

Wenn das Projekt noch nicht realisiert wurde, dann muss es noch realisiert werden. Hinweise zum derzeitigen Stand solltest Du im Netz finden.



Dieses Werk ist lizenziert unter einer
Creative Commons Namensnennung – Nicht-kommerziell –
Weitergabe unter gleichen Bedingungen 3.0 Deutschland Lizenz.

THE TIMELESS WAY

„There is one timeless way of building. [...] It is so powerful and fundamental that with its help you can make any building in the world as beautiful as any place that you have ever seen. It is so powerful, that with its help hundreds of people together can create a town, which is alive and vibrant, peaceful and relaxed, a town as beautiful as any town in history. And there is no other way in which a building or a town which lives can possibly be made.“ (7-8)

Die Eröffnungssätze aus „The Timeless Way of Building“ verdeutlichen sehr klar, dass Christopher Alexander absolut überzeugt von dem historischen Wert seiner Entdeckung ist. Dass jemand von seinen eigenen Ideen überzeugt ist und sie für weltgeschichtlich bedeutend hält, ist dabei nichts Besonderes. Aber als These angenommen, dass Alexander mit seiner Einschätzung Recht hat – vielleicht sogar damit, dass es der *einzigste Weg* ist –, und weiter angenommen, dass diese vorliegende Interpretation auf die Re-Produktionsweise nach Commons-Prinzipien sinnvoll ist; dann stünde uns die Gesellschaft nach Bedürfnissen und Fähigkeiten als konkrete Möglichkeit gegenüber, dann müssten wir sie „nur noch“ realisieren. Und auch wenn dieses „nur noch“ in Anführungszeichen steht, hätten wir statt einer vagen Hoffnung ein konkretes Ziel und eine konkrete Handlungsmöglichkeit, mit welcher diese neue Gesellschaftsform aus der Alten heraus entstehen kann. Und die Software selbst, zu deren Konzeption diese Interpretation dient, ist ein Werkzeug zu diesem Zweck.

Noch zwei Anmerkungen zu diesem ersten Zitat:

1. Wenn Alexander über Architektur redet, wird hier Commoning verstanden. Das erscheint erst einmal ungewohnt. Der Gedanke dazu ist, dass Alexander zwar Architekt ist, aber die Architektur nur der ihm naheliegende Gegenstand zur Anwendung einer allgemeingültigen Methode war. Die Methode selbst wird allerdings erst nach dieser Interpretation, im Anhang, erläutert.
2. Wenn auch die Analyse des Bestehenden auf das Commoning für die Sache der Transformation unbedingt notwendig ist, wird in diesem Text ausschließlich die Struktur des ununterbrochenen Commonings betrachtet. Die Abstraktion geht so weit, dass auch von bestehenden Eigentumsverhältnissen abgesehen wird und sämtliche Mittel als allgemein verfügbar und als dem Zweck der Bedürfnisbefriedigung untergeordnet verstanden werden.

„This one way of building has always existed. But it has become possible to identify it, only now, by going to a level of analysis which is deep enough to show what is invariant in all the different versions of this way.“ (10-11)

Commoning ist das miteinander-Teilen, das Beitragen, das Konflikte-auf-Au-genhöhe-klären, das füreinander-Dasein. Es ist nichts Neues, ganz im Gegenteil: Die Kooperation zur Herstellung und Erhaltung der gemeinsamen Lebensbedingungen ist das, was uns als Spezies seit jeher ausmacht, aber in einer un-

Qualität (S.5): Ein mittlerer Wert der sinnlich-funktionalen Qualität und der Prozessqualität eines Tätigkeitsmusters

Prozess- (S.7): Die Prozessqualität besteht sowohl aus dem Aufwand der bestimmten Tätigkeit, als auch dem Aufwand, welcher (spekulativ) zur Durchführung der Tätigkeit im lokalen Kontext notwendig wird.

sinnlich-funktionale – (S.7): Die sinnlich erfahrbare Qualität des Resultates eines Tätigkeitsmusters u. seine Zweckdienlichkeit
- ohne Namen: Der bei C. Alexander über die Anwend. von Mustern zu erreichende innere „anspannungslose“ Zustand.

- Im Commoning (S.4): Zustand der vollständigen Bedürfnisbefriedigung

Reparaturprozess (S.24): Manuelle Änderung von Konfigurationen durch die daran Beteiligten.

Re-Produktionsprozess: Produktion (im Kapitalismus: bezahlte Arbeit zur Geldverwertung) und Reproduktion (im Kapitalismus: meist unbezahlte (Fürsorge-)Arbeit zum Erhalt der in der Produktion tätigen Personen) als ein Prozess gedacht.

Selbstzuordnung (S.19): Zentrales Prinzip sich in das ununterbrochene Commoning einzubringen. Tätigkeiten werden im Rahmen des Konfigurations- und Reparaturprozesses den Beteiligten vorgeschlagen, welche sich schließlich einzelnen Tätigkeiten daraus annehmen.

Softwarefunktionen:

- Angabe von Kontinuität: S.10
- vorsorg. Reservierung von Mitteln: S.10
- Ortsveränd. im Konfig.-prozess: S.21
- Mengenzuordnung: S.21
- Kommunikation: S.23
- Gruppen-Kontinuität: S.24
- Analyse: S. 26
- Transparenz von Absprachen: S.20
- autom. Zeitplanung: S.28
- vorsorgende Selbstzuordnung: S.28
- Auflösung von int. Zusammenschl.:
- De-/Reaktivierung von Commons: S.29
- Autorisierung zur Teilhabe: 33

Sortierung (S.22): Reihenfolge, in welchen Anwender*innen freigeschaltete Tätigkeitsmuster des Konfigurationsprozesses vorgeschlagen werden.

Spannung (S.5): Vom lokalen Kontext abhängiger Aufwand zwischen dem Bedürfnis und seiner Befriedigung.

- Auflösung/Entstehung (S.7): Eine Spannung wird durch Tätigkeiten aufgelöst. Die sinnlich-funktionale Qualität einer Tätigkeit zeigt an, inwieweit sie eine bestimmte Spannung auflöst. Die Prozessqualität einer Tätigkeit zeigt an, inwiefern neue Spannungen durch ihre Ausführung entstehen.

Störung (S.28): Geschieht, wenn einer Tätigkeit in einem aktivierten Commoning-Prozess nicht bzw. nur problematisch (zeitverzögert oder mit niedriger sinnlich-funkt. Qualität) nachgegangen wird. Die Häufung von Störungen kann zur Krise führen

Tätigkeitsmuster (S.10): Beschreibt eine Tätigkeit, welche mit bestimmten Mitteln ein anderes Mittel erzeugt. (orts-)verändert oder erhält bzw. ein Bedürfnis direkt befriedigt.

- Rahmen (S.6): Der Bedarf den die Tätigkeit benötigt und das Resultat, welches daraus hervorkommt.

- Entstehung (S.9): Aus individueller Erfahrung heraus, welche in verarbeitbarer Form gesellschaftlich geteilt wird.

- Integration (S.10): Tätigkeitsmuster können in Beziehung gesetzt werden, wenn dasselbe Mittel Bedarf des einen und Resultat des anderen ist.

- Drei Aspekte der Beziehung (S.12): Das Problem, welches mit dem Resultat gelöst wird. Die Lösung, welche die Tätigkeit beschreibt. Der Kontext, in welchem die Tätigkeit auftritt.

komplexe - (S.11): Durch Fähigkeiten definierter Zusammenhang von Tätigkeitsmustern, denen sich „als ein“ Tätigkeitsmuster zugeordnet werden kann.

Tätigkeitsgewicht (S.16): Indikator der Dringlichkeit einer Tätigkeit durch Anzahl und Gewicht der Bedürfnisse, welche darauf verweisen
- Entstehung im Konfig.-prozess.: S.21

Zusammenschluss integrierter (S.29): Selbstorganisation mit eigener Regelsetzung, Tätigkeits- und Mittelkoordination als Teil des ununterbrochenen Commonings.

- Unterschiede zu Commons n. Ostr.: S.31
- Integration in die Softwarestr.: S. 31

Zuletzt zeigt sich ein fundamentaler Unterschied zwischen dem über die Software vermittelten Commonings und Alexanders Vorstellung einer Utopie: Diese auf dem technischen Fortschritt basierende Form des Commonings ermöglicht komplexe Strukturen zur generellen Bedürfnisbefriedigung und ermöglicht darin auch Räume der Selbstorganisation, welche Teil des Gesamtprozesses bleiben, in denen die Vermittlung über die Software allerdings nicht bestimmend ist. Die gesamtgesellschaftliche Koordination beruht allerdings auf dem Medium selbst, kann also nicht „erlernt“ und das Werkzeug kann nicht abgeworfen werden („*shedding it*“). Sobald keine Werkzeuge wie die hier beschriebene Software mehr verwendet werden, verliert sich auch die gesellschaftliche Transparenz. Und mit ihr nehmen die Möglichkeiten ab, wie sich in die gesellschaftliche Struktur eingebracht werden kann. Und mit ihr nimmt die Anzahl an Menschen ab, die sich prinzipiell den eigenen Bedürfnissen annehmen können und immer mehr Bedürfnisse bleiben dadurch wieder unbefriedigt. Ein wesentlicher Unterschied in der Vision – aber irrelevant für die Vorgehensweise an sich.

