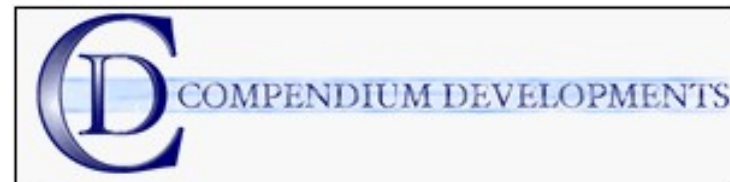


“It Depends...”

Context Driven Testing -
What does it mean to have no
“*best practices*”?

A personal view



Alan Richardson

Compendium Developments

<http://compendiumdev.co.uk/context>

Version 1.20050305NorthLondonBCS



I was a teenage methodology monster!

Gasp in amazement

As you see him know exactly what do do, *before he is even on site.*



You will Be shocked

As the rework mounts to truly heroic proportions

Can you bear to see

Him justify his actions by quoting text book after text book and expert after expert.

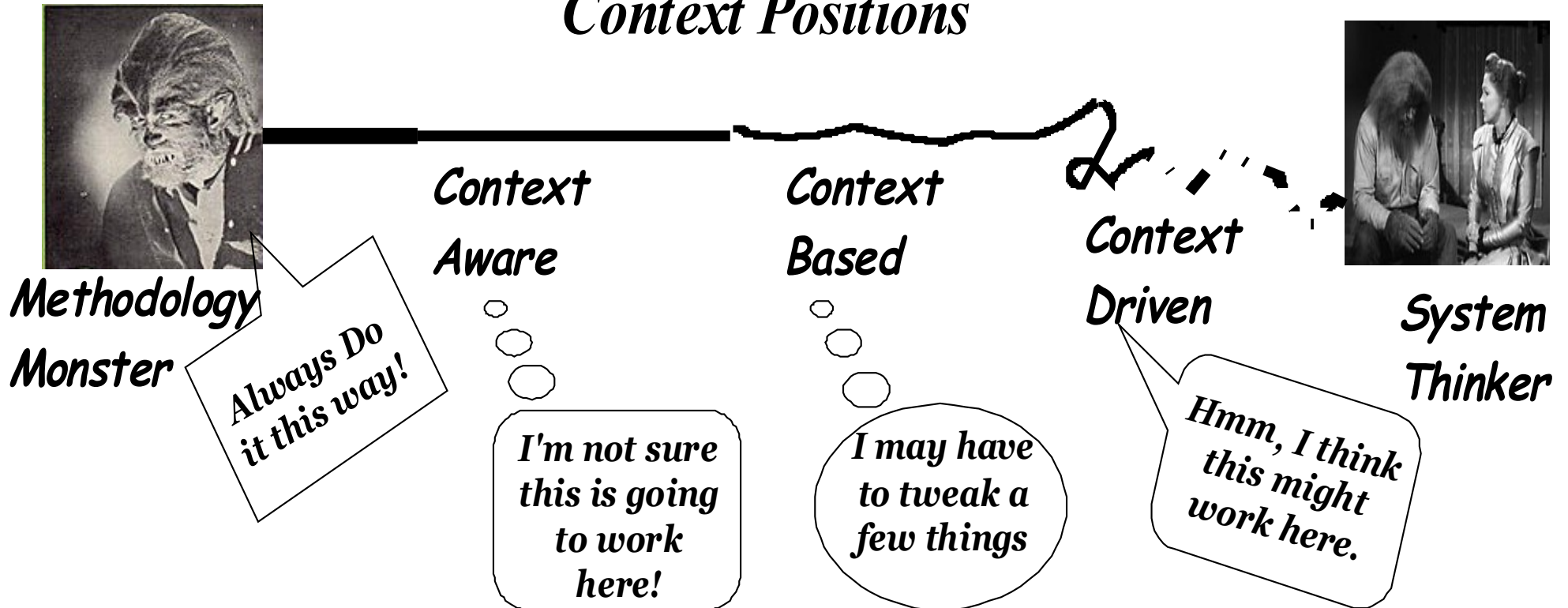
He can do no wrong.

Coming soon to a project near you.



