

Describing a Model Before Making a Model: A Framework

A way to talk about models before we make them, not by what they look like, but by what we intend the model to do and the kinds of play we want to invite.

The Problem:

Anyone who has used a model knows their value to making the complex clear and building a shared understanding amongst multiple parties.

However, it can still prove a challenge to get approval for the time and budget required to go through the process if you can't provide a sense of what the output will be.

(Additionally, once you do get approval for the modeling process, that same lacking sense of what it will be often impedes starting.)

The Typical (but problematic) Solution:

A way around this problem has been to come up with a set of 'typical' models that can be referred to by a name that primarily points to what it will look like more than what it will do.

(Sitemaps look like trees, Journey Maps have horizontal channels and flow left to right, Wireframes look like bad websites, Business Process Models are boxes and arrows, etc.)

The **problem with this approach** is that the best models are the ones that don't conform to the typical, but instead bend and adapt to best represent the nature of the thing.

My Proposed Solution:

To preserve the modeler's autonomy and stakeholder expectations, we need to talk about what we intend the model to do by describing what aspect of a thing it will focus on and at what level of abstraction.

Your Challenge:

My challenge to you is to stop talking about the models you will make based on pictures of the models you (or someone else) have made in the past and instead talk about them in terms of how the models will be used.

Instead of talking about models based on what they will look like, describe the models you plan to make in terms of:

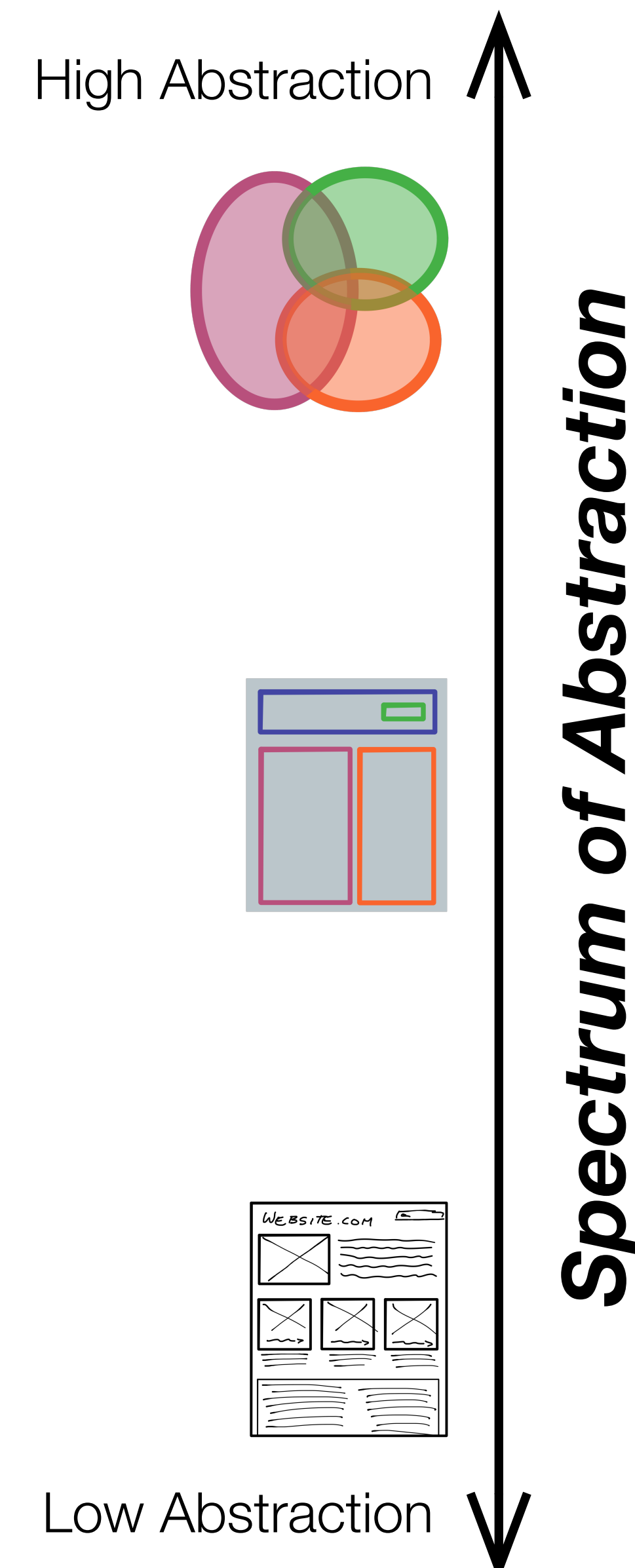
Description of a Model

- 1) How you will **USE** the model
- 2) What **ASPECT** of a Thing you'll focus on
- 3) What level of **ABSTRACTION** will best support your intended use

"The purpose of abstraction is not to be vague, but to create a new semantic level in which one can be absolutely precise."

- Edsger Dijkstra, Systems Scientist

Different conversations are enabled by different levels of abstraction.



| | | | Aspects of a Thing to Consider To create a clear model, only a few aspects can be considered at any one time. | | | | Facets of a Thing exist at different levels of granularity. All Things have many Facets, at many levels. | | | |
|--|--|--|---|---|---|---|--|---|---|---|
| Forces acting upon the Thing. All Things have many forces acting upon them. | | | Stakeholder Intent | User | Ecosystem/Environment | System/Concept | Structure | Interface | Interaction | Object |
| | | | "Stakeholder Intent Model" Use: Show what "good" means. Aspect: Stakeholder Intent Abstraction: Highly abstract | "User Model" Use: Share & discuss the diversity of thought and action of our users. Aspect: Users Abstraction: High-level | "System Information Model" Use: Express relationships of language and concepts of thought and action of our users. Aspect: Ecosystem, System, and Structure Abstraction: Won't look anything like a website. | "Structural Prototype Model" Use: Verify structural decisions with user, without being distracted by visuals. Aspect: System and Structure Abstraction: Loosely resembles interfaces, but no one will think it's the website. | | | | |
| "Interactive Prototype Model" Use: Verify good experience and understanding of our system. Aspect: Interface, Interaction, Object, and Structure (tiny bit) Abstraction: High-fidelity mock-up, might look like the website. | | | The Stakeholder Intent aspect of the Thing is what the stakeholders want out of the Thing. It answers the question "why was this Thing created?" (for existing Things) or "why should we make this Thing?" (for new Things) from the stakeholders' perspective. | The User aspect of the Thing looks at the intended users of the Thing. ("User" is intentionally non-specific, as there are many sub-aspects that may apply; "User Needs", "User Behavior", "User Lifecycle".) | The Ecosystem/Environment aspect of something looks at the Thing holistically, including external forces and factors. This aspect is less interested with the Thing than it is with how the Thing fits within and interacts with its environment. | The System/Concept aspect of a Thing looks at the Thing as a whole and considers the idea of the Thing rather than focusing on particular construction/makeup. | The Structure aspect of a Thing focuses on the primary relationships between the parts that make up that Thing. | The Interface aspect of a Thing focuses the parts of the Thing that a user is intended to interact with. | The Interaction aspect of a Thing focuses on the choreography and interplay of a user and the Thing. | The Object aspect of a Thing is focused on how a particular object, piece, or part of the Thing works and is designed to communicate its function. |