THE QUALITY

“We have been taught that there is no objective difference between good buildings and bad, good towns and bad. - The fact is that the difference between a good building and a bad building, between a good town and a bad town, is an objective matter. It is the difference between health and sickness, wholeness and dividedness, self-maintenance and self-destruction. In a world which is healthy, whole, alive, and self-maintaining, people themselves can be alive and self-creating. In a world which is unwhole and self-destroying, people cannot be alive: they will inevitably themselves be self-destroying, and miserable. - But it is easy to understand why people believe so firmly that there is no single, solid basis for the difference between good building and bad. - It happens because the single central quality which makes the difference cannot be named” (25)

Christopher Alexander macht sich folgend auf die Suche nach dieser *Qualität ohne Namen*, wie Marx sich auf die Suche nach dem *Wert* macht. Und wie Marx den Wert in den Dingen sucht, die nicht für einen selbst, sondern für den Markt und damit für andere produziert werden, sucht Alexander die *Qualität* als Essenz von Orten, an denen sich Menschen lebendig fühlen. Für beide ist schließlich das Resultat ihrer Suche der Ausgangspunkt, von welchem aus sie ein vollständiges System beschreiben können. Doch während der Wert bei Marx die Ware als einen äußeren Träger hat, ist die *Qualität ohne Namen* bei Alexander ein innerer Zustand.

Anhang

Ein Letztes: Ich habe bisher offen gelassen, warum die Interpretation von „The Timeless Way of Building“ auf das Commoning bzw. eine Software, welche Tätigkeit im gesellschaftlichen Rahmen nach Commons-Prinzipien vermittelt, meiner Ansicht nach sinnvoll funktioniert. Ich will dabei voranstellen, dass es sich mir selbst erst in der näheren Ausarbeitung des Textes vollständig erschlossen hat, wenn es sich auch von Anfang an sinnvoll *anfühlte*. Das Nachfolgende ist daher eher ein Erklärungsversuch, als eine von Beginn an feststehende Methode. Der Grund, warum ich die Methode überhaupt noch angebe ist rein pragmatisch: Vielleicht habe ich Fehler gemacht. Und falls dem so ist, kann über die Angabe der Methode möglicherweise herausgestellt werden, was ich falsch gedacht habe bzw. kann sie dabei helfen, die Interpretation leichter zu verstehen und gegebenenfalls manches nachzubessern oder zu verändern.

Für mich ist die große Gemeinsamkeit bei Alexander und dem Commoning ein im Zentrum stehender *problematischer innerer Zustand*, welcher nur durch eine *Veränderung der äußeren Welt* gelöst werden kann. Diese Veränderung der äußeren Welt, sprich: die Problemlösung, kann durch Alexanders Methode *in Einzelteile zerlegt* werden. Jedes dieser einzelnen Teile kann *für sich stehen* und als *Muster* beschrieben werden. In Bezug auf den jeweiligen Gegenstand kann einem solchen Muster eine *bestimmte Qualität* zur Problemlösung von *allgemeiner Gültigkeit* innerhalb *eines bestimmten Kontextes* zugeschrieben werden. Weiter kann ein solches Muster, unabhängig von seiner Qualität, Teil *verschiedener Problemlösungen* sein. Und noch weiter und zu Ende gedacht, kann *die Lösung sämtlicher wiederkehrender Probleme* (im Rahmen des jeweiligen Gegenstandes) durch eine Kombination solcher Muster beschrieben werden. Diese Muster dienen damit einerseits als Wissensspeicher und andererseits auch als Kommunikationsmittel. Und wieder explizit auf das Commoning bezogen, ermöglichen diese beiden Aspekte erst, dass die *Organisation der Tätigkeiten zur gemeinsamen Veränderung der Welt* auf Augenhöhe funktionieren kann.

Deswegen, ich wiederhole mich, empfinde ich die Software als so grundlegend notwendig um bestehende Herrschaftsverhältnisse auf emanzipatorische Weise überwinden zu können. Weitere sich aus der Struktur heraus ergebende Momente, wie etwa die *Bewegungstendenz zu immer weniger, aber hochwertigeren Mustern*, welche im Commoning einhergeht mit einer geringeren Zahl insgesamt verwendeter Mittel, sprich: *Modularität* wie sie auch von der „Open Source Ecology“ angestrebt wird, legen mir nahe, dass die Interpretation sinnvoll war. Der Text unterliegt dabei einer Creative-Commons-Lizenz (CC BY-NC-SA) und darf damit ohne weitere Nachfrage mit Namensnennung verbreitet und verändert werden, so lange der Text bzw. das daraus hervorgehende Werk nicht kommerziell verwendet wird und denselben Bedingungen unterliegt.

the space, and out of nothing else: they are the atoms and the molecules from which a building or a town is made.” (75)

Bei Alexander werden Muster von Ereignissen („*pattern of events*“) ins Verhältnis mit dem sie umgebenden Mustern des Raums („*patterns of space*“) gesetzt – im Commoning dagegen die Mustern von Tätigkeiten ins Verhältnis mit den dafür angewandten und daraus hervorgehenden *Mitteln* [Muster von Mitteln: SWK²]. Diese Mittel können dabei sowohl gegenständlicher, symbolischer oder sozialer Natur sein³: Von den Wohnblöcken bis zum Acker, von den Büchern bis zu den Smartphones, von der parlamentarischen Demokratie bis zum Konsensverfahren.

Die Mittel bilden dabei den *Rahmen* eines Tätigkeitsmusters: Der *Bedarf* einer Tätigkeit sind die Mittel, welche zur Ausführung einer Tätigkeit notwendig sind. Das *Resultat* ist das Mittel, welches durch die Tätigkeit entsteht. Einen zweiten Fall bilden Tätigkeiten, welche kein Mittel als Resultat haben, sondern ein Bedürfnis direkt befriedigen.

Der Zusammenhang zwischen den Mustern wird dabei in den *Tätigkeitsmustern* durchaus anders aufgefasst als bei Alexander. Bei Alexander *ist* jedes Muster Teil eines größeren Musters und *verweist* auf kleinere Muster. Zum Beispiel das Muster „LIGHTS ON TWO SIDES OF EVERY ROOM“: Das Muster selbst bleibt erhalten, während es aber Teil der *vorhergehenden* Muster „WINGS OF LIGHT“, „LONG THIN HOUSE“ und „CASCADE OF ROOFS“ sein kann, über welche überhaupt erst der Grundriss eines Gebäudes ermöglicht wird, damit in jedes Zimmer Licht von zwei Seiten fallen kann. Das Muster verweist schließlich (u.a.) auf die *nachfolgenden* Muster „WINDOW OVERLOOKING LIFE“ und „NATURAL DOORS AND WINDOWS“. Beide Muster betreffen den Ausblick aus den beiden Lichtquellen, womit sich selbstverständlich ihrem Vorhandensein selbst nichts ändert. Jedes Muster bei Alexander steht so zwar mit anderen in Zusammenhang und entfaltet sich durch diesen Zusammenhang, kann aber auch außerhalb des Zusammenhangs isoliert beschrieben und in anderen Zusammenhängen ebenso integriert werden.⁴

Tätigkeitsmuster dagegen beschreiben einen Prozess. Ein Tätigkeitsmuster zur Herstellung eines bestimmten gegenständlichen Produktes verweist etwa auf bestimmte gegenständliche Mittel, welche durch die Anwendung von bestimmten Werkzeug auf eine bestimmte Weise miteinander *verbunden* werden. Die verbauten gegenständlichen Mittel stehen anschließend *nicht länger für sich* – sie sind Teil eines neuen Mittels. Und das so lange, bis durch einen anderen Prozess das verwendeten Mittel wieder entnommen wird und das auch nur, falls durch den vorhergehenden Prozess das Mittel nicht unwiderruflich verbraucht bzw. verändert wurde.

“The specific patterns out of which a building or a town is made may be alive or dead. To the extent they are alive, they let our inner forces loose, and set us free; but when they are dead, they keep us locked in inner conflict.” (101)

Tote bzw. lebendige architektonische Muster werden übersetzt als Tätigkeitsmuster mit niedriger bzw. hoher *Qualität*. Da es sich bei Tätigkeiten aber um einen Prozess handelt, muss diese Qualität zweigeteilt werden: Da ist der *Prozess* und da ist sein *Resultat*.

stehen, desto besser können Beteiligte darauf vertrauen, dass Tätigkeiten mit *hoher Qualität* und *hohem Gewicht* dazu beitragen, eine Gesellschaft nach Bedürfnissen und Fähigkeiten herzustellen und zu erhalten. Ob Beteiligte dann diesen Tätigkeiten nachgehen bzw. diese Tätigkeiten selbst als sinnvoll empfinden, liegt immer in ihrem Ermessen. Abhängig allerdings ist all das weiterhin davon, ob eine angemessene Möglichkeit gefunden wird, den *allgemeinen Aufwand* einer bestimmten Tätigkeit festzustellen.

