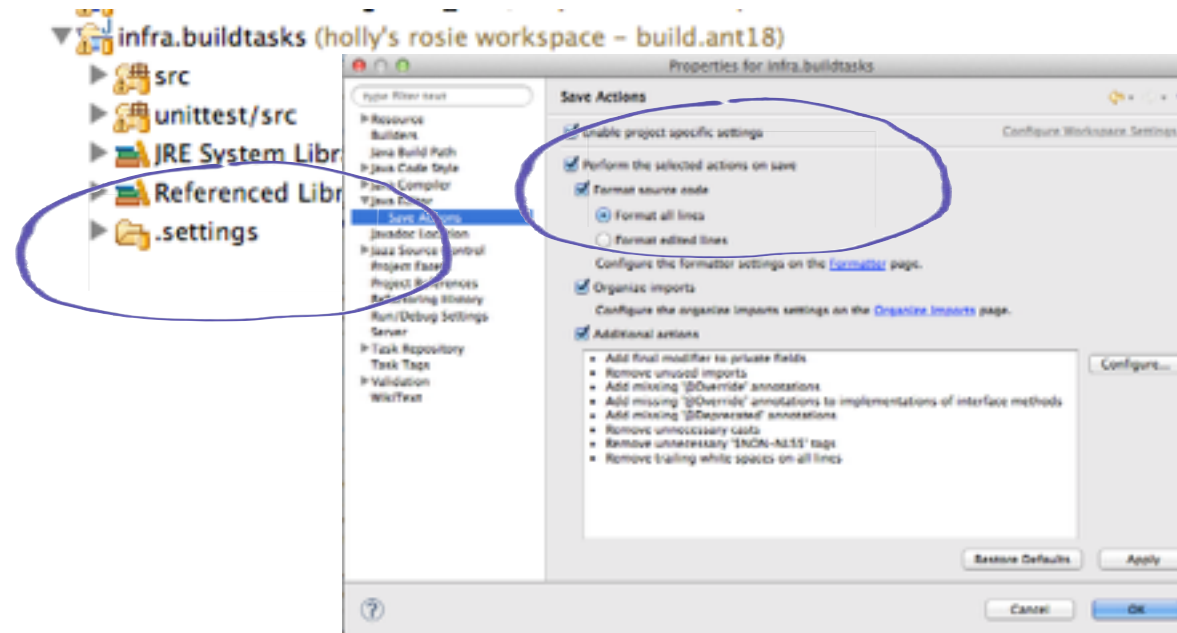


Project settings

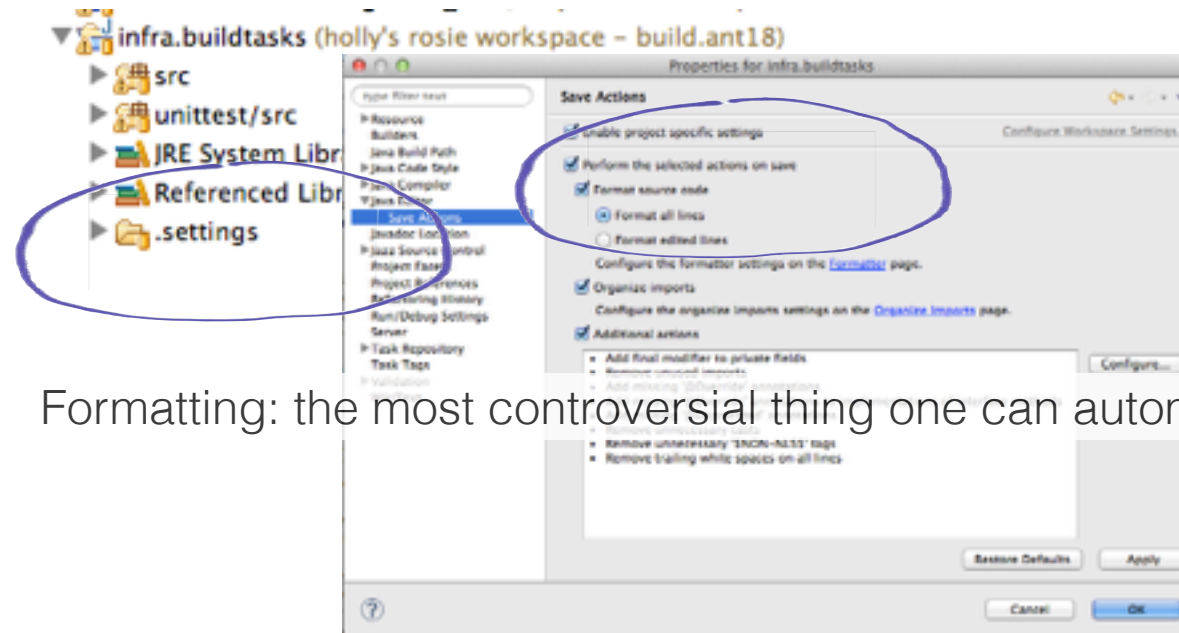
Code formatting

...

What can we automate?



What can we automate?



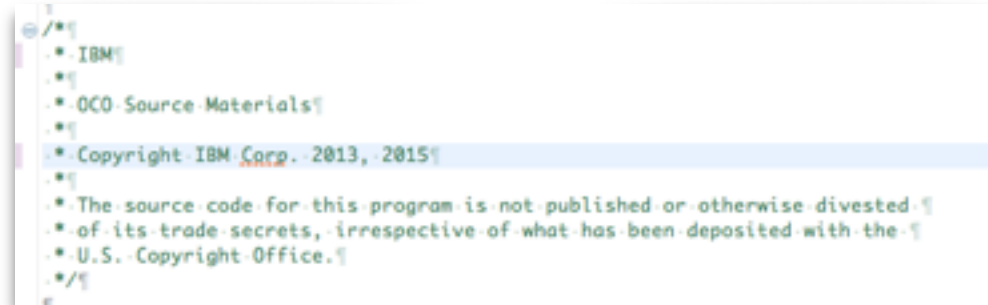
Formatting: the most controversial thing one can automate.

Project settings

Code formatting

...

What can we automate?