Introduction

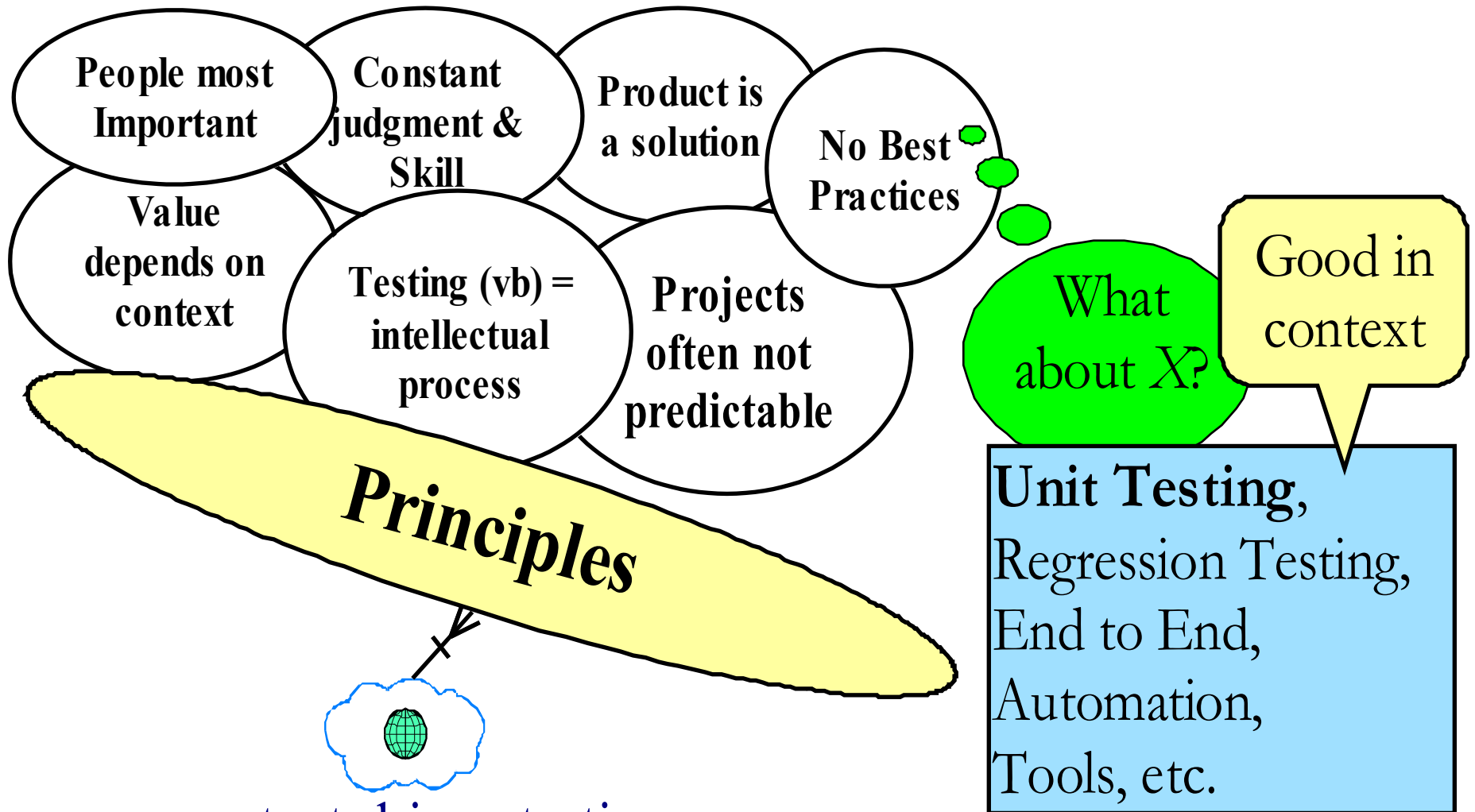
Context Positions



I ceased my evil ways

- became more context aware
- asked more questions – different types of questions (how? What?)
- worked more closely with people – retaining independence of thought
- became more of a consultant than a salesman/pusher
- became more effective as a tester
- Less identification: “I am a structured tester”

What is context driven testing



www.context-driven-testing.com

Context-Driven-Testing.com

- The value of any practice depends on its context.
- **There are good practices in context, but there are no best practices.**
- People, working together, are the most important part of any project's context.
- Projects unfold over time in ways that are often not predictable.
- The product is a solution. If the problem isn't solved, the product doesn't work.
- Good software testing is a challenging intellectual process.
- Only through judgment and skill, exercised cooperatively throughout the entire project, are we able to do the right things at the right times to effectively test our products.