Wenn Spannungen im softwarevermittelten Commoning auch auf gesellschaftlicher Ebene verlaufen und damit nicht durch Menschen selbst *empfun-*den und entsprechend *aufgelöst* werden können, kann die Anwendung von Tätigkeitsmustern und damit die mögliche Ausbreitung von Commoning-Strukturen, wie auch bei Alexander, ebenso als ein *“gate“*, ein Durchgang, betrachtet werden. Indem sich Commoning als gesellschaftliche Re-Produktionsweise etabliert, etabliert sich damit auch seine gesellschaftliche Logik und durch die Normalisierung der Logik zeigt sich die Verrücktheit der heutigen Normalität: Die Abhängigkeit von existenziellen Ängsten als Arbeitsmotivation. Die ewig gleiche Arbeitswoche trotz ständigen technischen Fortschritts. Der ständige Versuch, neue Arbeitsplätze zu schaffen im scheinbar notwendigen Kampf gegen Arbeitslosigkeit. Und wenn es auch nicht das „Wiederbeleben einer Erinnerung“ und auch nicht eine „Rückkehr zur ursprünglichen Gesellschaft“ ist, so ist das ununterbrochene Commoning dennoch zeitlos, indem jede Geste der Freundlichkeit das Allgemeine dieser Gesellschaft in sich trägt. Und zwischen der Vermittlungsform der Software und der rein zwischenmenschlichen Vermittlung gibt es keinen Bruch: Durch die Anwendung der Tätigkeitsmuster in den softwarevermittelten Strukturen kann ein Verständnis der bedürfnisorientierten Re-Produktionsweise nach Commons-Prinzipien *verinnerlicht* und durch diese Erfahrungen auch *außerhalb* der softwarevermittelten Strukturen weitergetragen werden.

“Almost everybody feels at peace with nature: listening to the ocean waves against the shore, by a still lake, in a field of grass, on a windblown heath. One day, when we have learned the timeless way again, we shall feel the same about our towns, and we shall feel as much at peace in them, as we do today walking by the ocean, or stretched out in the long grass of a meadow.” (549)

Größter Dank und größte Wertschätzung gilt dem Lehrer lebendiger Ordnung, Christopher Alexander. Doch nicht minderer Dank, nicht mindere Wertschätzung und alle Hoffnung gilt denen, die sich das Projekt zu eigen machen, sich mit Entwicklung, Design, Organisation, Übersetzung und Verbreitung einbringen, die Kritik nicht als Sport üben, sondern zum Zweck der Verbesserung, der Erneuerung, des stetigen Anpassens an neue Möglichkeiten; sowohl technischer Natur als auch, um neue Anwender und Anwenderinnen zu erreichen. Dank, Wertschätzung und Hoffnung gilt auch denen, welche die Grenzen sehen, die dem Commoning gesetzt sind und daran arbeiten, diese Grenzen auszuweiten, zu lockern und zu lösen. Für uns, die eine Gesellschaft erstreben, die sich den Bedürfnissen und Fähigkeiten nach gestaltet, sieht die Zukunft längst nicht mehr rosig aus. Wir können nicht darauf hoffen, dass unsere Zukunft – die Zukunft unseres Begehrens – eintreten wird, wenn wir nicht anfangen abseits bisherigen Traditionen der heute bestimmenden Vernunft entgegenzutreten,

„There is always repetition of the patterns. But there is always variation and uniqueness in the way the patterns manifest themselves – Each pattern is a generic solution to some system of forces in the world. But the forces are never quite the same. Since the exact configuration of the surroundings at any one place and time is always unique, the configuration of the forces which the system is subject to is also unique – no other system of forces is ever subject to exactly the same configuration of forces.“ (146-147)

Commoning ist eine auf Freiwilligkeit und Selbstorganisation basierende, beständige Veränderung der Welt zugunsten der menschlichen Bedürfnisse. Ein einzelner Commoning-Prozess ist die Befriedigung eines bestimmten Bedürfnisses durch in Kooperation stehenden Tätigkeiten, welche auf Softwareebene durch Tätigkeitsmuster beschrieben werden. Jede Tätigkeit benötigt Mittel und fähige Personen zu ihrer Ausführung und weder die Mittel noch die fähigen Personen sind gleichmäßig verteilt. Welche Form der Commoning-Prozess eines lokal vermittelten Bedürfnisses annimmt ist somit nicht beliebig, sondern abhängig davon, welche Mittel lokal vorhanden sind und den Tätigkeiten zugeordnet werden können und ob Personen im lokalen Umfeld sowohl für die Tätigkeit befähigt, als auch daran interessiert sind und weiter diese Tätigkeit als sinnvoll sowohl für sich als auch für den Gesamtprozess betrachten. Kurz: Die Konfiguration (Anordnung und Auswahl) der Tätigkeitsmuster eines Commoning-Prozesses ist davon abhängig, an welchem Ort die Bedürfnisse vermittelt wurden, welche Mittel dort zur Verfügung stehen, welche Fähigkeiten die Personen dort besitzen, welche Strukturen dort bereits bestehen und weiter noch, welche Formen der Tätigkeiten/Kooperationen dort sinnvoll erscheinen. Dieser Ortsbezug wird folgend als der lokale Kontext [H/B⁹] bezeichnet, in welchem ein Tätigkeitsmuster angewendet wird.

Die stetige Veränderung der Mittel im lokalen Kontext betrifft direkt die Qualität eines Tätigkeitsmusters. Ein Tätigkeitsmuster, das zu einem Zeitpunkt eine hohe Qualität hatte und innerhalb der Gesellschaft weit verbreitet war, kann an einem späteren Zeitpunkt eine so niedrige Qualität haben, das es kaum noch oder gar nicht mehr angewendet wird. Und diese Qualität der Tätigkeitsmuster wirkt sich – genau wie anstehende Bedürfnisse, die Fähigkeiten und Interessen der Teilnehmenden – direkt auf die gegebenen Strukturen aus, die genauso der stetigen Veränderung unterworfen sind. Da feste Strukturen daher dem Wesen des Commonings widersprechen, geht es auch mit der Software nicht darum, solche aufzubauen. Wie, wie lange und ob diese sich ergeben, liegt außerhalb ihrer Funktion.

„But when we make our own attempt to create nature in the world around us, and succeed, we cannot escape the fact that we are going to die. This quality, when it is reached, in human things, is always sad; it makes us sad; and we can even say that any place where a man tries to make the quality, and be like nature, cannot be true, unless we can feel the slight presence of this haunting sadness there, because we know at the same time we enjoy it, that it is going to pass.“ (154)

grierten Zusammenschluss *Beteiligten*, sondern auch die von den *Tätigkeiten des Zusammenschlusses Betroffenen* autorisiert sein müssen, um an Prozessen der Regelerstellung und Abänderung teilhaben zu können. Es braucht daher eine Softwarefunktion, durch die angegeben werden kann, selbst – in irgendeiner Form – von einem bestimmten integrierten Zusammenschluss betroffen zu sein und damit die *Autorisierung zur Teilhabe* an diesen Prozessen der Regelerstellung und -abänderung zu erlangen. In jedem Fall braucht es auch *Kommunikationsräume zur schnellen und wenig aufwendigen Konfliktlösung* zwischen am Zusammenschluss Beteiligten und nicht am Zusammenschluss Beteiligten, die auf Softwareebene zur Verfügung gestellt werden sollten. Die Überwachung und Planung der konkreten Tätigkeiten und zugeordneten Mittel, die Erstellung und Abänderung von Regeln, die Form und Höhe der Sanktionen, die Zuteilung des Bedürfnisgewichtes sowie die Formen der gemeinschaftlichen Entscheidungsfindung und Konfliktlösung *zwischen am integrierten Zusammenschluss Beteiligten* sind dabei interne Prozesse desselben. Falls es Personen gibt, welche die Tätigkeiten und den Zustand der Mittel innerhalb des Zusammenschlusses überwachen, sollten auf Softwareebene allerdings definiert sein, wer diese Personen sind. Ein *Musterspeicher für Regel- bzw. Sanktionssysteme und Konfliktlösungsmechanismen*, die unabhängig von den Tätigkeiten eines integrierten Zusammenschlusses vorgeschlagen werden können, ist denkbar, betrifft aber nicht die Softwarevermittlung selbst.