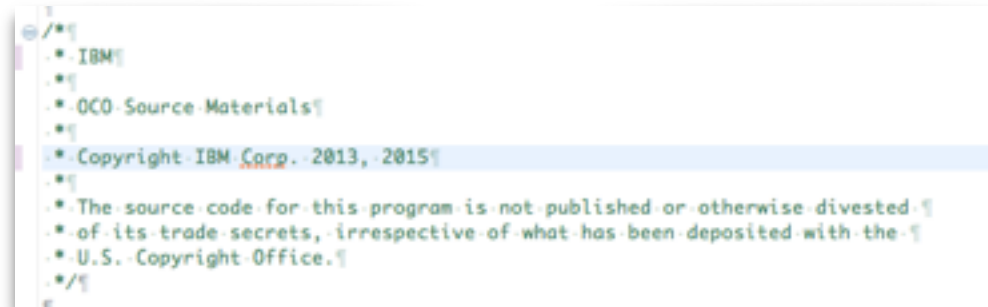


```
1  /*  
2  . * IBM  
3  . *  
4  . * OCO Source Materials  
5  . *  
6  . * Copyright IBM Corp. 2013, 2015  
7  . *  
8  . * The source code for this program is not published or otherwise divested  
9  . * of its trade secrets, irrespective of what has been deposited with the  
10 . * U.S. Copyright Office.  
11 . */
```

Actually kind of complex, legally.

Need to care, need to identify significant changes

What can we automate?

A screenshot of a code editor window showing a multi-line comment block. The comment is enclosed in /* and */. The text inside the comment includes: "IBM", "OC0 Source Materials", "Copyright IBM Corp. 2013, 2015", and a disclaimer: "The source code for this program is not published or otherwise divested of its trade secrets, irrespective of what has been deposited with the U.S. Copyright Office." The line "Copyright IBM Corp. 2013, 2015" is highlighted with a blue background.

```
/*  
 * IBM  
 *  
 * OC0 Source Materials  
 *  
 * Copyright IBM Corp. 2013, 2015  
 *  
 * The source code for this program is not published or otherwise divested  
 * of its trade secrets, irrespective of what has been deposited with the  
 * U.S. Copyright Office.  
 */
```

Copyright statements

Actually kind of complex, legally.

Need to care, need to identify significant changes

What can we automate?

```
1  /*  
2  * IBM  
3  *  
4  * OCO Source Materials  
5  *  
6  * Copyright IBM Corp. 2013, 2015  
7  *  
8  * The source code for this program is not published or otherwise divested  
9  * of its trade secrets, irrespective of what has been deposited with the  
10 * U.S. Copyright Office.  
11 */
```

Maybe.

Copyright statements

Actually kind of complex, legally.

Need to care, need to identify significant changes



TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

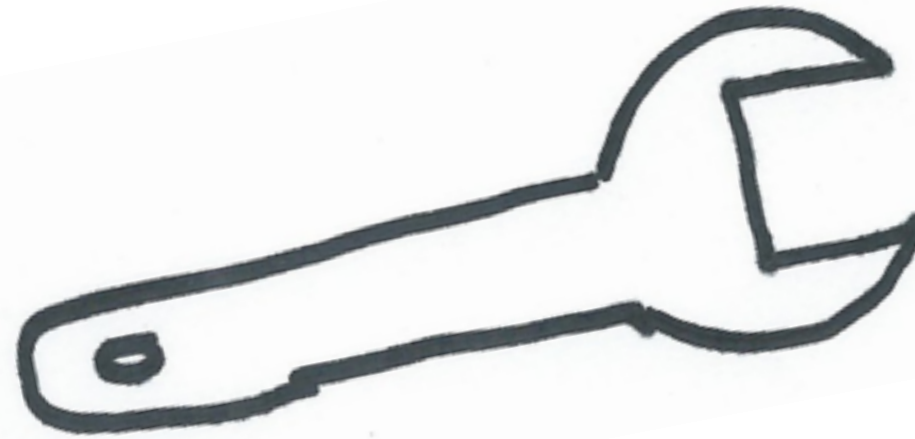
I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(an incomplete list)



#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(an incomplete list)



#automationconfessions



@holly_cummins

TODO shell picture

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Tools

(an incomplete list)

- Shell scripts



TODO shell picture

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Tools

(an incomplete list)

- Shell scripts
- sed



TODO shell picture

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Tools

(still an incomplete list)

- Scripting
 - python
 - ruby



Tools

(still an incomplete list)

#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

Jenkins or Rational Team Concert
builds

#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)



Chef or
Puppet

#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)



#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)



#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)



Docker, VMWare, and Vagrant images

#automationconfessions



@holly_cummins

TODO picture of wrench?

Kind of hard and stupid to even write a list because automation is such a general term, covers things from deployment to server management to source control to design documents.

I had to write a wrapper - automation, again - around sed to allow me to get the hang of it

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

example
It doesn't have to be fancy

#automationconfessions

@holly_cummins

Our infrastructure team provide 24/7 support for the critical internal services. They have a rota of who gets the phone.

If the phone-holder gets a callout, they have to be ready. However, in most cases they won't get a callout so don't want to invest in setting up an environment.

Solution:

The DOI 24/7 laptop, set up with the tools the support team need. Easier than a vmware image. Doesn't scale, but that's ok.

example
It doesn't have to be fancy



#automationconfessions

@holly_cummins

Our infrastructure team provide 24/7 support for the critical internal services. They have a rota of who gets the phone.

