

Test planning with morphology analysis

Christian Theune, ct@gocept.com
gocept gmbh & co. Kg

EuroPython 2006, Geneva, Switzerland

Warning

Experimental stuff

Warning

Just gathering initial experience ...

Warning

You might be more knowledgeable than me.

It's still pretty interesting, though.

Test planning

- We do (unit|functional|integration|*) tests.
- How do we plan them?
 - What to test?
 - What kind of tests?
 - How many tests?
 - „Enough“ !?!

Test planning

- Goal: Determine the **minimal amount of tests** while still providing **maximal coverage of functionality**
 - (That's an optimization issue!)
 - As few test fixtures as possible
 - Don't miss any „important“ (?) fixtures

Morphology Analysis

I stumbled over „Swedish Morphological Society“:
<http://www.swemorph.com/>

Morphology Analysis

Nice program that allows you to „play“ with solutions in a given problem space.

Fritz Zwicky



Fritz Zwicky (1898-1974)

- Professor of Astronomy (1942-1968) (Caltech)
- Research Director Aerojet Engineering and "father" of the modern jet engine
- President of International Academy of Astronautics
- First to discover evidence for dark matter in galaxies
- Formulated triple-hypothesis: supernovae, neutron stars & cosmic rays
- Galaxies and galaxy clusters act as gravitational lens
- Developed morphological analysis as a general method for non-quantified modeling

Morphology Analysis

- Goal: Find **all possible** solutions for a given problem
- Object: Morphological Field
 - allows n-dimensional modelling
- Well known derivation: The „Zwicky Box“

Morphological Field

- Describe a problem by it's influences
 - „Variables“
 - Input and Output!
- Describe all the different possible states of a variable
 - „Values“
- A „cross consistency“ matrix determines which values can be combined with other values
- A „laboratory“ allows you to play with possible solutions

Example morphological field

Sample from the swedish morphological society website, describing how to prepare the railroad staff for chemical accidents.

How to apply this for testing?

- Idea: a test performs a function with parameters
- A solution in the field is **one** test case
- Variables are formal parameters
- Values are actual parameters
- Cross consistency matrix to rule out combinations

Demo 1

- To test: dividing two numbers
- Variables:
 - Zähler
 - Nenner
- Exemplary Values:
 - +,-,0 for each

Demo 1

- 6 cases is not that much
- Anything to rule out?
- For the sake of demonstration: one test for „division by zero“ is enough

Demo 2

- Real field we used
- Pseudonymized customer data
- Task: A small rule engine that determines a gift product you receive when buying one or more products from this company.
- Remember to explain: trigger product, already registered product, chains, rules, valid dates, supplements, license

Result

- Two hours of work to fill in the real field
- Caught all errors that were influenced by the variables we chose
- Easy to reason about
- UI to do experiments with
- Cross consistency matrix shows patterns
- Customer liked the result
- Customer also was scared

Future

- Find and document patterns
- Create guidance documentation
- Finish and polish software
 - General Morphology library
 - Test plan development application / UI

Anti-Q&A

- Why „Morphology Analysis“, it doesn't have anything to do with a „form“!
 - Yes. I have know idea why it's called morphology analysis in respect to the general meaning of „morphology“.

World Cup update

BTW: I think Tarek was wrong, it's going to be:

| | |
|---------|---|
| Germany | 1 |
| France | 0 |

:)