"It is vastly more complex than any other kind of order. It cannot be created by decision. It cannot be designed. It cannot be predicted in a plan. It is the living testament of hundreds and thousands of people, making their own lives and all their inner forces manifest.“ (510)

Die Vermittlungsform einer Gesellschaft nach Bedürfnissen und Fähigkeiten ist unbedingt komplexer als jede bisherige gesellschaftliche Vermittlungsform es war. Diese Komplexität entsteht nicht von selbst, sie muss entwickelt werden und durch diese Entwicklung in den Hintergrund geraten, für eine Welt, die sich einfach erschließen lässt. Wenn die Software diese Funktion als Werkzeug erfüllen kann, dann kann eine gesellschaftliche Ordnung entstehen, die weder planbar noch vorhersehbar ist, aber durch nichts als die Bedürfnisse, Fähigkeiten und Vorstellungen der Beteiligten geformt wird.

"And as the whole emerges, we shall see it take that ageless character which gives the timeless way its name. This character is a specific, morphological character, sharp and precise, which must come into being any time a building or a town becomes alive: it is the physical embodiment, in buildings, of the quality without a name." (511)

Die Verfügbarmachung von Mitteln, welche sich durch den Akt des Weitergebens vermehren [H/B⁶], endet mit dem Abschluss der bestimmten Tätigkeit und befriedigt bzw. deckt damit alle darauf verweisenden Bedürfnisse bzw. Bedarfe. Für den Fall, dass es sich um Mittel handelt, welche sich entweder im Akt der Bedürfnisbefriedigung aufbrauchen bzw. nicht in der jeweiligen Situation *gemeinsam verwendet* werden können, benötigt die Verfügbarmachung von Mitteln für verschiedene Commoning-Prozesse eine (stetige) Wiederholung derselben Tätigkeit. Wenn einer Tätigkeit über der für einen einzelnen Commoning-Prozess notwendigen Dauer nachgegangen wird, wird nachfolgend von einer *kontinuierlichen Tätigkeit* gesprochen. Kontinuierliche Tätigkeiten haben tendenziell den Vorteil, dass ein Teil der dafür notwendigen Mittel nach Abschluss einer Durchführung nicht neu organisiert werden muss und damit die Prozessplanung vereinfacht wird. Weiter kann über eine kontinuierliche Tätigkeit auf Softwareebene festgestellt werden, welche Mittel zwar noch nicht lokal verfügbar sind, aber tendenziell einfach verfügbar gemacht werden können. Hierfür braucht es für am Commoning Beteiligte die Softwarefunktion, dass die Selbstzuordnung zu einer Tätigkeit auch bestehen bleibt, nachdem dieser erfolgreich nachgegangen wurde. Falls die Tätigkeit tendenziell häufig benötigt wird, kann eine Funktion der *vorsorgenden Reservierung* von dafür notwendigen Mitteln den Gesamtprozess unterstützen. Die Mittel wären damit nicht fest zugeordnet, aber durch die Reservierung können Absprachen und Auseinandersetzungen über die Verwendung von Mitteln zur generellen Bedürfnisbefriedigung unterstützt werden.

Kontinuität ist eine wesentliche Triebkraft zur Herstellung einer dichten gemeinsamen Mustersprache, welche selbst wiederum Voraussetzung für die *Effizienz* des ununterbrochenen Commonings ist.

Um sich dem anzunähern, noch einmal eine kurze Definition eines Musters:

"Each pattern is a rule which describes what you have to do to generate the entity which it defines" (182)

Jedes Tätigkeitsmuster beschreibt eine Tätigkeit, welche mit bestimmten Mitteln ein anderes Mittel erzeugt, (orts-)verändert oder erhält bzw. ein Bedürfnis direkt befriedigt.

Und was ist eine Mustersprache?

"A pattern language is a system which allows its users to create an infinite variety of those three dimensional combinations of patterns which we call buildings, gardens, towns." (186)

Die *Sprache aus Tätigkeitsmustern* ist ein System, das seinen Anwendern erlaubt jeden Prozess der Bedürfnisbefriedigung durch die Aneinanderreihung von Tätigkeitsmustern zu formulieren. Und wie jede Sprache hat auch sie eine Grammatik: Jeder Satz ist ein Commoning-Prozess. Ein Commoning-Prozess beginnt mit einem *Bedürfnis*, auf welches sich das erste Tätigkeitsmuster bezieht. Zwei Tätigkeitsmuster können nur über dasselbe Mittel verbunden werden und auch dann nur, wenn es bei dem einen ein *Bedarf* und bei dem anderen ein *Resultat* ist. Ein Satz ist nur vollständig, wenn es kein Tätigkeitsmuster mehr gibt, dessen Bedarf ungedeckt bleibt.

So ein Satz kann allerdings nicht ausgesprochen werden – um in dieser Metapher zu bleiben –, wenn es keine Menschen gibt, welche sich entsprechenden Tätigkeitsmustern zuordnen und auch dazu *befähigt* sind. Um sich den Aspekt

über diese Mittel („governance activities are organized in multiple nested layers“).

Wenn die integrierten Zusammenschlüsse auch starke Parallelen zu den nach Ostrom beschriebenen Commons haben, ist die Verwendung des Begriffes selbst irreführend und die Übertragung der Prinzipien darauf nicht widerspruchsfrei möglich: Das Commons bei Ostrom ist eine Besonderheit innerhalb einer durch (privates) Eigentum bestimmten Umgebung. Das zeigt sich besonders dadurch, dass es bei Ostrom als individueller Vorteil um die „Aneignung“ (*Appropriation*) der Ressource geht. Im Zentrum steht also ein Mittel (bzw. die „Ressource“), das nicht Eigentum einer konkreten Person bzw. einer Gruppe konkreter Personen ist und wodurch niemand alleine über dessen Nutzung bestimmen kann. Aus diesem Grund kann sich um dieses bestimmte Mittel ein Zusammenschluss von Personen bilden, in welchem alle dasselbe Nutzungsrecht an der Ressource haben und welcher schließlich dessen Nutzung regelt. Und dieser Zusammenschluss und dieses Mittel werden bei Ostrom als *Einheit*, als *Commons*, beschrieben. In einer von Commoning bestimmten Umgebung löst sich diese Einheit dagegen auf. Die für den Prozess des Commonings verfügbaren Mittel sind nicht länger Inseln, sondern das Fundament, auf welchem sich die Beteiligten – und damit auch Zusammenschlüsse von Beteiligten – bewegen.¹³

Die Übertragung der Prinzipien ist weiter deswegen problematisch, weil sich bei Ostrom sämtliche Regelungen, Sanktionen und Konfliktlösungsmechanismen den Zweck haben, die gemeinsame Verwendung des Mittels zu ermöglichen bzw. das Verhältnis des Mittels zu anderen Institutionen zu klären. Im ununterbrochenem Commoning dagegen hat der Zusammenschluss einen anderen Zweck: Die *generelle Bedürfnisbefriedigung*. Bei Ostrom entsteht dabei der individuelle Vorteil über diese *Aneignung* des Resultates der Tätigkeit am Mittel und im Gegensatz muss etwas *bereitgestellt* werden. Im ununterbrochenem Commoning dagegen wird das Resultat der eigenen Tätigkeit bzw. der Tätigkeit des Zusammenschlusses von anderen verwendet und der individuelle Vorteil entsteht auf *andere Weise* (abermals kann an dieser Stelle nur auf die Textreihe zum Softwarekonzept verwiesen werden).

In Einbeziehung dieser Unterschiede wird folgend die Software den Prinzipien von Ostrom nach differenziert, damit solche nicht über die Software vermittelten Zusammenschlüsse innerhalb des ununterbrochenen Commonings ihren eigenen Regeln nach wirken und tendenziell langlebig sein können.

Die Entstehung von integrierten Zusammenschlüssen bedeutet auf Strukturebene der Software eine Auflösung der für sich stehenden Tätigkeitsmuster innerhalb des entsprechenden Zusammenhangs, während der integrierte Zusammenschluss selbst deren Platz einnimmt. Ein integrierter Zusammenschluss kann, *muss aber nicht*, innerhalb der durch die Softwarevermittlung hergestellten Struktur und damit aus Tätigkeitsmustern heraus entstehen, ist aber ab seiner Entstehung unabhängig von diesen. Jeder integrierte Zusammenschluss ist dabei prinzipiell *einzigartig* durch die Regeln der daran Beteiligten und ihrer Organisation der Tätigkeiten. Während ein Tätigkeitsmuster daher einer möglichst allgemeingültigen Beschreibung bedarf, um es in möglichst viele verschiedene Commoning-Prozesse gleichartig durch unterschiedliche Personen integrieren zu können, ist die *Beschreibung* eines integrierten Zusammenschlusses innerhalb der Software optional. Die Beschreibung kann dazu dienen, die eigenen Strukturen und Regeln verständlich zu machen und sollte nur durch die am Zusammenschluss Beteiligten bzw. in Absprache mit

"The adaption between people and buildings is profound – Each detail has meaning. Each detail is understood. Each detail is based on some person's experience, and gets shaped right, because it is slowly thought out, and deeply felt. Because the adaption is detailed and profound, each place takes on a unique character. Slowly, the variety of places and buildings begins to reflect the variety of human situations in the town. This is what makes the town alive." (231)