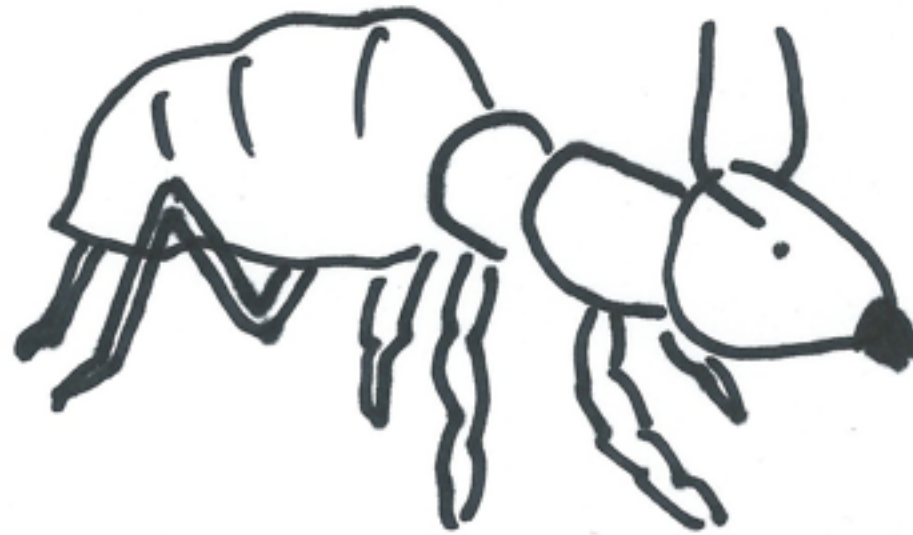
If the phone-holder gets a callout, they have to be ready. However, in most cases they won't get a callout so don't want to invest in setting up an environment.

Solution:

The DOI 24/7 laptop, set up with the tools the support team need. Easier than a vmware image. Doesn't scale, but that's ok.

Tools

(still an incomplete list)



#automationconfessions



@holly_cummins

Speaking in shoreditch and mention ant, uncool

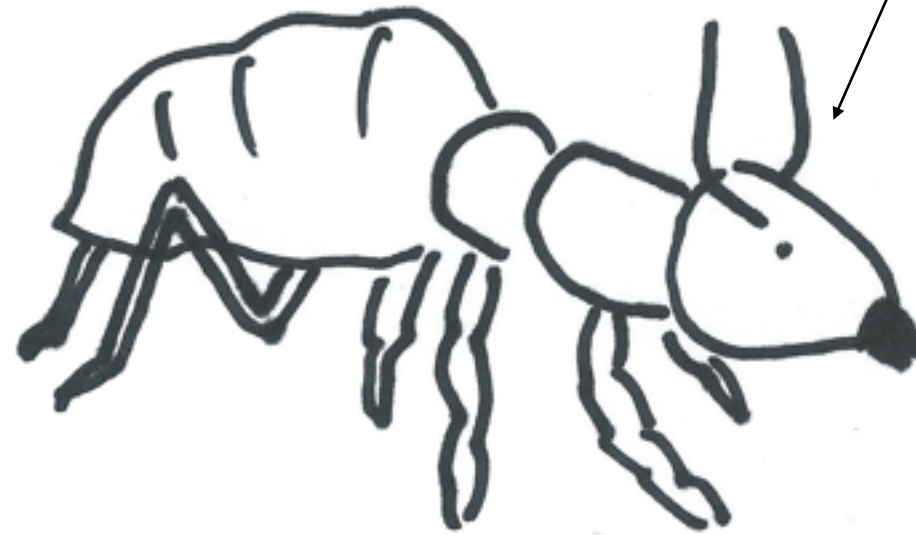
Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

An ant.



#automationconfessions



@holly_cummins

Speaking in shoreditch and mention ant, uncool

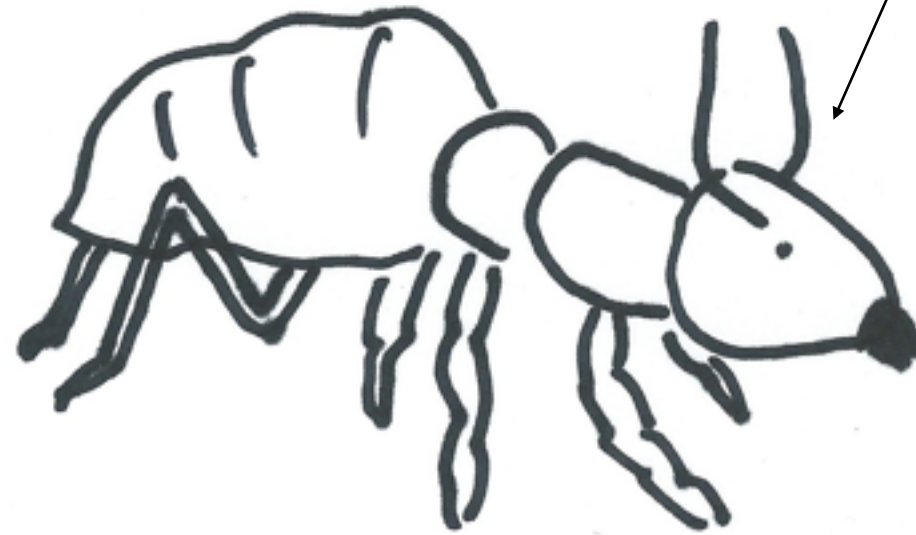
Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

An ant.
Not a reindeer.



#automationconfessions



@holly_cummins

Speaking in shoreditch and mention ant, uncool

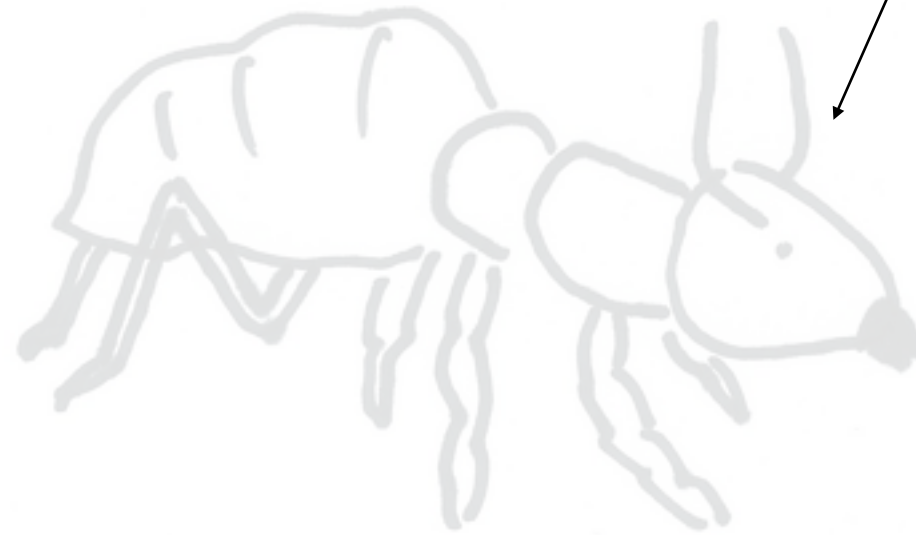
Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

An ant.
Not a reindeer.



