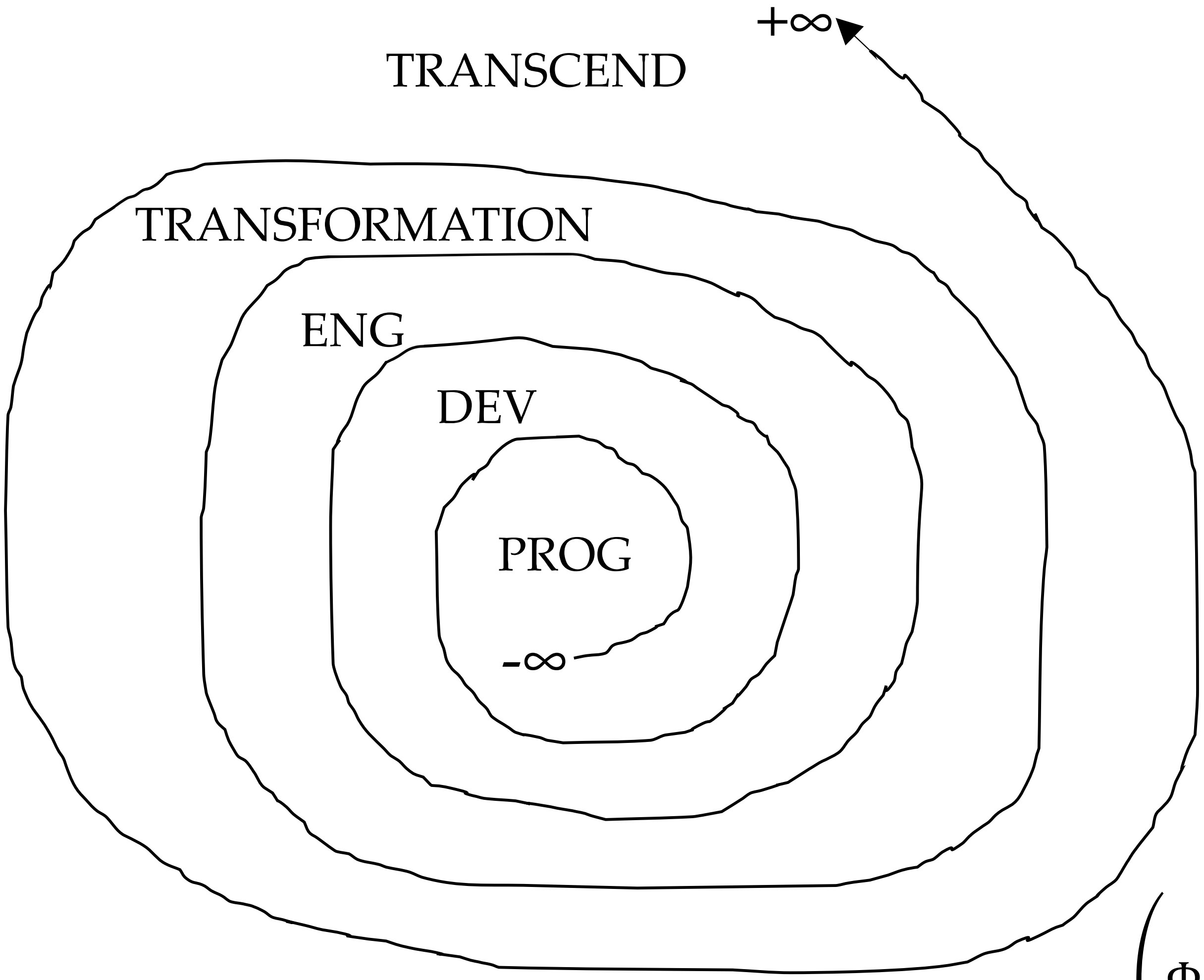
Levels of Functionality



Programming (ω^1): affecting practice areas through the creation of executable logic utilizing tools and following blueprints.

Development (ω^2): influencing programmers by creating their tooling according to blueprints.

Engineering (ω^3): analysis of problems to create solutions in the form of blueprints.

Transformation (ω^4): analysis of opportunities to identify problems

Transcend (ω^5): creating opportunity from chaos

The absolute, and nature of the, population of actors within a level of functionality can be defined by the following equation set:

$$\left(\mathfrak{D}^n = \int_{x=n}^{n+1} \omega^x \, d(-\infty)\right) \gg \mathfrak{D}^{n+1}$$

$$\left(\Phi^n = \iint_{x=n}^{n+1} \omega^x \, d(-\infty)\, d(+\infty)\right) \ll \Phi^{n+1}$$

$$PROG \cup DEV \ = \ Operations$$

$$DEV \cup ENG \ = \ Architecture$$

$$ENG \cup TRANSFORMATION \ = \ Innovation$$

$$TRANSFORMATION \cup TRANSCEND = \ Power$$

OPS: Execution of logic through tooling and applications.

ARCH: Design of tooling to enforce patterns and practices

INNOV: Creating blueprints to leverage new opportunities

POW: Create problems to spawn opportunity