MY TOPOS-CRUNCHING HACK PROJECT

STATE OF THE BEWL

WARNING

ABSTRACT MATH AHEAD

ABOUT PROJECT BEWL

- Bewl lets you define ‘systems of set-like objects’
- You can then talk fluently about the objects in those systems as if they were sets
- This gives you new, unexplored languages for talking about graphs, diagrams, permutations, musical compositions, etc
- In which you can hopefully do amazing things

A DSL FOR TOPOS THEORY

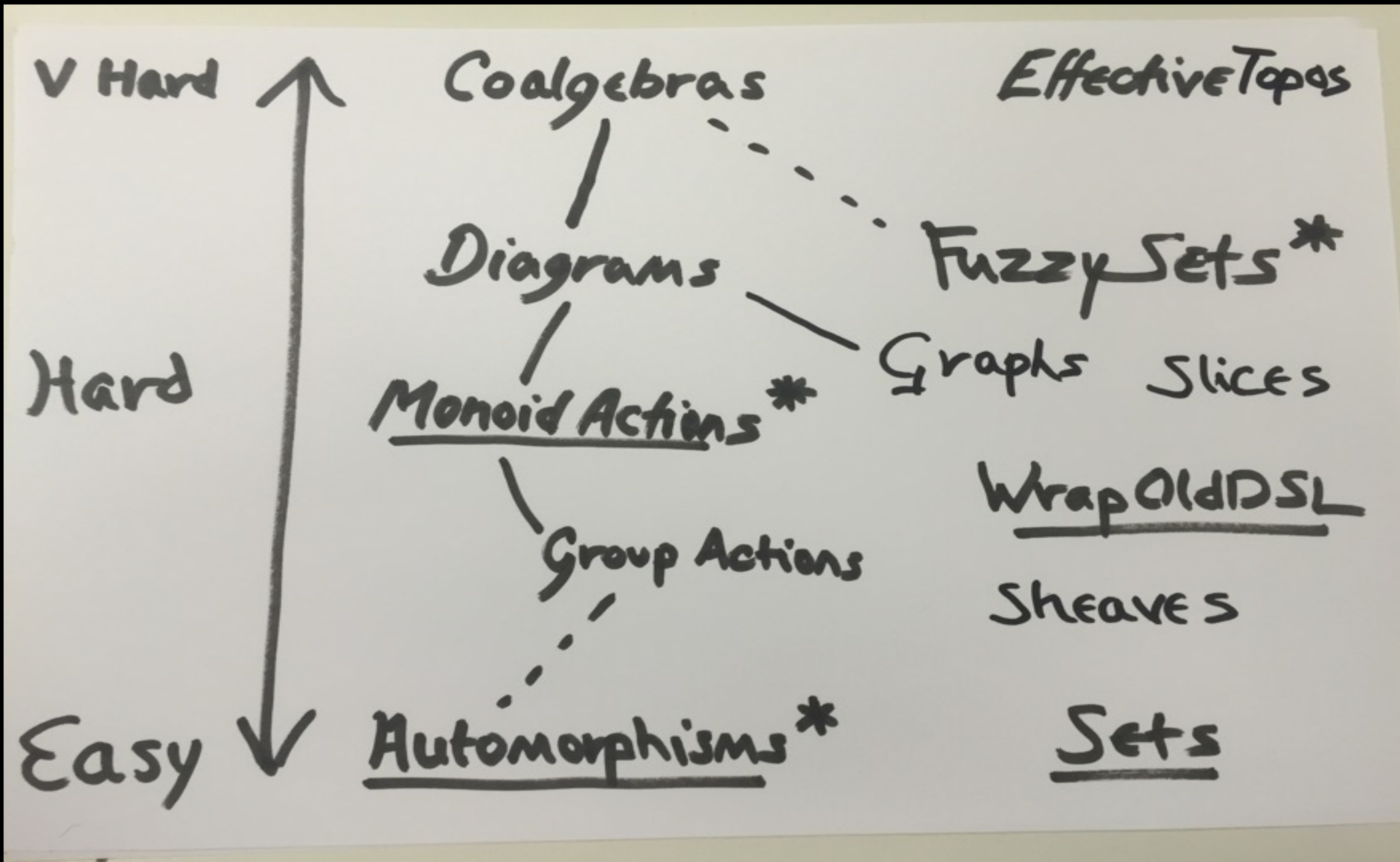
- A topos is a system of set-like objects - more precisely, a category with all the optional extras (*, +, ^, Ω)
- i.e. objects that can be added, multiplied, exponentiated and Ω'd in the sense of category theory
- Example: permutations form a topos. So do graphs, diagrams, fuzzy sets, 'musical objects', etc, etc
- Bewl is a DSL for the internal language of a topos
- Topos theory is like assembler ; Bewl is like C