#automationconfessions



@holly_cummins

Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

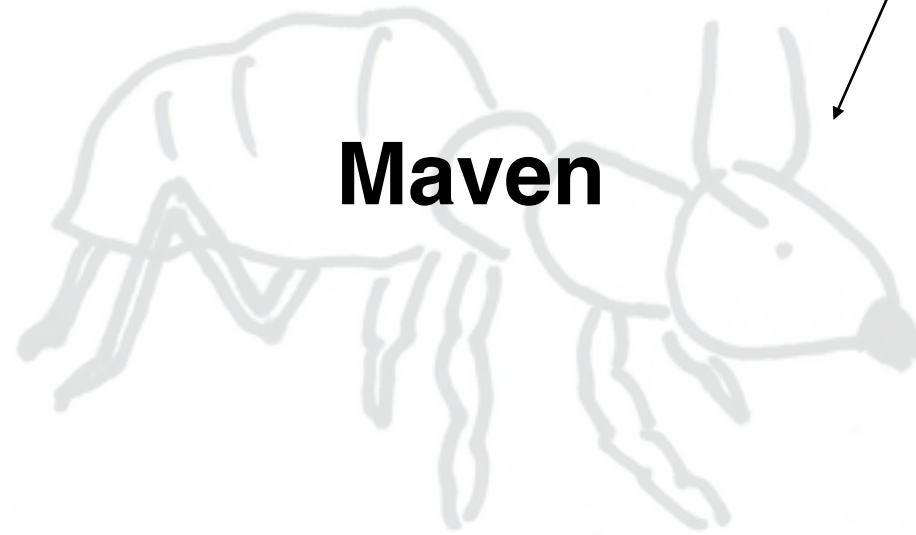
In

Tools

(still an incomplete list)

An ant.
Not a reindeer.

Maven



#automationconfessions

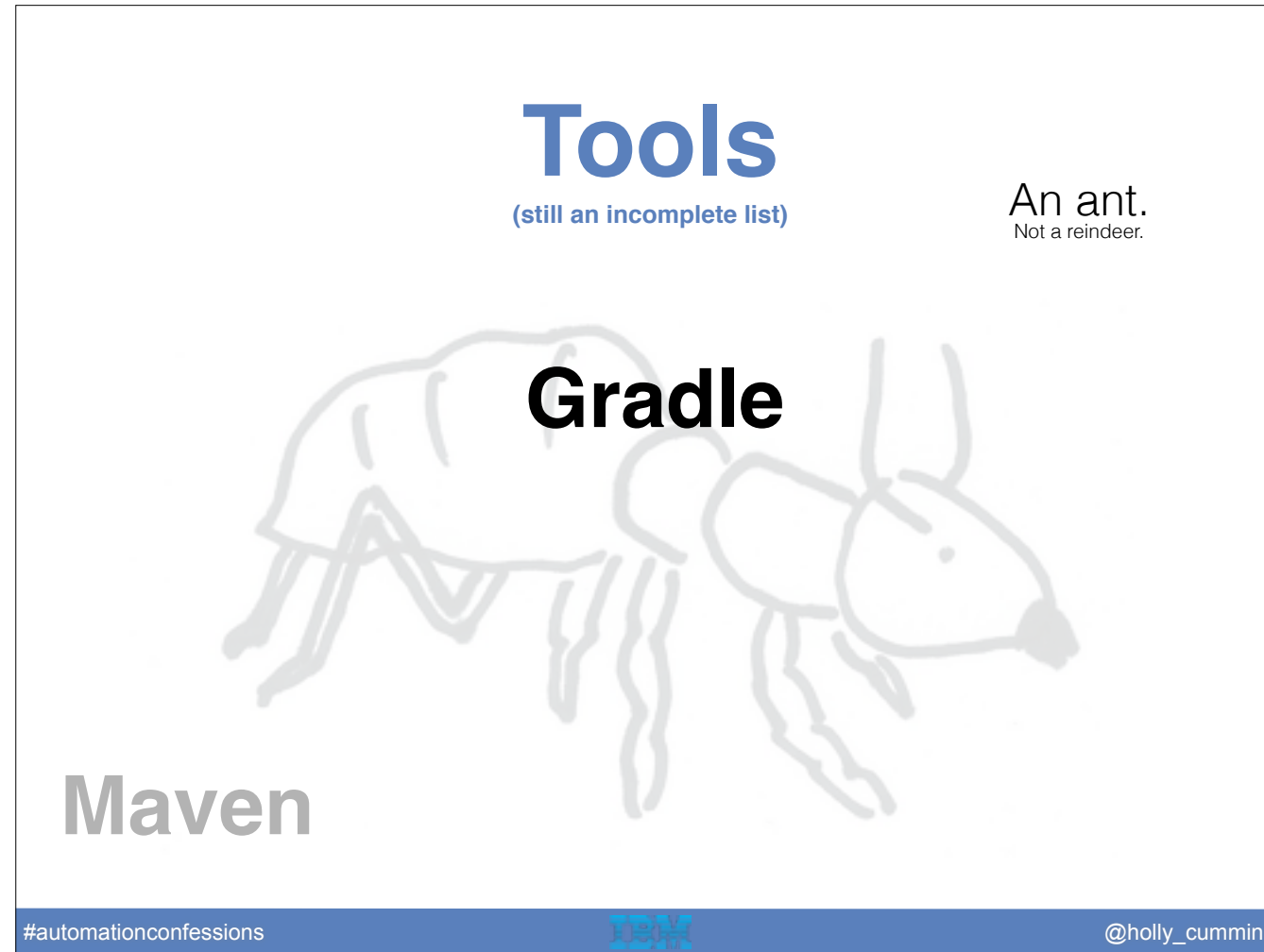


@holly_cummins

Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In



Speaking in shoreditch and mention ant, uncool

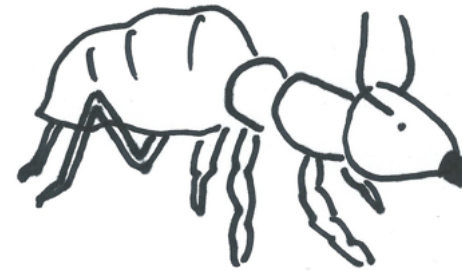
Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)

An ant.
Not a reindeer.



