Chapter 14 – Designing Complex, Adaptive, and Evolvable Systems

Abstract

The Purpose of this Chapter

The word “design” used in this chapter is meant to cover the broad spectrum of thought activities that result in a functional design of an artifact of arbitrary complexity. The word as typically used within the systems engineering community often refers to the act of specifying components and their layouts to implement the artifact. The word “architecture” and the gerund “architecting” are used to distinguish the act of conceptualizing and specifying general properties of the artifact as opposed to the detailed design work. What we are interested in in this chapter is the process by which the results of a detailed deep systems analysis as described in Chapter 5, and having captured the knowledge in the knowledgebase described in Chapter 7, are the basis for the design of complex artifacts and artifactual systems as covered in the previous chapter.

Structured, deep systems analysis gives us “Design for Free”, with respect to existing systems that are to be “automated.” Designing new systems can profit from the same basic process. In this case the systems analysis is conceptual and virtual. That is, we are not analyzing an existing system, but one that we intend to bring into existence. The system to be designed exists in the abstract. So, also, is the case for modifying an existing system. In this case we combine the analysis of an existing system and then extend the analysis in the virtual domain as we project the modifications to be made. In other words the methods provided in this volume can be used globally in the evolution of artifacts and artifactual systems.

We showed in Chapter 8 the analysis of an existing system, the economic subsystem of the HSS, and quickly found problems with its design. Those problems were shown to arise as the result of an immature intertwined natural evolution and intentional evolution. That is, humans made choices about how an economic process should work based on both what had evolved so far as a result of natural selection (e.g. trade of goods) and what they thought should be involved in the process (e.g. the use of money).

The way in which our cultural artifacts come into being is a multi-step process. First someone conceives of a way to solve some problem by having a ‘thing[1]’ that works to do so. Next, they envision what that ‘thing’ will look like and how it will work in practice. This is what

---

[1] Actually, the word ‘thing’ here doesn’t just apply to physical objects. Developing procedures or a set of steps to follow to accomplish a goal, is also a design-engineer process.

we mean by design. This will be the subject of the present chapter. The next steps involve the actual working out of the details of the 'thing.' This was, historically, an empirical process in which one would try to construct something and see if it worked the way intended. Usually that process would require multiple iterations to get to an acceptable level of performance. Over history, however, we have learned a lot about transforming designs into implementations. We've learned how to 'engineer' a thing, that is, we can elaborate the design on paper (or in a computer tool) and create a set of specifications that when implemented provide a more sophisticated 'thing.' Engineering is a structured, formal, and principled method of converting a design into a working 'thing.'

Our focus, however, will be on the design of CAESs as covered in Chapter 9. Systems engineering is increasingly called upon to tackle extremely complex problems and design extremely complex systems to solve those problems. In fact, the whole conceptualization of complex problems has evolved as more experience has been gained, particularly in the realm of techno-socio-systems, those that involve human components. The fact that humans learn new things, and think new thoughts, therefore being evolvable systems in their own right, means that we need to bring evolutionary principles directly into the design arena. This makes systems engineering very different from the traditional approach to design and engineering.

The approach we will use is to base our designs on the archetype models of Part 3, chapters 9-12. This is not a new approach within the engineering profession. But the term, 'model-based design,' has two senses. One is that a design is done by building models first[2]. The other is that designs should be based on general models of the category or kind of thing being designed. This latter sense is sometimes described as using 'design patterns,' re-usable patterns that have been used to solve problems previously[3]. The methods we describe in this chapter employ both senses. The archetype models presented in Part 3 are generalized patterns of subsystems found in all CAESs and we will be using these as the core design models for the design of CAES artifacts.

Note that this approach is actually the same one used by our brains in constructing models of systems in the world. Recall from Chapter 3 that our brains come pre-wired with systemese, including notions of agents and agency, and a built-in hierarchical cybernetic governance system that allows us to interact with the things in the world that we have constructed models of in our heads. We construct elaborate models by reusing elements of systemese and recursively combining them into complex concepts. Recall, also, that this process is both experimental and intentional. We build theories of what a thing is and how it behaves and then test the theory against experience. Gradually a veridical concept serves us to 'predict' or 'anticipate' how our

---

[2] The 'official' model-based design (MBD) has a rather narrow set of systems on which it is applied. See the Wikipedia article: https://en.wikipedia.org/wiki/Model-based_design for background. In this chapter we are generalizing the notion of designing using models to all complex artifacts, not just control systems. Accessed 9/17/2019.