From www.context-driven-testing.com

Isn't Unit Testing a *best* practice?

- Unit Testing =
 - *Any* testing done by the *developer*
 - Any testing done *under the debugger*
 - *Only* testing which is *automated* at the code level
 - *<insert your definition here>*
- That isn't well defined enough to be a *best* practice

What about X ?

- Regression Testing
- End to End testing
- Full Code Coverage
- Automated Testing
- Fully Scripted Testing
- Automated Test Data Generation
- Non-Functional Testing
- V-Model, etc.

- X is Not defined here
- X *might* be good
- X is not a guarantee of *bestness*

What is the
reasoning
behind wanting
this practice?

X is a best practice for us

- Oh *really*. Honestly...
 - What extra baggage has *X* given you?
 - When did you last wish that you didn't do *X*?
 - Did you ever think...*X* is a little out of place here?
 - When did you last think of an alternative to *X*?
 - How did you decide that?

It might well be a good practice here. Let me re-check

When best practices went bad!

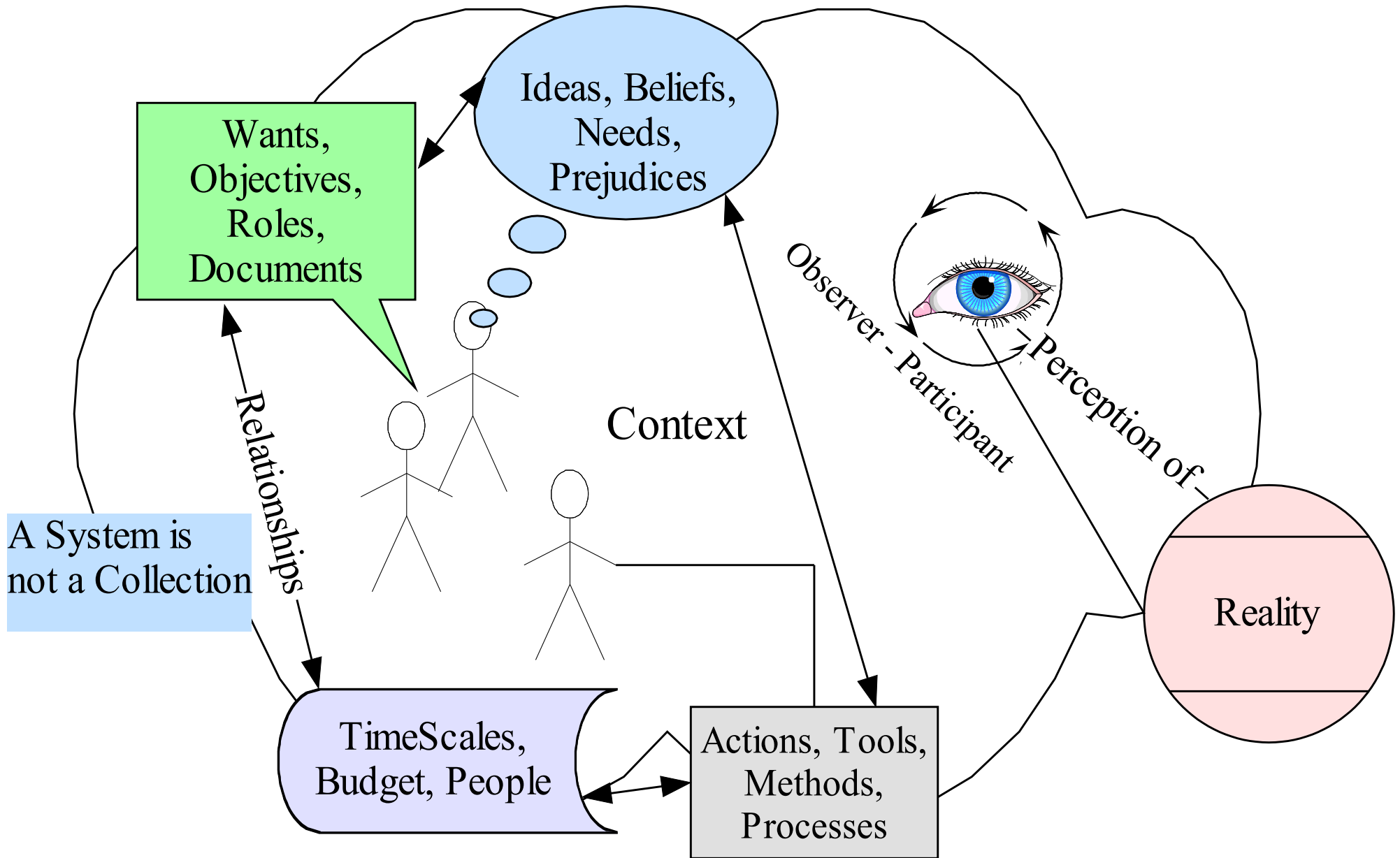
- **It worked for him, let's have everybody do it!**
- **Thou Shalt, regardless!**
 - Hey, how come the developers get off so lightly?
(they have a job to do)
- **You say you're doing this, I know you're not SO...**
 - V-model...add lots of xrefs to enable impact analysis