Gradle Maven

#automationconfessions



@holly_cummins

Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

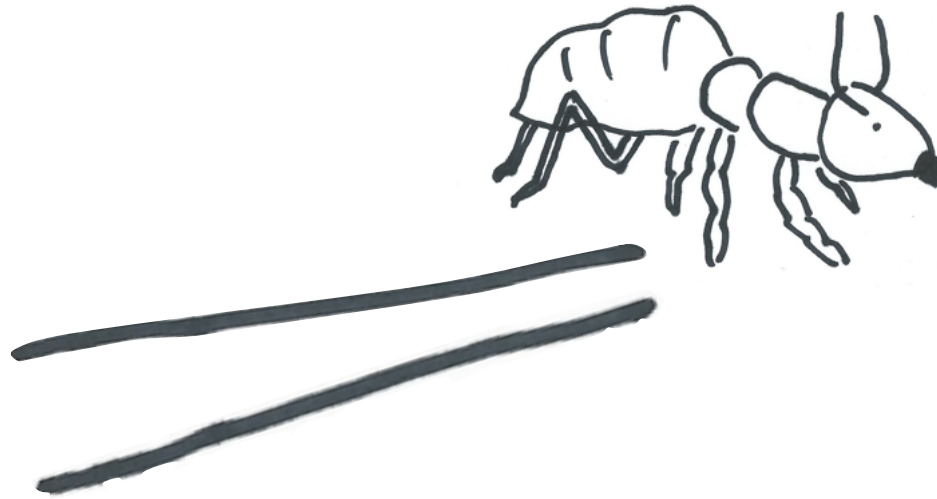
In

Tools

(still an incomplete list)

An ant.
Not a reindeer.

