Casey Reas and Ben Fry first met at MIT in 2001. Under the mentorship of John Madea the two technically minded designers created generative art through code, then moved on to develop the Processing Software. The Processing software package that we know and love today started in the MIT media lab in 2003 and was released in 2005. They were inspired by their mentors and the current technological context to make Processing as open source. This means that anyone could access the resources that comprise Processing and reuse the code for their own purposes. Now a decade later this software is still has been made popular by it's fully extensible code base, and intuitive IDE. Many developers have extended the capabilities of processing. This has extended Processing's medium to work on project surfaces, mobile phones, and electrical systems. It's been optimized and design for these various environments, as Casey Reas says

"We want to create a new culture of programming embedded within the visual arts"

The development environment itself has been reused for other applications such as the are doing a coding environment- in fact it is employed by Arduino as a circuit design suite.

Casey Reas now tours the world and presents at various conferences. He freelances as a design consultant while also coding exhibitions. He continues to develop his own motion graphics, preferring to do the coding himself. He has an extensive portfolio of graphics- with much work pertaining to large data sets. He's revelled as the driving force behind processing- and is active in documenting, discussing and developing the Processing engine.

Ben Fry works at Fathom design on the East Coast, building immersive visuals for enormous companies. In leveraging his extensive programming knowledge, Ben has designed high immersive visualizations for a variety of fields. Recent clients include GE and PG&E. As seen in the slides, his work

often comes at the intersection of architecture and graphics, as the canvas(or sketch) often becomes the walls in a building.