[3] See the Wikipedia article: https://en.wikipedia.org/wiki/Design_pattern for background. Accessed 9/17/2019.

interactions with the thing will go in the future. However, our concepts are subject to changes based on changes in the world. That is, our concepts can evolve as a response to how the world around us is evolving. Our brains are evolvable systems.

Thus, in our formal engineering process we incorporate evolvability directly into our modeling process as well as into the designs of the system components themselves.

Concepts of Systems Design

We human beings are in a remarkable position in terms of being biological entities. We are *intentional* beings. That is, we are able to develop plans of action to accomplish some end that we desire. We are clever and able to create instruments and machines that are able to affect our interactions (as a species) with our overall environment – the Ecos. For most of our history that has been instantiated by the invention of very small-scale machines and procedures that have been targeted to solve fairly constrained problems. We needed to turn dirt so as to plant grains so we invented the plow. We needed to get from point A to point B in a short time so we invented the wheel and the cart and harnessed oxen or horses to pull them. More recently we have mastered internal combustion engines capable of moving a large amount of mass quickly between those points. We have gotten more powerful, faster, and seemingly better off as we have conquered the physical world.

Intentional invention is a complex process. It involves complex affordances. It involves complex intentions – wanting some kind of system capable of solving what we conceive as a problem. That is our nature. We pursue a desire. We want to be able to accomplish some goal and we go about the process of finding a way to do so. We design technological solutions.

Stone Age humans discovered how to work stone to produce refined sharp edges and hard points for spears and arrows that improved their capabilities to hunt meat. They worked out how to use animal skins to protect their bodies as they moved into colder climates. They learned how to construct shelters to protect them from the elements and predation. This is our heritage.

Today, we invent new ways of processing data. We know how to master communications of information. We know how to build skyscrapers, how to build freeways. We have mastered an ability to transform our intentions into actual physical things that seem to solve our perceived problems. The process is intentional design. And we have become very good at it.

Why we intend to have an artifact (and also improve on an existing artifact) is that we perceive that some kind of ‘problem’ or an ‘opportunity’ exists that such an artifact would ‘solve’ or ‘facilitate.’ That is, we conceive of an artifact that has a purpose.

Purpose of an Artifact

All CAS/CAESs have purposes, that is they perform functions that benefit humans in some way. That said, it is important to recognize that purposes are also generally fairly complex in their own right. For a simple example, one could say the purpose of a modern commercial airplane is to transport people from one airport to another. That is certainly true but very insufficient as a description. The transportation act needs to be safe and comfortable. The plane's design has to take into account those factors that make the trip sufficiently pleasant for the passengers so that they will be repeat customers.

Much more complex artifacts, and by extension, systems of artifacts, will generally fulfill multiple entangled purposes. Herein, immediately, is a potential problem. Some artifacts may end up trying to serve antagonistic purposes. One of the best examples of this is a corporation that is created (designed) to "produce the very best quality product of its kind." But it is also designed to sell this product at a profit. Most often those two purposes need not conflict so long as the cost of production is less than the price the market will bear. But a problem arises when the profits made are meant to maximize shareholder wealth. Investors expect high returns on their investments and signal their expectations through stock purchases and stock prices. The management of the company is under pressure to maximize profits and that starts to come into conflict with the production of the best product when cutting costs means reducing quality (e.g. buying cheaper components)[4].

At the time of this writing there has been a tragic example of components in a complex system working at odds with one another. In theory one component, a computer program, was designed to compensate for a tendency for a commercial airplane, the design of which was retrofitted with a heavier engine, to climb at too steep an angle on takeoff. The computer was supposed to push the nose down without the pilots' knowing what was happening. In two instances the pilots may have reacted wrongly causing the airliners to crash. In this case three components of the whole system, the engine (weight), the computer, and the pilots all were trying to achieve conflicting ends and under some set of conditions it resulted in unhappy ends.