HOW DO YOU USE BEWL?

- Write a class implementing the trait **com.fdilke.bewl.topos.Topos**
- Now you can talk fluently about its objects as if they were sets
- As the DSL developed, it got easier to write topos implementations
- But it’s still too hard
- So far, I have only written four

STATE OF THE BEWL 2016

MY 4 TOPOS IMPLEMENTATIONS



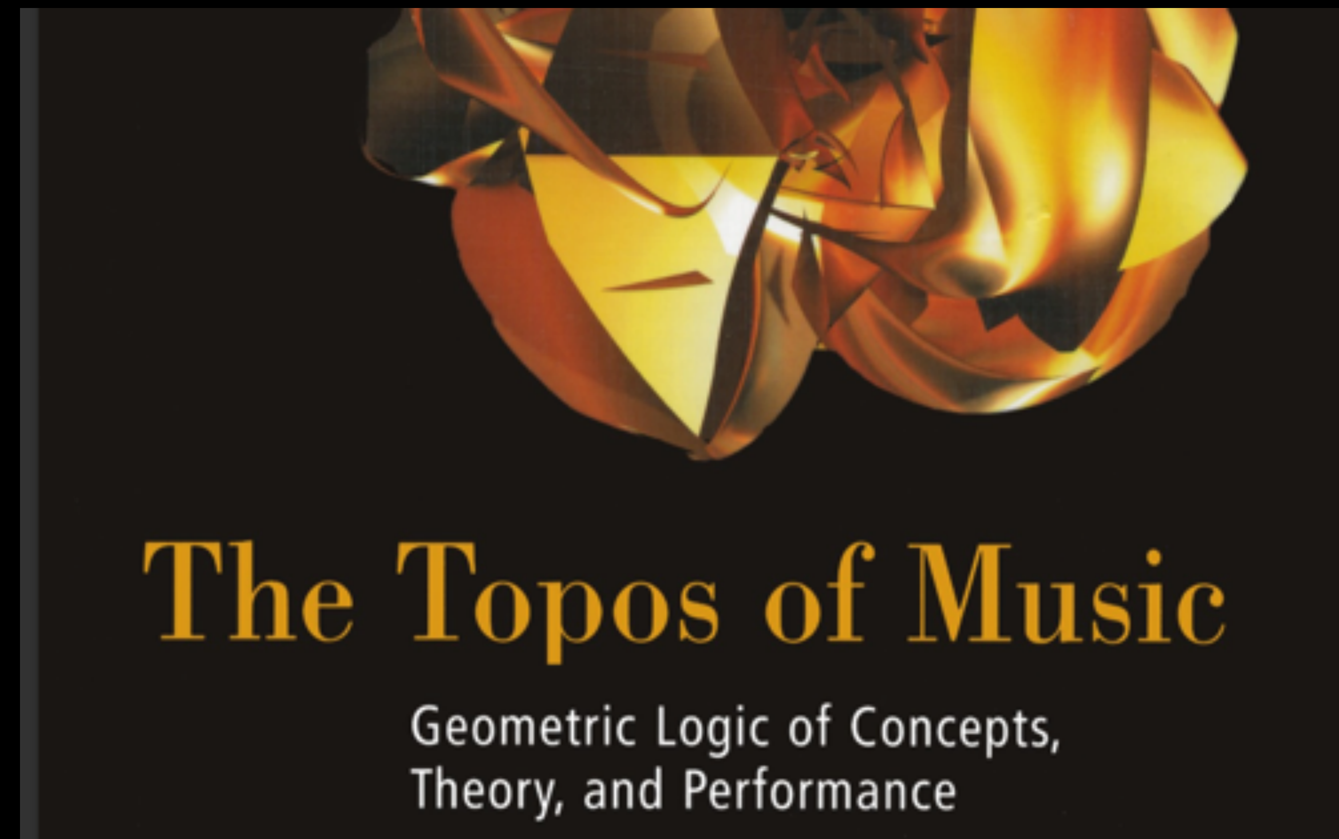
* = A TENUOUS CONNECTION WITH REALITY

CONNECTIONS WITH REALITY

- <u>The topos of monoid actions</u> can be applied to the structure theory of music (once I have speeded up my algorithms) because music turns out to be a kind of geometry
- <u>The topos of fuzzy sets</u> can be applied to inference and recommendation engines, because it is all about expressing truth values other than TRUE and FALSE
- <u>The topos of permutations</u> can be used to investigate the deep unexplained mystery of permutation parity (this was the original motivation of Bewl)

WHAT NEXT?

- More topos implementations
- More constructions, mapping whole blocks of math into software
- Strengthen the tenuous connections with reality
- Speed up the monoid actions code and apply it to the theory of music



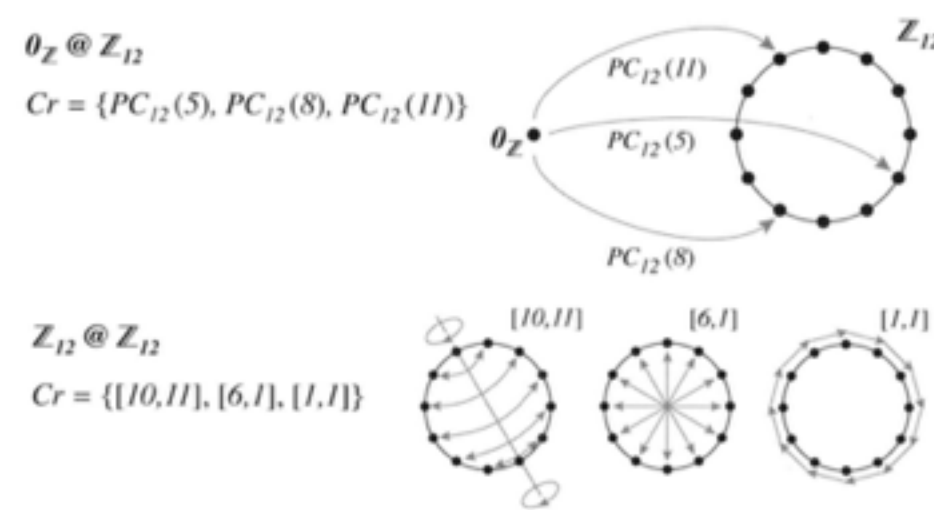