Es ist nicht nur die statische Auswertung des lokalen Kontextes durch die Verfügbarkeit der Mittel oder kontinuierlicher Tätigkeiten, welche einem neu entstehenden Commoning-Prozess seine Form geben. Über die Software werden lediglich Tätigkeiten vorgeschlagen, welche den bekannten Daten nach sinnvoll erscheinen. Die Auswahl der Tätigkeitsmuster, deren Ausführung die lokale Umwelt auf eine bestimmte Weise verändert, liegt alleine in den Händen der Beteiligten, ihren Fähigkeiten, ihrem Interesse und in dem, was ihnen in dieser Welt wichtig ist. Und jedes Tätigkeitsmuster entstammt der persönlichen Erfahrung eines anderen Menschen zu einer anderen Zeit an einem anderen Ort und was dieser Mensch zur Lösung desselben Problems als sinnvoll erachtet hat. Und die Strukturen des ununterbrochenen Commonings bilden sich heraus, indem andere Beteiligte diese individuellen Erfahrungen ebenfalls als sinnvoll zur Lösung desselben Problems betrachten. Wenn sich auch noch Bewegungstendenzen innerhalb der Mustersprache herausstellen werden, können diese Strukturen, durch welche an bestimmten Orten auf bestimmte Weise die Welt verändert wird, durch keine abstrakte Logik vorhergesagt werden, sondern ergeben sich aus der Vielfältigkeit der daran beteiligten Menschen selbst.

"Each pattern is a three-part rule, which expresses a relation between a certain context, a problem, and a solution – As an element in the world, each pattern is a relationship between a certain context, a certain system of forces which occurs repeatedly in that context, and a certain spatial configuration which allows these forces to resolve themselves. [...] It is both a process and a thing; both a description of a thing which is alive, and a description of the process which will generate that thing." (247)

Drei Aspekte stehen bei einem Tätigkeitsmuster im Zusammenhang: Das Problem, die Lösung und der Kontext.

Das *Problem* eines Tätigkeitsmusters ist der Zweck, zu welchem es abgerufen wird. Der Bedarf also, der damit gedeckt oder das Bedürfnis, das damit befriedigt werden soll. Das Problem soll durch das *Resultat* der Tätigkeit möglichst vollständig gelöst werden. Inwiefern das Resultat geeignet ist, dieses Problem zu lösen, wird über die *sinnlich-funktionale Qualität* festgestellt.

Die *Lösung* eines Tätigkeitsmusters ist die Beschreibung der Tätigkeit mit ihrer Definition der dafür notwendigen Mittel und Fähigkeiten. In ihr wird beschrieben, wie unter Anwendung der notwendigen Mittel das Resultat hervorgebracht wird. Über die *Prozessqualität* wird festgehalten, wie aufwendig die Tä-

stützende Funktion höchstens angeboten werden, solche Absprachen transparent zu machen, um so Planungsprozesse für andere zu erleichtern.

„*Vierter Schritt: Aufweis des Dominanzwechsels zwischen der für die frühere Stufe charakteristischen Funktion und der neuen Funktion, womit durch einen zweiten qualitativen Sprung die qualitativ spezifische Funktion auch die für die gesamte Systemerhaltung bestimmende Funktion wird. [...] Eine solche Umkehrung des Verhältnisses zwischen bestimmender und nachgeordneter Funktion als Dominanzwechsel ist, obwohl sich beide Funktionen in der Entwicklung kontinuierlich darauf zubewegen, selbst nicht kontinuierlich, sondern ein punktuelles Umkippen.*“ (Holzkamp: 80)

Der „*Dominanzwechsel*“ geschieht für die Beteiligten nicht plötzlich. Er entsteht, indem einzelne Beteiligte sich immer mehr auf die selbstgesetzten Regeln und Absprachen beziehen und sich damit immer weniger (im Rahmen dieses Zusammenhangs) über die Software vermitteln. Der „*zweite qualitative Sprung*“ sind dabei solche Absprachen, die in die Organisation der Mittel und der Tätigkeiten selbst eingreifen, wodurch die Softwarevermittlung zum *Hemmnis* wird. Wenn es also für die Beteiligten ein *kontinuierlicher Prozess* zur von der Software weitgehend unabhängigen Selbstorganisation ist, braucht es auf Softwareebene einen *Bruch* mit den für sich stehenden Tätigkeiten, welche durch etwas *qualitativ anderes* ersetzt werden müssen. Dieser Bruch innerhalb der Software muss dabei durch die Beteiligten ausgelöst werden, geschieht also nicht von selbst.

Sofern ein solcher softwareunabhängiger Zusammenschluss, in welchem sich die daran beteiligten Personen ihre eigenen Regeln setzen und Tätigkeiten und Mittel selbstständig koordinieren, Teil des ununterbrochenen Commonings bleibt, wird er als „*integrierter Zusammenschluss*“ bezeichnet. Eine definierte Gruppe von Personen schließt sich dabei auf einer Ebene zusammen, welche die Software nicht fassen kann, aber dieser Zusammenschluss muss durch die Integration in das ununterbrochene Commoning von der Software gefasst werden können. Und das dabei nur insofern, wie die daran Beteiligten sich in das ununterbrochene Commoning einbringen wollen. Denn, und das nur eine allgemeine Erinnerung, das softwarevermittelte Commoning ist lediglich *eine Vermittlungsform* und kann nur das abbilden und nur das unterstützen, was von den Beteiligten gewollt ist. Es ist ein *Werkzeug* um Commoning auf gesamtgesellschaftlicher Ebene überhaupt betreiben zu können, aber nur ein Aspekt des Commonings an sich. Wenn ein Zusammenschluss innerhalb der Softwarestruktur auftritt, bedeutet es damit nicht unbedingt, dass er erst unmittelbar entstanden ist und wenn ein solcher Zusammenschluss aus der Softwarestruktur verschwindet, bedeutet es nicht unbedingt, dass er nicht mehr existiert.

Die Integration eines Zusammenschlusses *ersetzt* die Tätigkeitsmuster an entsprechender Stelle. Und sowie die Entstehung solcher Zusammenschlüsse aus der Vermittlung über Tätigkeitsmuster heraus unterstützt werden soll, muss auch die Auflösung in Tätigkeitsmuster zurück möglich sein. Außerdem benötigt die Software die Funktion, dass ein Zusammenschluss auf Ebene der Softwarestruktur *deaktivierbar* ist, um ihn zu einem späteren Zeitpunkt wieder *reaktivieren* zu können. Diese Funktion schließt damit an die Softwarefunktio-

"The language is a good one, capable of making something whole, when it is morphologically and functionally complete – It is morphologically complete, when the patterns together form a complete structure, filled out in all its details, with no gaps. And it is functionally complete when the system of patterns has that peculiar self-consistency in which the patterns, as a system, generate only those forces which they themselves resolve – so that the system as a whole can live, without the action of self-destroying inner conflicts" (316)

Das Ziel ist eine vollständige *Sprache aus Tätigkeitsmustern*, durch welche jeder einzelne Prozess zur Herstellung, Erhaltung und Veränderung der gesellschaftlichen Lebensbedingungen durch hochwertige Tätigkeitsmuster beschrieben werden kann. Ein darauf beruhendes Gesellschaftssystem wäre nicht statisch, sondern in ständiger Veränderung mit den Bedürfnissen jedes einzelnen Gesellschaftsteilnehmers, durch ihre steigenden Handlungsmöglichkeiten durch die stetige gesellschaftliche Verfügbarmachung neuer Erfahrungen und durch die inneren und äußeren Konflikte, in denen sich das System ständig bewähren muss. Und gleichgültig, ob es Auseinandersetzungen um die sinnvolle Anwendung begrenzter Mittel oder systemgefährdende Wandlungen der nicht-menschlichen Natur sind: Eine vollständige Mustersprache bietet zu jedem Problem Lösungen an.

"A language is a living language only when each person in society, or in the town, has his own version of this language. – For it is then not only an intellectual thing which expresses patterns as invariants, as rules to follow, as knowledge about what makes a building or a town work right. – It is, a deeper thing, a felt thing, a thing lived through, which expresses people's innermost attitudes about their way of life, their hopes and fears about the ways in which they live and work together" (337)

Auf Softwareebene sind *individuelle Tätigkeitsmuster* („eigene Erfahrungen zur Lösung eines bestimmten Problems“) nur gesellschaftlich verarbeitbar, wenn es sich dabei auch gleichzeitig um geteilte Tätigkeitsmuster handelt. Geteilte Tätigkeitsmuster, denen sich individuell angenommen werden kann, sind die eigene *Bibliothek*. Jedes bewältigte Tätigkeitsmuster, dem sich mindestens einmal erfolgreich angenommen wurde, kann automatisch in dieser Bibliothek aufgenommen werden. Da Fähigkeiten („verinnerlichte Tätigkeitsmuster“) ebenso aus erfolgreich abgeschlossenen Tätigkeitsmustern bestehen, können die entsprechenden Tätigkeiten innerhalb der Bibliothek durch die Anwender markiert werden. In dem Fall kann sich zukünftig Tätigkeitsmustern angenommen werden, welche diese markierten Tätigkeitsmuster als Fähigkeiten voraussetzen.