Architecture vs. Design

Though we think of the whole artifact creation process as

Design by Evolution

Not all of human invention is due to intentional design alone, however. Indeed, much of what humans do as they develop technology is to learn from past mistakes or shortcomings in their intentional designs precisely because those designs (analogous to genotypes) and their implementations (analogous to phenotypes) have either failed to live up to expectations (analogous to selection) or gave clues to improvements in design that would increase, for example, the efficiency of actions facilitated by the designed thing. The history of the steam

---

[4] There are multiple factors at work in modern corporations in addition to maximizing shareholder wealth, of course. For example, management is often motivated to keep stock prices high because the company's stocks are part of the manager's pay package.

engine is a case study of this[5]. That history not only provides a beautiful lesson in how a form of intentional selection, which some would argue is just another kind of 'natural' selection, leads to improvements in form and function of machines reminiscent of improvements in organisms' fitness in their environments. Designs are selected for by how well they serve human intentions.

On occasion, Role of serendipity and insight to employing new physical phenomena

Design by a Hybrid Process

The first thing that we should be humble enough to recognize is that we always start out in ignorance when approaching invention. Edison's invention of the light bulb is instructive. What we have initially is a desired function we want to have implemented. What we also might have is a few existing designs that, maybe, if put together in the right way, would get an approximation of that function. Affordance is always a key to envisioning or conceptualizing what could be.

Those existing pieces are suggestive of possibilities, but we may not exactly see how they might need to be modified and integrated to produce what we imagine would be the final configuration.

All of human generated designs are actually a hybrid of intentional design (what we want to have happen) and evolutionary design (incremental improvements with occasional leaps of integration due to either serendipity or mental models of more creative individuals). Ignorance of what is ultimately possible means that we are most generally constrained to empirical methods to find better designs. In the case of the steam engine, once we grasped the basic concepts of heat flow and efficiency, we strove to improve our machining of cylinders and pistons along with heat exchangers to achieve greater efficiency. More efficient steam engines led to more power per unit of fuel being delivered to the work process, and, from an economic perspective, more profit!

Throughout the previous stages of the industrial, post-industrial, and information revolutions the design of new and improved machines has followed this empirical approach. Incremental improvements driven by intentional objectives have led to increasingly capable machines (as well as increasingly effective procedures). But something else has been evolving in our overall cultural strategy of using technologies to improve life. Our machines have been getting more sophisticated, more complex, and more interactive without human intervention. In other words, our artifacts are becoming complex (and even sometimes adaptive) systems in their own rights. Today we talk of "cyber-physical" systems or extremely complex machines comprised of multiple complicated mechanical subsystems along with very complex autonomous decision agents (computers running artificial intelligence algorithms and with sensory-actuator interfaces). We can no longer produce the kinds of machines we want strictly from the efforts of

---

[5] See the Wikipedia article: https://en.wikipedia.org/wiki/History_of_the_steam_engine for more background. Accessed 4/5/2018.

mechanical or electrical or computer engineers. We need for them all to work together to produce integrated designs from which whole new behaviors emerge. This is something new for technology.

Engineering Complex Systems

Systems engineering requires a whole new approach to engineering in general. It is no longer possible to just learn one kind of application of the laws of physics, say mechanics and thermodynamics, to the design of machines (as was the case with the automotive industry before the late 1970s). Modern machines now incorporate engineering from multiple (historically separate) disciplines and their efforts have to be carefully integrated. Consider, again, the automobile. Long gone is the mechanical carburetor that supplied an approximately good mixture of vaporized gasoline and air mixture to the cylinders. Today we have fuel injectors (electrically activated devices) controlled by computers that take into account a multiplicity of signals such as temperature of the air, command for acceleration, engine load, and so on, to much more precisely mix fuel and air for maximum efficiency. Thus, computer engineers and computer scientists, physicists, and perhaps several other specialists (e.g. materials scientists) have to engage in a collective process of design. This is a challenge, especially when our education system is very slow to recognize the need for a governance structure able to implement the management of a multiple work process system. They still produce specialist engineers.

That is starting to change, of course. There has been a recognized engineering field for systems since the late 1950s. The whole American and former Soviet Union space programs, for example, were based on concepts of systems engineering as it was understood at that time. Over the last several decades the need for another supra-layer of design and engineering capable of working with diverse modular components, ensuring that all the various kinds of parts work together has become evident. Systems engineers have largely had to invent their own discipline on the fly – more in response to increasing complexity of machines than to foresight. It has only been in the last few years that, as a profession, systems engineers have begun to recognize that, like all of the other engineering professions, they need their discipline to be based on firm scientific principles that can guide their work. As things stand of this writing, there has been only the beginnings of trying to find those principles. This book is one attempt to address that issue.