Gradle
Maven



#automationconfessions

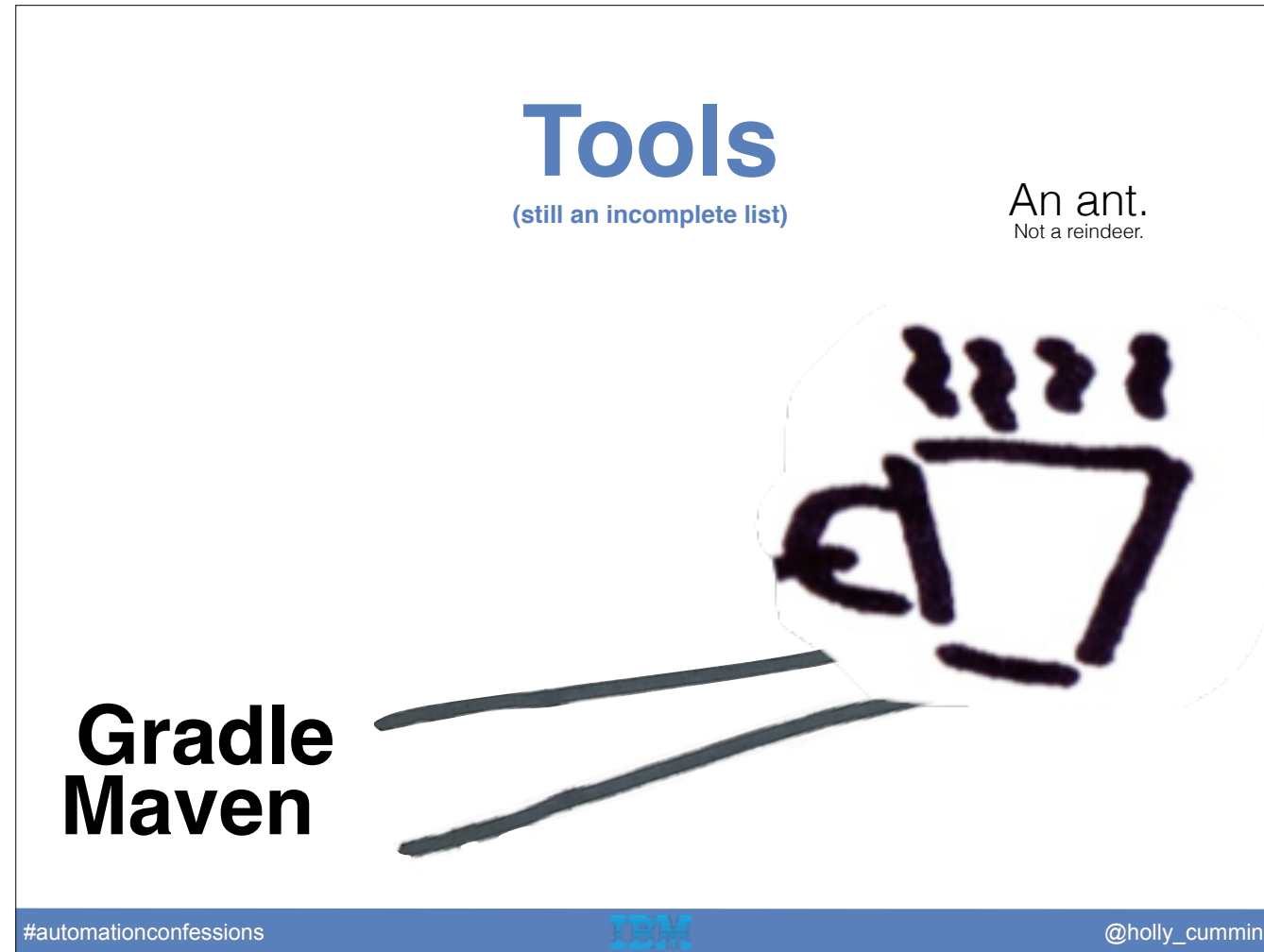


@holly_cummins

Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

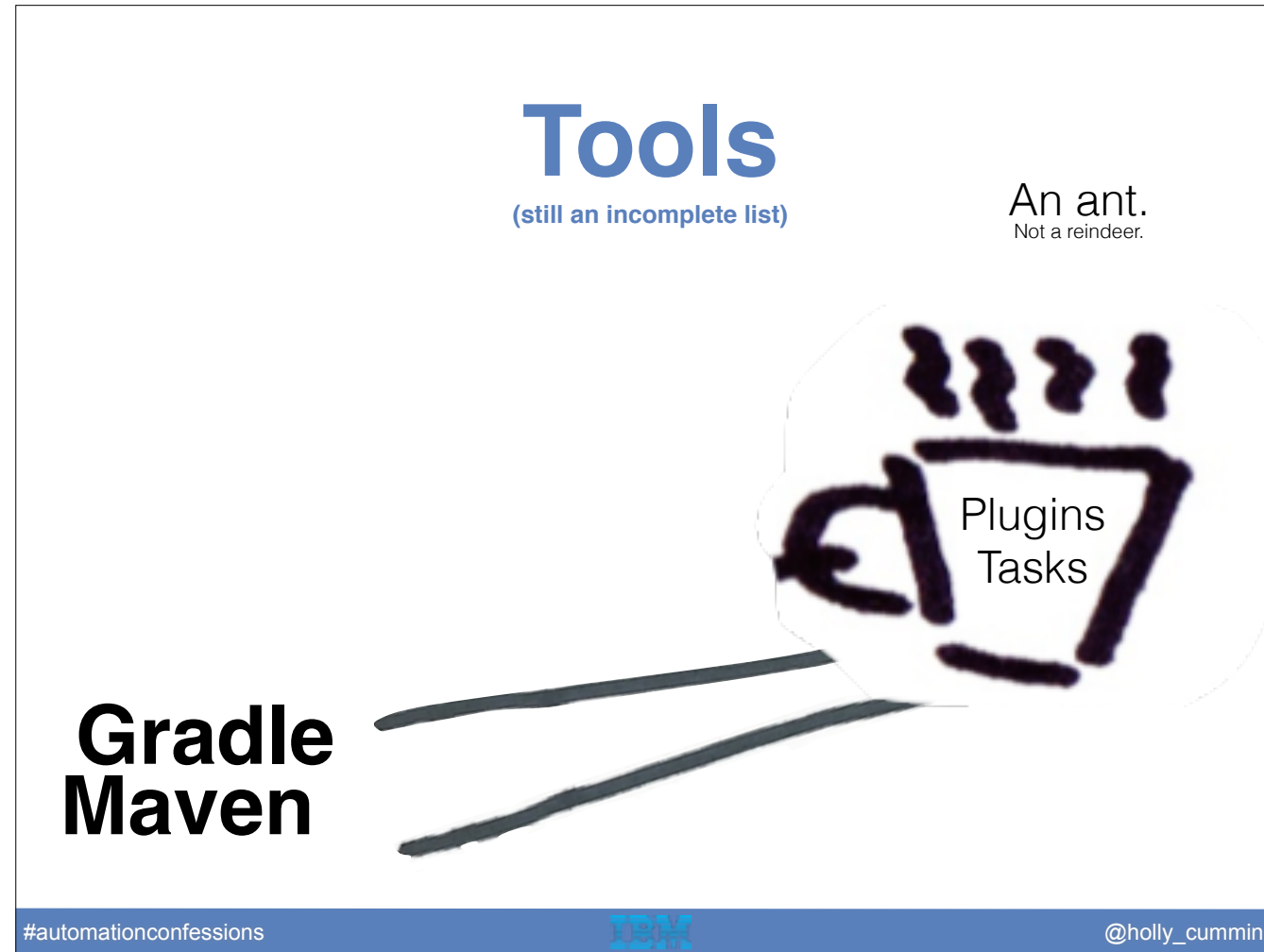
In



Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In



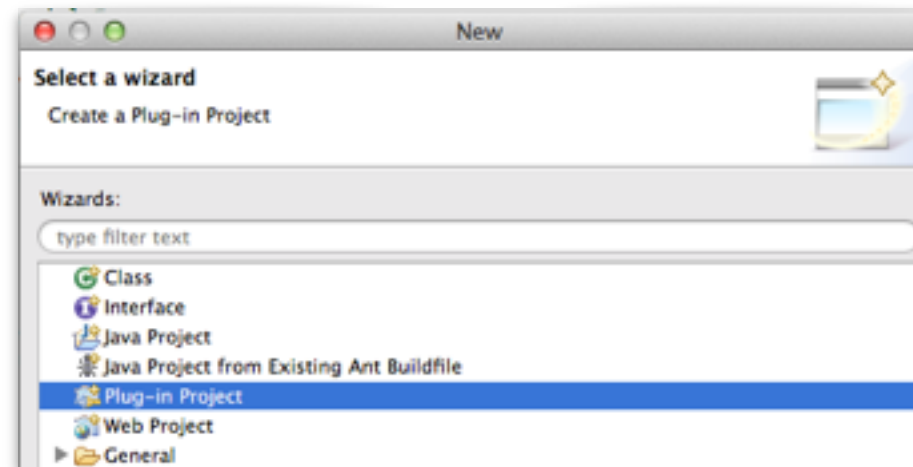
Speaking in shoreditch and mention ant, uncool

Maven can of course be used for automation, but I haven't included it in the list since often in Maven you'll shell out to ant for any kind of procedural process, unless you write your goal

In

Tools

(still an incomplete list)



#automationconfessions



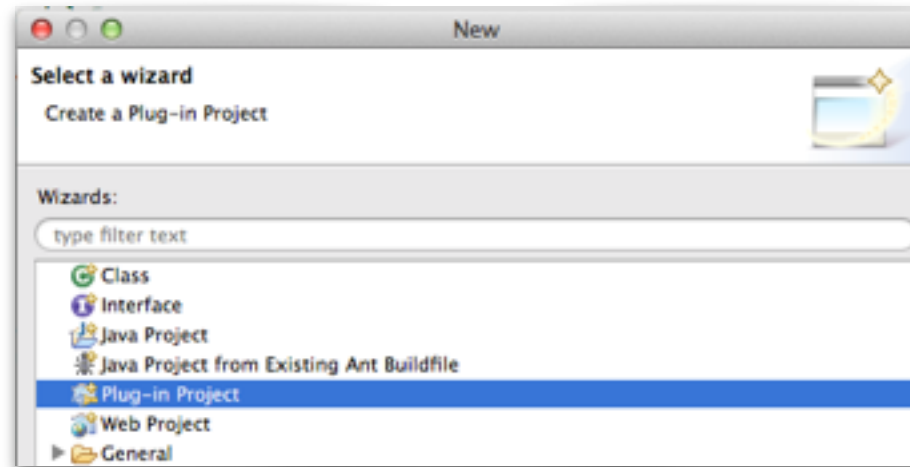
@holly_cummins

maintenance a problem

Tools

(still an incomplete list)