Eine solche Bibliothek als individuelle Mustersprache ermöglicht, dass nicht nur aktiv nach Möglichkeiten gesucht werden muss, um sich in den gesellschaftlichen Re-Produktionsprozess einzubringen. Über die Bewertung bestimmter Tätigkeitsmuster in der eigenen Bibliothek kann eingestellt werden, wann eine Benachrichtigung erfolgen soll, wenn Commoning-Prozesse in der lokalen Umgebung anfallen, welche auf diese Tätigkeitsmuster zurückgreifen.

zessen besteht, kann uns hier nicht helfen. Wer sich dagegen intensiv mit genau solchen transformativen Prozessen auseinandergesetzt hat, ist Klaus Holzkamp.

Auftritt Holzkamp.

Wie schon bei Alexander interessiert uns in Holzkamps Werk nicht sein Gegenstand, sondern seine Methode, die er in seinem Hauptwerk „die Grundlegung der Psychologie“ (1985) herausgearbeitet hat: Ein Fünfschritt, in welchem der Umschlag von der *Quantität* zu einer neuen *Qualität* präzise beschrieben wird, unter der Voraussetzung bestimmter Bedingungen und gegebener Entwicklungswidersprüche. Die erste Anwendung dieses Fünfschritts im Bereich der gesellschaftlichen Transformation stammt dabei von Stefan Meretz, welcher diese in „Kapitalismus aufheben“ (2018) gemeinsam mit Simon Sutterlütti näher ausgearbeitet hat. Und während Klaus Holzkamp seine Methode verwendet hat, um zu analysieren, *was bereits passiert ist*, wird folgend spekulativ angedacht, welche Handlungsmöglichkeiten bei steigender Komplexität sinnvoll werden *könnten* und wie die Software sie im Falle ihres Auftretens unterstützen kann. Weiter soll damit auch herausgestellt werden, *wie bestehende Formen der Selbstorganisation*, welche sich nicht innerhalb des Rahmens der Softwarevermittlung gebildet haben, sich – soweit es für die daran Beteiligten sinnvoll erscheint – in die Struktur des ununterbrochenen Commonings integrieren können.

„Erster Schritt: Aufweis der realhistorischen Dimension innerhalb der jeweils früheren Stufe, auf denen der qualitative Umschlag sich vollzieht. [...] Es soll genau die ›Position‹ bestimmt werden, die beim qualitativen Umschlag dialektisch ›negiert‹ wird.“ (Holzkamp: 78)

Die durch die Software ermöglichte *Vorbedingung* (Holzkamp mit dem Blick zurück: „*realhistorische Dimension*“) ist eine transparente Struktur aus tendenziell kontinuierlichen Tätigkeiten in lokaler Nähe, wobei jede davon für sich steht, diese aber durch ihren Bedarf an Mitteln und dem Zweck in ihrer Ausübung in Abhängigkeit zueinander stehen. Die Organisation der Mittel und die Integration der eigenen Fähigkeiten in die ununterbrochenen Prozesse der direkten Bedürfnisbefriedigung wird über die Software vermittelt.

„Zweiter Schritt: Aufweis der objektiven Veränderungen der Außenweltbedingungen, mit denen der ›innere‹ Entwicklungswiderspruch, durch welchen die neue Qualitätsstufe in evolutionärer Progression hervorgebracht werden kann, in seinem Umwelt-Pol zustandekommen soll.“ (Holzkamp: 79)

Wird die Software selbst als der „*Umwelt-Pol*“ der am ununterbrochenen Commoning Beteiligten betrachtet, dann ergeben sich die „*objektiven Veränderungen der Außenweltbedingungen*“ durch die für sie steigende Komplexität des Gesamtprozesses und die damit einhergehende, auf dem fragilen Fundament der Freiwilligkeit beruhende, zunehmende Abhängigkeit von der Ausführung bestimmter Tätigkeiten. Weiter noch, als Teil dieses „*inneren Entwicklungswiderspruches*“, kann die zunehmende Komplexität und steigende Abhängigkeit von einzelnen Tätigkeiten erfordern, dass Personen, deren Tätigkeit in verschiedenen Commoning-Prozesse integriert ist, einer immer strikteren Zeitplanung unterworfen sind und somit Handlungsfreiheit, welche das Commoning

lokal verfügbaren Mittel den Bedarf dieser Tätigkeit stetig decken müssen und andererseits auf das Resultat dieser Tätigkeit stetig zurückgreifen können. Für jede dieser Tätigkeiten gilt dasselbe wie für die erste Tätigkeit, dass sie also umso eher *kontinuierlich* werden können, je höher ihre Qualität ist, also je weniger aufwendig ihre Ausführung bei einer hohen sinnlich-funktionalen Qualität ihrer Resultate.

Indem die Mittel der lokal anstehenden Tätigkeiten dauerhaft aufeinander verweisen, entsteht eine *Verzahnung* des lokalen Kooperationsprozesses und durch eine Integration jeder Tätigkeit in möglichst viele verschiedene Commoning-Prozesse eine *Verdichtung* der insgesamt benötigten Mittel, welche durch die Verzahnung der Tätigkeiten tendenziell weniger aufwendig verfügbar gemacht werden können. Falls schließlich neue Bedürfnisse anstehen und es Tätigkeiten gibt, welche diese Bedürfnisse weitgehend durch die lokal verfügbaren (bzw. leicht verfügbar zu machenden) Mittel befriedigen können, dann sind diese Tätigkeiten von Anfang an qualitativ hochwertig und werden *immer hochwertiger* je leichter die Verfügbarmachung der Bedarfe im lokalen Kontext wird. Insofern ihre Resultate häufig benötigt werden bzw. vielseitig sind, gliedern sich diese Tätigkeiten damit ebenfalls in den lokal verzahnten Kooperationsprozess ein.

Damit diese Bewegungstendenz anlaufen kann und ununterbrochenes Commoning damit zur individuellen Bedürfnisbefriedigung immer effizienter wird, braucht es neben der Qualität einer Tätigkeit noch einen Indikator, in wie viele Commoning-Prozesse sich eine bestimmte Tätigkeit integrieren lässt. Dieser Indikator wird folgend als das *Tätigkeitsgewicht* [SWK⁹] bezeichnet und steigt, je mehr Bedürfnisse darauf verweisen. Weiter aber muss es nicht nur gesellschaftlich, sondern auch *individuell sinnvoll* werden, sich Tätigkeiten anzunehmen, welche einerseits eine hohe Qualität und andererseits ein hohes Gewicht aufweisen. Möglich wird das einerseits zu durch die *Sortierung* der Tätigkeiten nach ihrer Qualität und andererseits über eine Kopplung des Gewichtes einer Tätigkeit, welcher eine Person nachgeht, mit dem *Gewicht ihrer eigenen Bedürfnisse* (näher dargestellt in der Textreihe zum Softwarekonzept¹⁹). Die Gewichtung der eigenen Bedürfnisse wirkt sich damit auf das Gewicht der damit zusammenhängenden Tätigkeiten aus und „zieht“ das ununterbrochene Commoning in Richtung der eigenen Bedürfnisbefriedigung, ohne dabei einen Zwangsaspekt in das Commoning hineinzubringen.

„Of course, this evolution will never end. – Although the process of evolution will always move towards greater depth and greater wholeness, there is no end to it – there is no static perfect language, which, once defined, will stay defined forever. No language is ever finished.“ (346)

Durch das Werkzeug der gesellschaftlich geteilten Mustersprache und entsprechenden Funktionen, diese Tätigkeitsmuster sinnvoll anwenden zu können, haben wir es mit einer Gesellschaft nach Bedürfnissen und Fähigkeiten zu tun, die nicht einmal erreicht ist, sondern sich in ständiger Anpassung an die Veränderung der menschlichen und nicht-menschlichen Natur befindet. Ermöglicht wird dies, indem jeder Mensch mit seinen eigenen Erfahrungen die Möglichkeiten aller Beteiligten erweitern kann und durch seine eigenen Fähigkeiten und Interessen die Richtung beeinflusst, in welche sich die Gesellschaft entwickelt.

