Time, Place, and Class Size: Friday 7PM, ATC 3.914, 10 ATEC and 10 CS students maximum

Overview: The goal is to cover creative programming practice using Max/Msp. Scripts and models will be visually crafted using the BPatcher capability in Max. End products are multimedia focused: sound/music, video, time-varying imagery, shader-based imagery, 2D/3D geometry. This class is a natural 2nd class in scripting where the focus is on visual scripting rather than text-based programming.

Visual Languages: Visual languages are less-well known than purely textual ones; however, their influence is rapidly growing in industry. Major game engines, image and video compositing tools, and model-based software engineering projects support visual node-based editing. Scripting for the Internet of Things (IoT), service oriented architecture networks, and workflows are visual. Most introductory languages for K-12 are visual. Visual languages offer natural integration with text-based coding, which exists underneath, and integrates with, the visual structures. Even though visual languages tend to be domain-specific, the underlying methodology is the same: flows through networks. We will cover the history of these languages as well as describing how fundamental principles (e.g., data storage, branching, iteration) are employed. We will cover the differences between thinking of process as writing vs. thinking visually.

Tools: We will be using Max/Msp. You will either use one of the ATEC labs or purchase your own copy of the software from cycling74.com. The cost is minimal for students: $10 each month during the class. Purchasing is highly recommended due to lab availability while classes are in session.

Pre-req: The student will need prior experience in traditional text-based scripting or programming.

Cost: No costs beyond purchasing the Max/Msp software. No required textbooks.

Instructor: Dr. Paul Fishwick, ATC 3.206 (paul.Fishwick@utdallas.edu)

Grading: Two Exams. Projects. Estimated # of projects estimated: 3-5 for the semester, each with incrementally phased deliverables. Students will give talks and demonstrations.