- Eclipse plugins



#automationconfessions



@holly_cummins

maintenance a problem

Tools

(still an incomplete list)



#automationconfessions



@holly_cummins

Web apps are interactive
can do more, because user makes choices
need to interact with user
need a better class of user!

Tools

(still an incomplete list)



- Web applications

#automationconfessions



@holly_cummins

Web apps are interactive
can do more, because user makes choices
need to interact with user
need a better class of user!

Tools

(still an incomplete list)



- Web applications
- If a UI is involved, proceed with caution

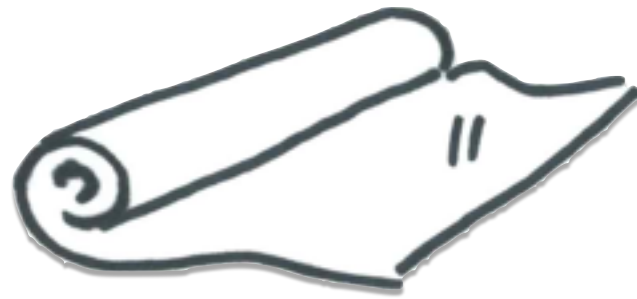
#automationconfessions



@holly_cummins

Web apps are interactive
can do more, because user makes choices
need to interact with user
need a better class of user!

Rolling your own



#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

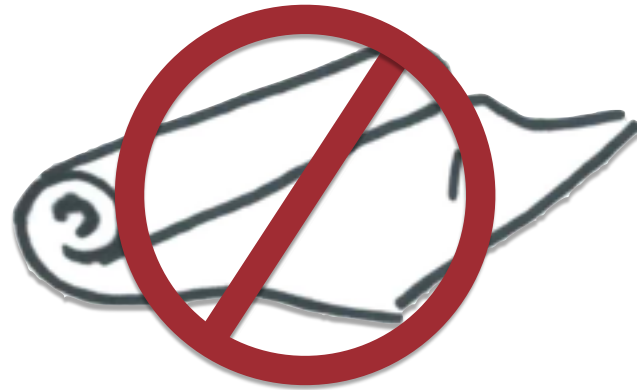
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

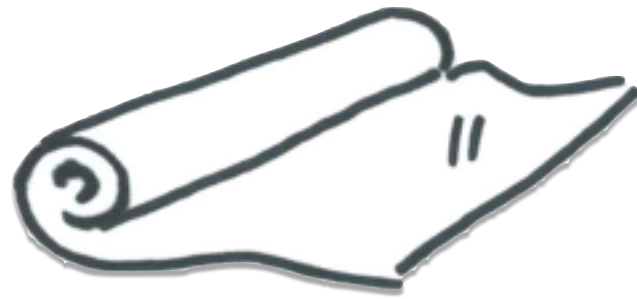
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

Should you roll your own, or is there something already out there?

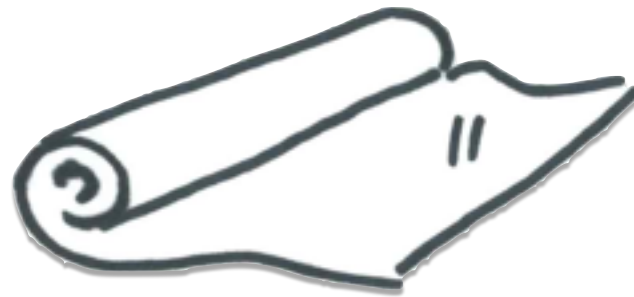
If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own

- Beware not-invented-here



#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

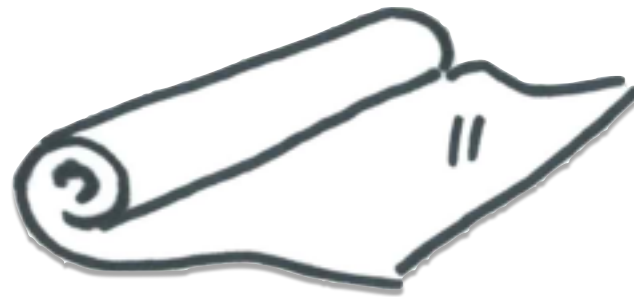
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



- Beware not-invented-here
- Are you *sure* no one else has done this?

#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

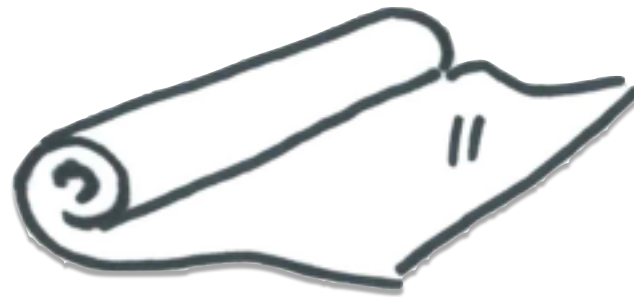
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



- Beware not-invented-here
 - Are you *sure* no one else has done this?
 - If not, maybe there's a good reason?