Commonalities: Ignored Context

What is not context driven?

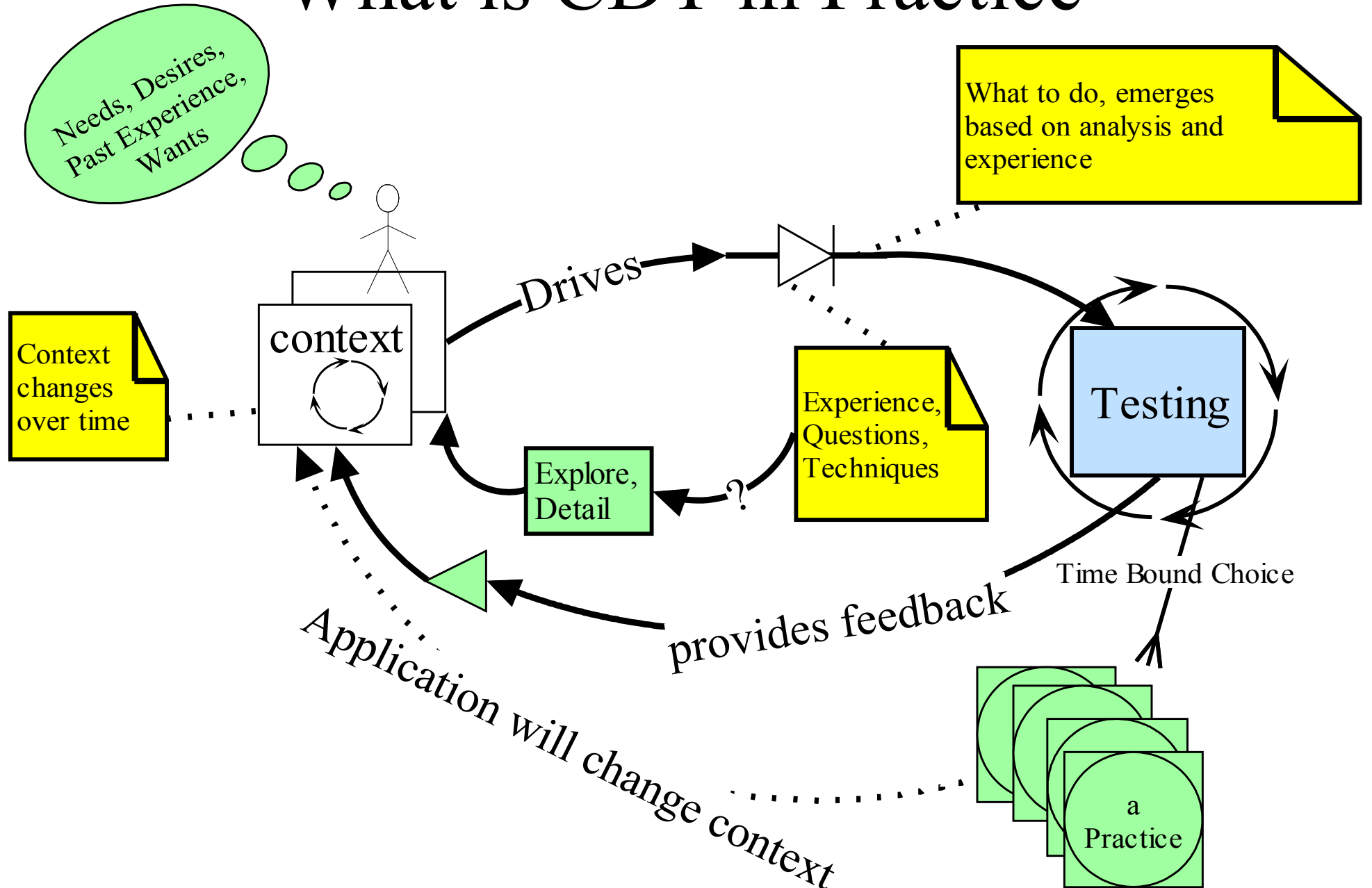
- Not context *aware*
 - *I know you're not doing it 'right', but this is what I always do, so I'll work harder to make what I'm doing work.*
- Not context *based*
 - *I'll introduce the V-model and 'tweak' it a little*
 - *We can just 'amend' our methodology a bit and we will be fine*
- Not an excuse for the slipshod