Pattern/Model-based Design

Over the last several centuries we humans have learned to catalog and categorized a number of patterns of design that we are able to use repeatedly with variations to produce new solutions. This is also proving the case for complex systems. Throughout this book you have seen examples of patterns, such as the economic (related to metabolism) and governance subsystems, that might provide reasonable templates for designs in actual systems. Every CAES, it has been argued, needs an internal metabolism/physiology and processes for interacting with its environment (an economy) and a governance subsystem in order to be sustainable for the long-haul.

Systems engineering is learning more and more about patterns or generic models found in multiple kinds of natural and manmade systems. As our ability to characterize these patterns/models improves (and that means as our language for describing them improves as suggested in Chapter 3) our ability to use those descriptions as starting points for subsystem or even whole system designs becomes potent.

Two particularly relevant developments in the systems science domain are also worth noting. One is the conceptualization of what Tyler Volk (1995) has called ‘metapatterns.’ These are highly generalized patterns that can be found throughout nature, such as a sphere, a tube and barriers, among others. He provides examples of these patterned entities from the subatomic to the whole Earth system and how they are fashioned. This view may provide additional strength to the arguments advanced in Chapter 3 regarding the lexicon of systemese. For example, he describes barriers involved in the role of boundaries, including some of the same ideas regarding how ‘things’ get into and out of barriers (like interfaces). In describing how these metapatterns come into existence in specific instances like cell membranes and social bonding he is also addressing the issue of ontogenesis[6].

The second is the work of Lenard Troncale et al. mentioned in Chapter 2 – the existence of what he calls isomorphies or very particular patterns that are found variously throughout nature, including human systems. Between the system language of Chapter 3, the concepts of CAES archetypes in Part 3, the metapatterns of Volk, and the isomorphies of Troncale we have a rich set of models to work with in building models of real systems[7].

Model/pattern-based design is now an extremely active area of research in systems engineering academia. It is recognized that nature often chooses to solve the same kind of problem by using the same architectural solutions regardless of the substrate of the system. We saw this with metabolism and economics outlined in Chapter 9 and further developed in Part 3. Or, at least we saw the possibilities for it in Chapter 8. In this chapter we intend to demonstrate the next step in using the model of metabolism as a starting point for designing a sustainable economy for the HSS. In the next chapter we will extend that effort by translating the concepts of that design into engineering specifications of realizable subsystems that could serve the existing HSS to achieve a sustainable future.

The second sense of model-based design is the use of abstract models (usually computationally) to simulate the system to be built and test variations on designs variables. It has become de rigueur to build models first to determine that a specified design will ‘work.’ We will have more to say about this below.

---

[6] In Volk’s (2017,) “Quarks to Culture,” these themes are revisited in the form of ‘combogenesis’.

[7] The author, Volk, and Troncale have recently started a project to reconcile our three very similar models of system ontogenesis. That will be followed by reconciliation and integration of our three concepts of universal patterns.

Generic and Archetype Models

Len Troncale and associates have examined and categorized over 50 low-level patterns that are isomorphic across many kinds of systems at multiple scales of space and time. In chapters 2 & 3 of this volume we identified similar subsystem components again that are found at all scales of space and time as well as across levels of organization. These elements form the basis for all higher-order complex systems.

What systems science is adding to this notion of design is that there are some generic (or universal) patterns (Volk, 1995), models that recur over and over throughout nature. In Chapter 8 and Chapter 11 we saw important examples of this. An economy is a recurring theme in complex adaptive systems. So is a governance subsystem.

Part 3 provided the breakdown of a generic model (archetype) of a CAES into three sub-models, an economy, governance, and the agents that constitute the decision nodes in complex hierarchical networks. We saw that an economy is where the main work processes convert resource materials into products with some wastes. An economy uses high-potential energy to make these transformations. We saw the economy must be governed beyond what market mechanisms alone can supposedly do, so a governance subsystem is needed. We saw that both economic processes and governance processes are, themselves, under the care of decision agents that process information about the state of processes and provide control signals to keep those processes operating in a nominal range. We saw how the governance architecture, based on decision types and time scales, provides operation-level management of each individual process, coordination-level management in the form of logistical management to keep disparate and dispersed work processes working together and tactical management to coordinate the whole system with external entities, and in the case of CAESs, strategic-level management to address possible evolutionary changes that might be needed to accommodate larger changes in the environment that would potentially affect the system in the future.

We also examined the various peripheral processes involved adaptive response and evolutionary change.

These models were claimed and shown by examples to be universal. Thus we can assert that any future artifact or artifactual system will also be based on these models.