#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

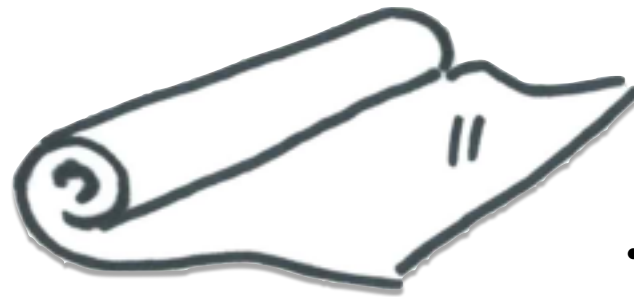
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



- Beware not-invented-here
 - Are you *sure* no one else has done this?
 - If not, maybe there's a good reason?
- Beware *fun*

#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

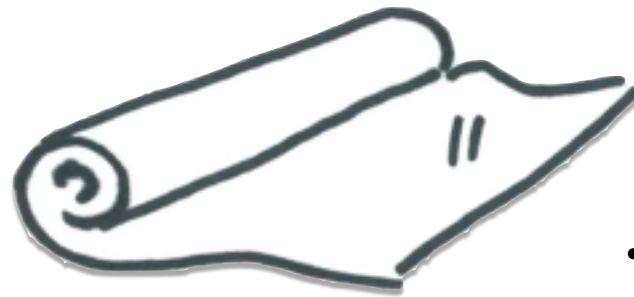
Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Rolling your own



- Beware not-invented-here
 - Are you *sure* no one else has done this?
 - If not, maybe there's a good reason?
- Beware *fun*
- But do innovate (and have fun)

#automationconfessions



@holly_cummins

What's this? [click]

Roll your own.

This is really important, too. There's no right answer.

Should you roll your own, or is there something already out there?

If not, why not? (maybe you have a bad process, or a weird one?)

In general, be very wary of rolling your own. Beware of not invented here, getting seduced by interesting problems (like me and the design document generator)

On the other hand, what does roll your own even mean? Automation often involves some measure of inventing something, but probably starting with the tools I listed on the previous page

Finally ...



#automationconfessions



@holly_cummins

I do love automation, I can't resist it. Sometimes it gets me in trouble, but I still get to do it. And that's really cool.

Finally ...

Aren't we lucky to have careers where automation is not just allowed, but *encouraged*?

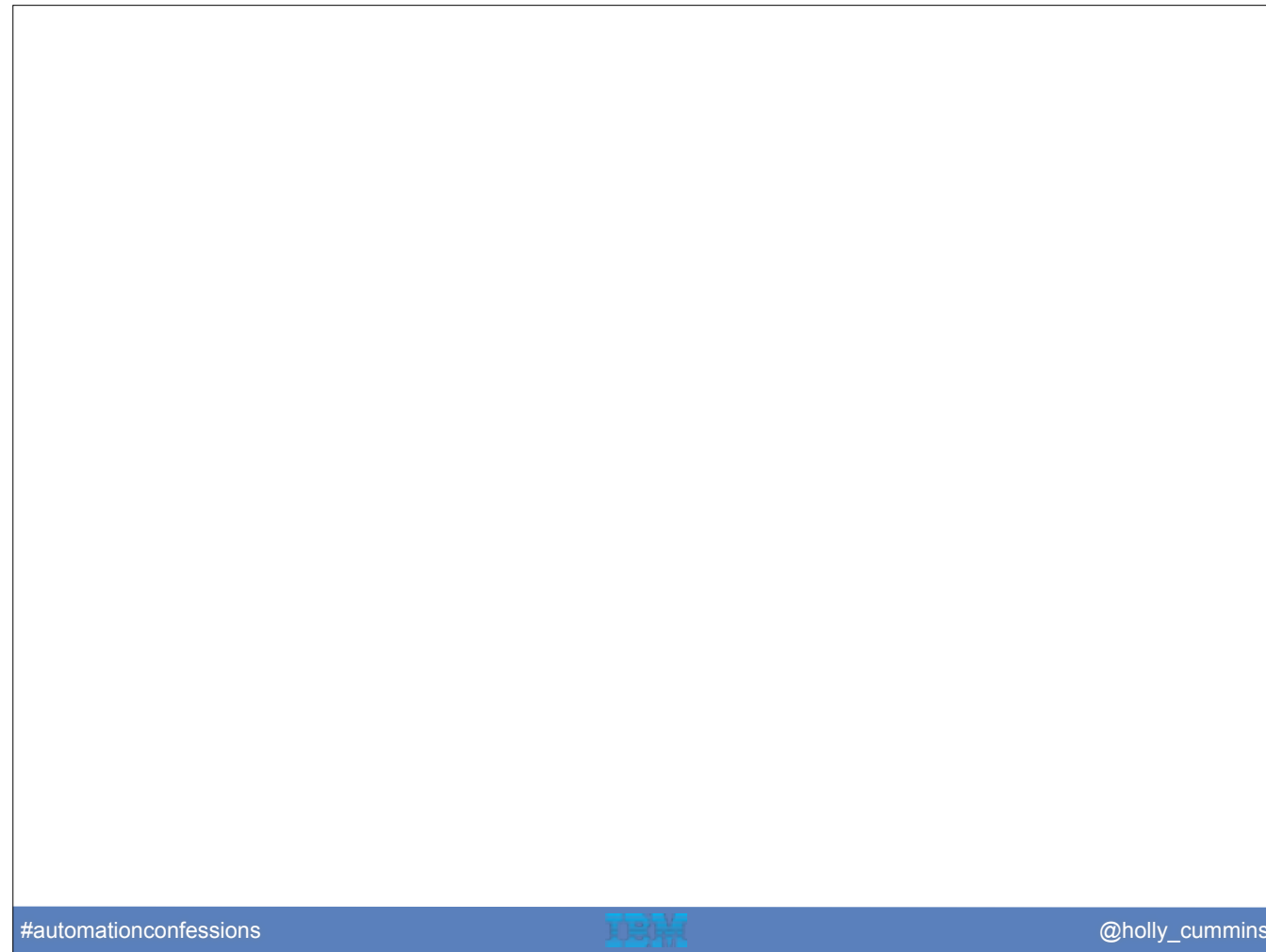


#automationconfessions



@holly_cummins

I do love automation, I can't resist it. Sometimes it gets me in trouble, but I still get to do it. And that's really cool.



I do love automation, I can't resist it. Sometimes it gets me in trouble, but I still get to do it. And that's really cool.

Any Questions?

@holly_cummins
www.wasdev.net

#automationconfessions



@holly_cummins

I do love automation, I can't resist it. Sometimes it gets me in trouble, but I still get to do it. And that's really cool.