Die Veränderung eines Commoning-Prozesses kann jederzeit vorgenommen werden, sofern sie in Übereinkunft mit allen davon Betroffenen geschieht. Rein auf *Ebene der Effizienz* kann ein solcher Reparaturprozess sinnvoll sein, um Tätigkeitsmuster mit höherer Qualität und höherem Gewicht in den Gesamtprozess zu integrieren. Besonders bei kontinuierlichen Tätigkeiten im Zusammenhang kann das notwendig werden, wenn sich die Verfügbarkeit von Mitteln im lokalen Kontext so verändert, dass es sich auf die Prozessqualität der Tätigkeiten signifikant auswirkt, sprich, diese deutlich aufwendiger werden, als es während des Konfigurationsprozesses der Fall war. Auf rein *menschlicher Ebene* kann ein Reparaturprozess sinnvoll sein, um durch die Softwarevermittlung entstandene Strukturen, welche auf Softwareebene sinnvoll wirken, an das tatsächliche Leben und dem, was sich darin sinnvoll und richtig anfühlt, anzupassen. Und weiter kann es vorkommen, dass Personen aus aktivierten Commoning-Prozessen wegfallen und niemand anderes die bestimmte Tätigkeit übernehmen kann oder will – in dem Fall ist eine Veränderung der Konfiguration notwendig, um die bestimmte Bedürfnisbefriedigung weiter gewähren zu können.

“When things are first built, the gaps between the parts are often left unwhole. – But these gaps must be healed and made as whole as the parts on either side of them – [...] Slowly, as the »process of repair« repairs the gaps between the wholes, the structure becomes complete and whole at every level.“ (483-485)

An dieser Stelle zur Erinnerung: Die Vision ist eine Gesellschaft, die aus dem Alltagsbewusstsein heraus einfach verstanden werden kann. Bei der jede ausgeführte, gesellschaftliche Tätigkeit gezielt und nachprüfbar reale Bedürfnisse befriedigt. Eine Gesellschaft, deren Strukturen sich den menschlichen Bedürfnissen anpasst, also dem, was individuell als richtig und gut empfunden wird. Und das in jedem Lebensbereich. Eine Struktur, die damit auch nicht starr ist, sondern im stetigen Wandel, wie die Umstände und die Menschen selbst. Eine Gesellschaft, in der es zwar auch Krisen und Engpässe geben mag, an denen aber gemeinsam gearbeitet werden kann, in denen auf Augenhöhe Konflikte gemeinsam gelöst werden. Die Software selbst ist ein Werkzeug – eine Vermittlungsform –, das den Weg in eine solche Welt unterstützen kann. Durch den *Konfigurationsprozess* wird versucht, eine *Effizienz der Bedürfnisbefriedigung* herzustellen, wodurch allerdings Strukturen entstehen können, die den Bedürfnissen der darin Beteiligten widersprechen. Der Reparaturprozess ist daher auch ein Werkzeug, um *gegen die Logik der Software* die Welt den eigenen Bedürfnissen nach zu gestalten.

“When we repair something in this new sense, we assume that we are going to transform it, that new wholes will be born, that, indeed, the entire whole which is being repaired will become a different whole as the result of the repair – In this sense, the idea of repair is creative, dynamic, open.“ (485)

Die Software muss also um die Funktion erweitert werden, dass Konfigurationen manuell und in Absprache mit sämtlichen davon Betroffenen verändert werden kann. Das heißt, es werden neue Tätigkeitsmuster hinzugefügt und andere aus dem Prozess herausgenommen. Ob sich dann direkt in diesen Absprachen Personen den neuen Tätigkeiten zuordnen oder eine Selbstzuord-

THE WAY

"Assume, to start with, that some version of the pattern language has been adopted in a town, or in a neighborhood, or by a group of people or a family who adopt it as the basis for the reconstruction of their world – What is the relation between this common pattern language and the constant process of construction and destruction which gives the town its shape?" (354)

Bisher wissen wir, wie ein Tätigkeitsmuster aussieht, wie es entsteht, dass es eine bestimmte Qualität zur Befriedigung von Bedürfnissen hat und, dass es durch die Anwendung dieser Muster eine Bewegungstendenz gibt, welche die Gesamtzahl der häufig angewendeten Tätigkeitsmuster immer weiter reduziert und solche mit hoher Qualität hervorhebt. Wir wissen auch, dass die Auswahl und Anordnung von bestimmten Tätigkeitsmustern abhängig ist vom lokalen Kontext und, dass eine bestimmte Tätigkeit im Geflecht des ununterbrochenen Commonings nicht nur zur Befriedigung eines bestimmten Bedürfnisses beiträgt, sondern Teil verschiedener Commoning-Prozesse sein kann.

"It's a flux of millions upon millions of these tiny acts, each one in the hands of the person who knows it best, best able to adapt it to the local circumstances. – But what will guarantee it there that this flux, with all its individual acts, will not create a chaos?" (355)

Folgend wird herausgestellt, wie durch den Prozess der *Selbstzuordnung* sinnvolle Re-Produktionsstrukturen konfiguriert werden können, welche sich schließlich manuell den Bedürfnissen der Beteiligten anpassen lassen. Außerdem wird herausgestellt, wie sich aus der Logik dieser Struktur einerseits softwareunabhängige Selbstorganisation ergeben kann, andererseits wie sich bestehende selbstorganisierte Zusammenschlüsse in die Struktur des ununterbrochenen Commonings integrieren können.

Um sich dem anzunähern, muss im Kopf behalten werden, dass *Tätigkeiten* dem Abbau von *Spannungen* dienen und diese *Spannungen* der Aufwand zwischen Bedürfnissen und ihrer Befriedigung sind. Zu diesem Zweck stellen sich sinnvolle Tätigkeiten – insofern sie einmal als Tätigkeitsmuster beschrieben wurden – anhand der *lokal verfügbaren Mittel* heraus. Worum es folgend geht, ist eine solche Spannung schrittweise in einzelne Tätigkeiten zu *differenzieren*.

"The image of the differentiating process is the growth of an embryo. – The unfolding of a design in the mind of its creator, under the influence of language, is just the same. – A language allows you to generate an image of a building in your mind, by placing patterns in space, one pattern at a time" (370-372)

Die Anwendung der Mustersprache ist bei Alexander der *innere Prozess* einzelner Personen, bei welcher sich die Architektur aus der schrittweisen Einteilung durch die Muster ergibt. Die Vorstellung eines Designers von gesellschaftlichen Re-Produktionsprozessen mag dagegen in Teilbereichen sinnvoll sein

An dieser Stelle muss festgehalten werden, dass auch bei über die Software vermittelter, kontinuierlicher Tätigkeit direkte zwischenmenschliche Beziehungen nicht notwendig sind. Sind die konkret verwendeten Mittel (besonders auch die Lokalität der Tätigkeit) und Zeiträume des Prozesses eindeutig definiert, braucht es an sich keinerlei näheren Kontakt zwischen den am Prozess beteiligten Personen.

Was der direkte zwischenmenschliche Kontakt allerdings ermöglicht, sind Absprachen. Sowohl zeitlicher Natur, als auch, ob noch Hilfe benötigt wird oder etwa, an welcher Stelle das zuerst gefertigte Resultat einer kontinuierlichen Tätigkeit am dringendsten gebraucht wird. Weiter wird die Erfahrung ermöglicht, dass es echte Menschen sind, die sich darauf verlassen, dass die eigene Tätigkeit gemacht wird und, dass sie gut gemacht wird. Und direkte zwischenmenschliche Beziehungen ermöglichen direkte Rückmeldung, wenn etwas gut oder mangelhaft gemacht wurde. Weiter ermöglichen direkte zwischenmenschliche Beziehungen, dass auftretende Probleme schnell und unkompliziert geklärt werden können. Statt lange darauf zu warten, dass etwas selbst Benötigtes zur Verfügung gestellt wird, schnell in der entsprechenden Produktionsstätte mit anpacken zu können oder einen Transport selbst zu übernehmen. Und weiter Menschen kennenzulernen, mit denen solche Prozesse gerne zusammen gegangen werden oder schlicht die Eigenheiten von anderen berücksichtigen zu können und dabei auch unvermittelte Fürsorge zu leisten. Direkte zwischenmenschliche Beziehungen sind daher nicht nur hilfreich, sondern auch sinnvoll für die Effizienz des ununterbrochenen Commonings.

Und was die Software elementar hierfür bereitstellen muss, sind Möglichkeiten der *Kommunikation* zwischen den am Commoning Beteiligten. Jede Person innerhalb eines eigenen Commoning-Prozesses muss kontaktiert und jede Person im gleichen Zusammenhang einem solchen Gespräch hinzugefügt werden können. In den dadurch entstehenden Gruppen muss schnell ersichtlich sein, welche Personen welcher der in Zusammenhang stehenden Tätigkeiten nachgeht. Weiter müssen Ergebnisse solcher Kommunikationen transparent für alle Beteiligten des jeweiligen Commoning-Prozesses gemacht werden können.

"Now we come to the actual building of the building. – Again, just as before, the process is sequential. Only now the patterns operate not on a mental image, but on the building itself, as it is being built. Each pattern defines an operation, which helps to differentiate, and to complete, the building as it grows: and when the last patterns are introduced into the growing fabric, the building is complete." (459)

Nachdem ein Commoning-Prozess aktiviert wurde und die teilnehmenden Personen in Verbindung gebracht wurden, folgen die Tätigkeiten einer klaren Reihenfolge: Angefangen von der am „weitesten“ vom Bedürfnis entfernten Ebene werden die Mittel für die nächsthöhere verfügbar gemacht. Sind alle Mittel für eine bestimmte Tätigkeit verfügbar gemacht worden, kann diese einen Bedarf der übergeordneten Tätigkeit decken usw., bis schließlich als letzter Schritt das Bedürfnis selbst befriedigt wird.

