

## **KIT: Kunst. Informatik. Theorie.**

Das Vorhaben untersucht die Wechselwirkung von künstlerischer Praktik und neuen Technologien unter der Perspektive einer künstlerischen Ausweitung/ Modifikation und Veränderung digitaler Werkzeuge. Insbesondere widmet es sich den noch nicht angemessen gewürdigten engen Beziehungen zwischen Kunsttheorie und Informatik im Rahmen einer Entwicklung zeitgemäßer künstlerischer Praktiken. Das Projekt verbindet somit theoretische und praktische Fragestellungen. Praktische Ziele sind insbesondere die Realisierung zweier Module (TXT.KIT und CODE.KIT), die dem Programm Kubim sowie allen Interessierten im Netz zugänglich sein werden.

### **TXT.KIT**

#### Aufbereitung eines Textkompendiums zur exemplarischen Entwicklung innovativer Wissensformen im Netz

Das zu bearbeitende Kern- oder Ausgangsmaterial liegt vor als Buch mit dem Titel 'KUNST ALS MEDIENTHEORIE. Vom Zeichen zur Handlung' (Prof. Dr. Hans Ulrich Reck). Das Material soll so aufbereitet werden, dass kollaborative Zugangsformen und Eingriffsmöglichkeiten zum Einsatz kommen. Es geht dabei nicht primär um eine erweiterte Archivfunktion, sondern um die Anwendung generativer, experimentierender Methoden im Rahmen geisteswissenschaftlicher Forschung. Dabei gilt es ein jeweils dem Thema zugehöriges Inventar (informatisch Ontologie genannt) zusammen zu stellen, das den Zugriff darauf durch die Software und den Benutzer ermöglicht. Durch Produktionsregeln werden diese in der Datenbank relational verknüpft und gefiltert. Es geht im wesentlichen um die "Materialisierung von Fragestellungen", weniger um die Verdopplung bisheriger Publikationsformate, wie sie durch elektronische Bücher (eBook/PDF) bereits betrieben wird. Die digitale Aufbereitung theoretischer Inhalte verfolgt vielmehr das Anliegen, die Arbeit am und mit dem Text aufzuzeichnen, zu visualisieren und damit auch das "Nicht-Begriffliche" als Bestandteil diskursiver Formationen transparent werden zu lassen. Diese Wissensbasis (Verlauf, Geschichte) ist ganz klar von der durch die Ursprungsdaten vorgegebenen Faktenbasis (Datenbank, Tabellen) zu trennen. Einzelne Wissensklassen können dabei autonome Funktionalitäten ausführen, um so entsprechend dem Gedanken des Konnektionismus eine Schnittstelle zu formen. Die Auseinandersetzung mit theoretischen Wissensquellen wird in diesem Zusammenhang als eine "im Werden begriffene wissenschaftliche Erfahrung" kenntlich.

Es soll die Situation geschaffen werden, in der Erkenntnisgewinn auf Erfahrung basiert, in dem Sinne, dass sich der Prozess der Aneignung der Ideen hin zu den Objekten des Wissens der Handelnden verschiebt. Die Handlungen der Nutzer generieren mit den Interface-Werkzeugen eine visuelle Matrix. Bei der Lektüre des Materials legen die Nutzer Spuren anhand von Termen und selektiven Filtern, die auch von anderen Nutzern verfolgt werden können (Shared Information Space). Die Aufzeichnung der Spuren basiert zwar auf Algorithmen, ähnelt ihnen jedoch keinesfalls. Stattdessen können sich heuristische Regeln der Spurensuche (Tracer oder Marker) herausbilden, die das spezielle Unternehmen von anderen, möglichen Anwendungen dieser Art unterscheidet.

### **CODE.KIT**

#### Entwicklung eines Java Programmierkurses für künstlerische Ausbildungskontexte

Die künstlerische Auseinandersetzung mit dem Computer und sein breiter Einsatz als Werkzeug der künstlerischen Praxis haben in den vergangenen Jahrzehnten Formen wie ‚Interaktive Kunst‘, ‚Computer Musik‘ und ‚Elektronische Literatur‘ hervorgebracht. Obwohl die zugrunde liegenden algorithmischen Strukturen bei vielen der Arbeiten essentiell sind, bleiben sie in der Regel verborgen, sie werden verdeckt (versteckt) durch die visuellen,

akustischen und taktilen Oberflächen. Unter dem Begriff ‚Software Art‘ ist derzeit eine neue Form der Auseinandersetzung mit dem Computer erkennbar. ‚Software Art‘ wird definiert als Verschiebung des Blickwinkels, weg von den Bildschirmen, Projektionen, Installationen und ihren vordergründigen Interaktionsmöglichkeiten, hin zur Erzeugung und der Auseinandersetzung mit algorithmischen Systemen und Prozessen. Der Programmcode und die algorithmischen Strukturen selbst sind nun das künstlerische Material, nicht mehr die Interfaces die oft dem Ziel untergeordnet bleiben, oberflächliche Wahrnehmungen und Erlebnisse beim Betrachter zu erzeugen.

Auch für die Frage, welche prinzipiellen Kenntnisse und Fertigkeiten Künstler in Bezug auf die neuen Technologien haben sollten, kommt der Programmierung eine zentrale Stellung zu. In der künstlerischen Ausbildung kann es nicht um oberflächliches technisches Wissen gehen, das mit jeder neuen Hard- oder Softwaregeneration veraltet. Vielmehr muss die Vermittlung allgemeiner Prinzipien im Zentrum stehen, deren Wandel sich in viel größeren Zeiträumen vollzieht. So können zum Beispiel Einsichten in die Prinzipien und Grenzen der Mechanisierung und des mechanistischen Weltbildes - das inzwischen als gleichermaßen kritisiertes wie verteidigtes Erklärungsprinzip für nahezu alle natürliche Erscheinungen herangezogen wird – nur über das Verständnis algorithmischer Grundprinzipien gewonnen werden.

Der Modul CODE.KIT stellt sich der Frage, welche Bedeutung die Programmierung für die künstlerische Praxis hat und unternimmt erstmals einen theoretisch fundierten Versuch, Künstler an die Programmierung heranzuführen. Als Programmiersprache wird die objektorientierte und plattformunabhängige Sprache Java verwendet. Begleitmaterial und alle Beispielprogramme des Kurses werden im Netz frei verfügbar gemacht.