The CAES model is the starting point for designing complex systems. That is, using the CAES archetype as the base model, the designer begins the process of top down analysis as was shown in Chapter 5, demonstrated in Chapter 6, and used to do a deep analysis of the energy sector of the HSS economy in Chapter 8. The process of design of a CAES is, in essence, no different from the analysis of an existing system. The CAES archetype is mapped onto the SOI and, starting with the boundary/environment analysis (ignoring the internals for the time being) determine the outputs to be generated by the SOI, that is all products and wastes and all of the sinks associated. Note that this is exactly what systems architects/engineers do today in the form identifying the “stakeholders” or recipients of the output flows. This is where the designer considers the purpose of the SOI in terms of how the product outputs serve the recipient sink entities, and considers how their benefiting serves the larger supra-system.

Similarly, the designer knows in advance because of the architecture of a generic CAES that it will be necessary to identify all of the relevant inputs and their sources. These are already known in their generic forms as material, energy, and message flows.

Once the boundary of the SOI has been determined, the procedure is to follow backward from the output interfaces the flows from internal work processes that produced those flows.

Extremely complex CAESs are found to be a "system of systems"[8]. That is the target CAES (SOI) is generally a system that is comprised, for at least level 1 subsystems, of other CAESs. So, the internal components at level 1 can themselves be modeled using the generic CAES as a starting point.

Figure 9.2, recall, shows all of the component types to be found in a CAES. To design a CAES the designer takes a list of all of these, as in Table 14.1 and reserves slots in the knowledgebase. If the design is to be for an adaptive system, not requiring evolution, then the strategic management and evolvability components may be left out. However, it is probably wiser to consider a larger context for the system in which it will 'evolve' in future generations.

Table 14.1. The list of major CAES component subsystems.

Work Process Network	
	Acquisition processes (under cooperative logistic and tactical management)
	Export processes (under cooperative logistic and tactical management)
	Work processes (under cooperative logistic and operational management)
	Value-added flows
Management Processes Network	
	Logistical Management
	Tactical Management
	Strategic Management (where applicable)
Peripheral Processes	
	Adaptation processes
	Evolvability processes (where applicable)

---

[8] See the Wikipedia article: https://en.wikipedia.org/wiki/System_of_systems for background. Accessed 10/20/2019.

	Repair & Maintenance processes
	Recycle processes

The level 1 decomposition will take each of these and consider how they are to be composed of subsystems (many being, themselves, CAESs). For example, as shown in Figure 9.2, all of the processes are shown with their own operational management sub-processes. Generally speaking, at this level in a system of systems, this will be the case.

Process of System Design

In this section we will outline the process for designing a CAES-model system, in preparation for what is to be presented in the next chapter, the systems design of society. In the next chapter we will be introducing the modularity principle applied to social units. Many modules at the lower levels of organization will be essentially identical across regions, however some modules, as we will show here, may have some specialization due to factors like resource type distributions or having craft persons in particular trades.

Let us imagine a situation in which there are just two modules that are situated relatively close together. One module occupies a prime food producing area, and we'll assume a reasonably stable (though likely different) climate. The other module cannot produce all of the food it needs but has access to metals and quality clays so is capable of manufacturing various artifacts, including those needed for farming. Thus, the food-producing module supplies its excess food production to the manufacturing module and the latter supplies implements and clay containers to the former. In this section we will focus on the design process itself and then in the following section we will apply the process to the food-producing module.

CAES Archetype as Template

The design of an artifactual system such as a food-producing community presupposes that the system will be a CAES. It will have a basic internal economy, a governance subsystem, and include provisions for adaptivity and evolvability. We will therefore proceed with using a CAES archetype model as a template for this design. Using the elements (subsystems) given in Chapter 9 we will erect a scaffold and placeholders for each of those elements. Below we will show one way to parse the whole system.

Start with the Economy

The economic subsystem is where the export products will be produced. As recommended in Chapter 5 and amplified below, we start our analysis with the outputs of the system. But the economy is also where the material and energy inputs are imported, so we will have nearly covered the boundary set with this approach. What will not yet be covered are the message inputs and outputs, for example messages received by an acquisition process.