What is Context?



A System is not a Collection

What is CDT in Practice



What is context *driven*?

- Context *Driven*
 - “*What to do?*” *emerges* based on analysis and *experience*
- Bring lots of techniques and practices and *choose* which to apply.
- *Know* that the application of techniques & practices *changes the context*

Remind me, What is *Testing*?

- Testing doesn't define the context
- Testing fits in the context
- Testing provides information
- Testing provides feedback
- One size fits all \neq best practice
- Standard \neq best practice

How is it different

What can you expect from context-driven testers?





- “It depends...” on the context, but generally (*I suspect they will*)...
 - Ask a lot of questions of you and of themselves
 - Know the underlying assumptions of their techniques
 - Change their minds & re-plan
 - Be able to justify what they are doing, in terms of the context as they perceive it

How are the Questions different?

- Testers always ask questions
- Context Driven Qs are aimed at exploring the context, not just working within it
- Questions that try to avoid assumptions
 - When will the requirements document be ready?
 - *becomes*
 - Are requirements being elicited and communicated?
How? Who from? How much change? etc.

How You Can Help

How can you help them?

- Tell them not just what you are doing, but what your aims are, what your needs are Dictate Context
 - Don't say “Do this & that and everything will be ok” 
 - Wants are not Needs. Want less, Need more  Explain
- Provide examples of general situations you face  Detail
- avoid “I'd like to help but...”  Explore
- Be truthful and trust

*From “Scoring a whole in one” by
Dr Edward Martin Baker*

What is the context?

- People, methods, tools, aims, objectives, dreams, ideas, prejudices, beliefs, actions, documents, roles, time scale, money, etc.
- A *system* of relationships between the above.
- A system is not a collection.
- Context can be hard to identify: observer-participant
- We never work with the context – we work with our perception of the context – expect change.

Identify your Context

- What did we say we were doing?
- What are we actually doing?
- Who for?
- Why? What will we gain by doing that?
- Remind me - what is our aim/objective?
- What could we do instead?
- What would happen if we did something else?

There is no such thing as Context

- Work through our filters
- Question our assumptions
- What is important?

How Context Driven is your....

- Testing?
- Development?
- Is there a mismatch?
- If there is, what does the mismatch lead to?

End Notes

Learning More About Context

- www.context-driven-testing.com
- Yahoo group “software-testing”
- I recommend:
 - General Semantics, NLP, Gerald Weinberg's books
 - Systems Thinking/Theory
- Others recommend: *(and I might too, I just haven't explored these enough yet)*
 - Critical Thinking, Epistemology
- Honestly evaluate your own context
-