

### **Simple Book (Blocks: hero, metadata)**

The “Simple Book” project type was meant to address the use case that we’re seeing with most press users of Manifold. The most common approach, among presses, is to take an open access book and create a project for it. In these cases, it’s common for the project to only contain the hero area and the book itself. Once the book is up, there is very little activity, and no resources are added to it. We found that in Manifold 2.0, every project page would have a texts list with a single entry in it, which seemed redundant and even confusing, which is why we removed the texts list for this project type.

### **Enhanced Book (Blocks: hero, activity, texts, resources, metadata)**

The “Enhanced Book” represents what we’ve also referred to as the “Manifold Edition” over the course of the project. Wythoff’s Gernsback book and Metagaming are good examples of this on the UMN instance. This project assumes that there will be texts besides the published text—works in progress, supplemental material, notes, etc. It also assumes that the project will be augmented with resources. We include the activity block because the project is alive and changing over time.

### **Journal or Collection from single source (Blocks: hero, text TOC block, resources, metadata)**

We envisioned two different approaches to journals & collections in Manifold. The first approach assumes that the journal issue is contained within a single text that has multiple text sections, each of which represents an article within the journal. With this approach, each issue of the journal would be a project, and each project would have a single text. We expected that users would want to see the contents of the issue when they came to the project, which is why we included the TOC block here (instead of the texts list). We’re hopeful that the journal issue would be augmented with resources, thus the resources block. Because it’s a published work with ISBN, DOI, etc, we include metadata.

There is a current technical limitation with this approach, discussed on Slack yesterday. We do not currently have a mechanism to associate authors with specific text sections (because, if memory serves, this is not part of the EPUB spec which heavily informed how we originally modelled Texts in Manifold). With journals and collections, it’s important that users see the authors of each section (article) in the TOC, so this limitation needs to be addressed.

### **Journal or Collection from multiple sources (Blocks: hero, text list with abstract + author visible, resources, metadata)**

We also conceived of cases in which publishers would ingest one text (I’m using text in the Manifold sense—referring specifically to the model that belongs to a project—not in an abstract sense) for each article in the journal issue. For example, Authors might submit articles as google

docs, each with its own metadata (eg, a description of the article, the author name, etc). These individual texts could be grouped into text categories within the project to create structure for the volume. Because we're dealing with multiple texts in this approach, we would replace the TOC from the previous journal example with a Text list configured to show metadata about each article.

**OER or Teaching Resource (Blocks: hero, markdown block with placeholder text, activity, text list with abstract + author visible, resources, metadata)**

The was the least well-defined of the the options that we came up with in MN, and we basically threw the kitchen sink at it. I'm less able to recount our thinking here.

**Report or Handbook (Blocks: hero, text TOC block, resources, metadata)**

The idea here was that the Report or Handbook was similar to an enhanced book, except for the fact that it would only have one text (the published report or handbook). Because there is only one text, we proposed replacing the text list with the TOC to make the contents of the report front and center on the project page.

**Resource Only (Blocks: hero, markdown block with placeholder text, resources)**

I don't think we've seen cases of this in the wild, but the thinking was that we should encourage people to think of Manifold projects as containers for resources, and to show that Manifold could be used even when you don't have traditional texts.