Figure 7.2: Above, a zero-addressed 12-tempered class 3-chord, below a self-addressed 12-tempered class 3-chord.

Whereas there are infinitely many (coordinate sets for) just class chords, there are only 2^w (resp. $\binom{w}{n}$) different (coordinate sets for) w-tempered class chords (resp. n-element class chords). For every couple Cr_1, Cr_2 of A-addressed chords or class chords, we can build their Boolean combinations: *union* $Cr_1 \cup Cr_2$, *intersection* $Cr_1 \cap Cr_2$, and *difference* $Cr_1 - Cr_2$. For a w-tempered class chord Cr, one may also build its *complementary chord* $Cr^\frown = \chi_w - Cr$, i.e. the difference from the w-*chromatic* class chord χ_w of support $\mathbb{Z}_w$.

We conclude with a remark on different addresses for chords. With the identification from (6.41) and notation from chapter 6, RegDen-9, a 0-addressed w-tempered class n-chord can be

"In mathematics there is a time lapse between a mathematical discovery and the moment when it is useful; and that this lapse of time can be anything from 30 to 100 years, in some cases even more; and that the whole system seems to function without any direction, without any reference to usefulness, and without any desire to do things which are useful."

– JOHN VON NEUMANN

THANK YOU

github.com/fdilke/bewl