Spätestens wenn das letzte an einer kontinuierlichen Tätigkeit anstehende Bedürfnis befriedigt ist, löst sich der Commoning-Prozess wieder auf. Einzelne Tätigkeiten des Prozesses bleiben selbstverständlich bestehen, insofern sie

keitsmuster, das festgelegt werden muss, ist dabei die Tätigkeit, welche das Bedürfnis befriedigt. Anschließend werden die Tätigkeitsmuster festgelegt, welche den Bedarf dafür decken usw. usf.

Das sinnvollste Tätigkeitsmuster ist jeweils das mit der höchsten Qualität und ein Teil dieser Qualität ist die Prozessqualität. Die Prozessqualität entsteht aus der Analyse der im lokalen Kontext verfügbaren Mittel und setzt sich aus dem Aufwand der Tätigkeiten zusammen, die notwendig sind, um durch diese lokal verfügbaren Mittel das anstehende Bedürfnis zu befriedigen. Die Prozessqualität enthält dabei sowohl den Aufwand der Tätigkeit, die im Tätigkeitsmuster beschrieben wird, als auch den Aufwand der Tätigkeiten, die zur Verfügbarmachung des Bedarfes notwendig werden. Das heißt damit einerseits, dass jedes neu festzulegende Muster den Aufwand der gesamten nachfolgenden Konfiguration *enthält* und andererseits, dass sich die Qualität der bereits festgelegten Tätigkeitsmuster während dieses *Konfigurationsprozesses* ständig verändern kann. Denn wenn sich auch durch die Analyse des lokalen Kontextes bestimmte Tätigkeiten zur Bedarfsdeckung als *ideal* herausstellen, dann ist noch nicht gesagt, dass sich auch Personen *diesen* Tätigkeiten zuordnen werden. Ob das der Fall ist, kann allerdings erst festgestellt werden, *nachdem* die Tätigkeit festgelegt wurde. Die Aussage, ob eine bestimmte Tätigkeit im bestimmten lokalen Kontext sinnvoll ist, ist also *spekulativ*. Und diese Spekulation kann nur auf der Grundannahme geschehen, dass nach Festlegung der Tätigkeit sich weitere Beteiligte immer den nachfolgenden jeweils qualitativ höchsten Tätigkeitsmustern zuordnen werden.

Im Konfigurationsprozess wird das Muster mit der (spekulativ) höchsten Qualität zuerst für die Selbstzuordnung *freigeschaltet*. Ordnet sich in einem geregelten zeitlichen Abstand niemand dieser Tätigkeit zu oder gibt es in der lokalen Umgebung keine Person mit den notwendigen Fähigkeiten und einem prinzipiellen Interesse an der Tätigkeit, kann das Tätigkeitsmuster mit der nächsthöchsten Qualität zur Selbstzuordnung *zusätzlich* freigeschaltet werden. Der Prozess wiederholt sich, bis eine Selbstzuordnung stattfindet. Wurde sich einer freigeschalteten Tätigkeit zugeordnet und angegeben, welche Bedarfe selbstständig gedeckt werden können, werden die jeweils qualitativ höchsten Tätigkeitsmuster zur Bedarfsdeckung dieser Tätigkeit ebenfalls freigeschaltet und der Prozess wiederholt sich für jeden Bedarf erneut.

Die Qualität des jeweils übergeordneten Tätigkeitsmusters verändert sich, wenn zur Bedarfsdeckung nicht das jeweils höchste Muster gewählt wurde. Es ist daher möglich, dass ein bereits festgelegtes Muster während des Konfigurationsprozesses so weit an Qualität verliert, dass ein anderes Tätigkeitsmuster auf derselben Ebene spekulativ eine höhere Qualität zur Bedürfnisbefriedigung haben kann. In diesem Fall kann auch dieses Tätigkeitsmuster *zusätzlich* freigeschaltet werden und bildet einen *unabhängigen Strang* im Konfigurationsprozess. Tätigkeiten können von da an für beide Stränge freigeschaltet werden, bis sich ein Strang als eindeutig qualitativ hochwertiger herausstellt und die Freischaltung von Tätigkeiten für den anderen Strang dadurch unter- bzw. abgebrochen wird. Über diese Möglichkeit kann eine Konfiguration durch Selbstzuordnung herausgestellt werden, die im lokalen Kontext weitgehend ideal ist. Es folgt daraus allerdings auch, dass nicht jede Selbstzuordnung zu einer Tätigkeit führt. Für die Beteiligten, welche sich einer bestimmten Tätigkeit zugeordnet haben, muss daher der Prozess und Fortschritt des entsprechenden Konfigurationsprozesses transparent sein.

Durch den Konfigurationsprozess beginnt der Commoning-Prozess, vom Bedürfnis ausgehend, Schritt für Schritt an Form zu gewinnen („each pattern moulds the whole“). Einen Sonderfall bildet dabei die Tätigkeit der *Ortsveränderung*. Nicht im Fall der Ortsveränderung eines bereits verfügbaren Mittels, das der Tätigkeit zugeordnet werden kann, sondern die Ortsveränderung von Mitteln, die erst durch eine andere Tätigkeit verfügbar gemacht werden. Innerhalb der Konfiguration ist sie strukturell näher am Bedürfnis, kann aber erst abgefragt werden – insofern es notwendig ist – *nachdem* die Tätigkeit, welche das Mittel verfügbar macht, und der Ort ihrer Ausführung festgelegt wurden. Die Tätigkeit der Ortsveränderung eines Mittels zwischen zwei aufeinander bezogenen Tätigkeiten wird daher erst freigeschaltet, nachdem die Tätigkeiten festgelegt und ihr Ort bestimmt wurde oder aber, - falls die Ortsveränderung im jeweiligen lokalen Kontext tendenziell unproblematisch ist – erst nach Abschluss der konkreten Tätigkeit, um die Planung des Gesamtprozesses zu vereinfachen. Ob die Tätigkeit einer solchen Ortsveränderung in den Konfigurationsprozess aufgenommen werden soll oder nicht, muss als Softwarefunktion in den Händen der Personen liegen, zwischen denen das Mittel transportiert werden muss.

Der Konfigurationsprozess zu einer bestimmten Bedürfnisbefriedigung ist abgeschlossen, wenn der Bedarf jeder Tätigkeit durch eine andere Tätigkeit gedeckt werden kann.

"When the order of the patterns in the language is correct, the differentiating process allows the design to unfold as smoothly as an opening flower." (384)

Der beschriebene Auswahlprozess findet *für jedes einzelne Bedürfnis* statt. Bedürfnisse *derselben Art* bündeln sich damit nicht automatisch, sondern werden erst miteinander verbunden, wenn im lokalen Kontext ihrer jeweiligen Vermittlung dasselbe Tätigkeitsmuster freigeschaltet wird. Ganze Commoning-Prozesse werden somit – durch den Verweis auf dasselbe Tätigkeitsmuster im selben lokalen Kontext – miteinander verschlossen. Dass zwei oder mehr Bedürfnisse dadurch auf dieselben Tätigkeiten verweisen, erhöht das *Gewicht* dieser Tätigkeiten. Je höher das Gewicht einer Tätigkeit, desto sinnvoller ist damit die Tätigkeit für den Gesamtprozess und desto sinnvoller kann die Tätigkeit auch individuell sein, wenn das Gewicht der ausgeführten Tätigkeit mit dem Gewicht der eigenen Bedürfnisse gekoppelt ist.

Ebenso wie die Qualität einer Tätigkeit während des Konfigurationsprozesses, ist das Gewicht einer Tätigkeit bei der Selbstzuordnung *spekulativ* und davon abhängig, wie viele darauf verweisende Commoning-Prozesse *nach* der Selbstzuordnung tatsächlich aktiviert werden. Weiter ist hier zu beachten, dass das gesamte Gewicht nur bei Mitteln vollständig „abgebaut“ wird, welche sich durch Teilen vermehren (oft soziale und symbolische Mittel) bzw. welche im lokalen Kontext seriell genutzt werden können. Bei gegenständlichen Mitteln, die sich in der Bedürfnisbefriedigung aufbrauchen, ist eine wiederholte, sprich: *kontinuierliche* Tätigkeit notwendig. Als Softwarefunktion braucht es hier die Möglichkeit, sich nur einer bestimmten *Menge* der darauf verweisenden Prozesse anzunehmen.

Auch wenn für einen einzelnen Commoning-Prozess das jeweils qualitativ höchste Tätigkeitsmuster unbedingt am sinnvollsten ist, kann es für den Gesamtprozess effektiver sein, wenn Tätigkeitsmuster gewählt werden, die zwar eine niedrigere Qualität haben, dafür aber in mehr Prozesse integriert werden