

Chapter XIX

The Philosophies of Software

Lindsay Grace

University of Illinois, and Illinois Institute of Art, USA

ABSTRACT

Software is philosophical. Software is designed by people who have been influenced by a specific understanding of the way objects, people and systems work. These concepts are then transferred to the user, who manipulates that software within the parameters set by the software designer. The use of these rules by the designer reinforces an understanding of the world that is supported by the software they use. The designer then produces works that mimic these same philosophies instead of departing from them. The three axes of these philosophies are analogy, reductivism, and transferred agency. The effects on computer-based artistic expression, the training in digital art production, and the critique of art are evaluated in this chapter. Tensions between the dominant scientific approaches and the dominant artistic approaches are also defined as destructive and constructive practice respectively. The conclusion is a new critical perspective through which one may evaluate computer integrated creative practice and inspire fresh creative composition.

INTRODUCTION

There is a simple logical proof that describes software's relationship to philosophy. Software is designed. Design prescribes philosophies. Since software is designed, it must also dictate philosophy. The existence of these philosophies, their sociological effects, and the need to critique these philosophies is the focus of this writing. This writing does not seek to define ontologies of philosophies, nor does it seek to be an exhaustive examination of the many philosophies that

have been institutionalized into the practices of developing and using software. Instead, this article seeks to highlight the existence of a few important philosophies in an effort to encourage practitioners to critically examine their relationship to software and its effects on their practice. In particular, critical assessment of software philosophies engenders fresh approaches to universal, original and effective design.

There are several existing areas where philosophy exerts an influence on software. Each of these areas is not only affected by inherent