It is always a good approach to start with the internal economic subsystem since this is where the material work is accomplished and the work flow needs to be deeply understood before it is determined what sort of coordination management is going to be needed. Figure 14.1 depicts the essential parts of an economy subsystem (extracted from Figure 9.2) starting with resource inputs and ending with product outputs. In the center we represent a single generic work process (giving off waste heat and waste materials). Each of ovals inside the boundary is a starting placeholder for its particular purpose in the whole system.

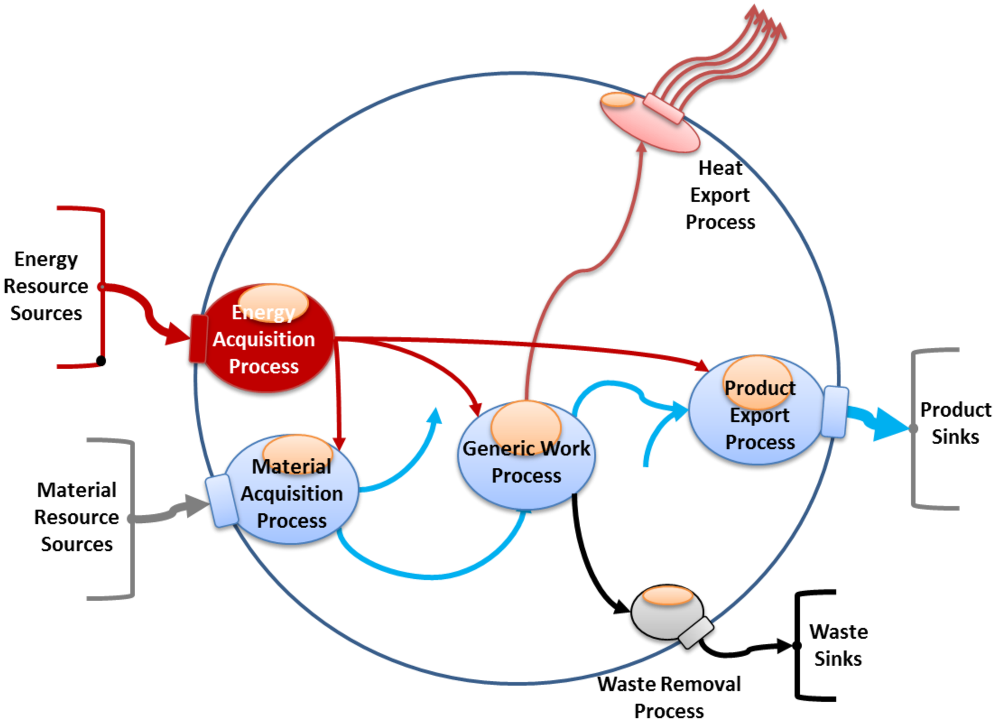

Fig. 14.1. A CAES economy archetype.

In all of the ovals, we see an inner orange oval that represents the local operational management agency. As per Chapter 11, these agents monitor the activity in the work process, making adjustments when needed, and report to a coordination agent. Also shown are the sources of resources and the sinks for products and wastes, again as in Chapter 9. Additionally, to emphasize their importance, we have shown the major interface subsystems interacting with the external environment.

Using this template, we use the methods of Chapter 5 to analyze and decompose each of these template objects. All of these are absolutely necessary to produce a complete economic subsystem for the CAES. The only difference between the analysis of an existing system and the analysis of a to-be-designed system is that the system is virtual in the latter case and the designer(s) determine what is to be present.

All CAESs need to have high-potential energy acquisition processes, here represented by a single (red) oval. Over the course of the analysis of work that takes place in the economy, it may become evident that multiple energy sources are needed. Chapter 5 methods, you may recall, provided for adding subsystems as needed.

All CAESs also need to acquire medium-entropy material resources, so we put in a placeholder for such process (blue oval with an interface for receiving material). As with the energy inputs, if we discover later that multiple types of material inputs are needed, they can readily be added.

Finally, all CAESs need to expel waste materials and export waste heat to the environment. We put in placeholder processes representing these as well. We will not worry about these further; this exercise is just to demonstrate how the model provides guidance in design.

As per the guidelines in Chapter 5 we start with the outputs, particularly the product. In actual CAESs there will generally be multiple products (or services) exported, so the product sink may be replicated for each one as will the product export process.

Once the products and waste exports have been determined and characterized, we can proceed to apply the deep analysis process to the generic work process, splitting it into as many level 1 processes as will be involved in producing the products. Once they have been delineated the analysis of their inputs will tell what resource inputs need to be imported, so we apply the same logic to the acquisition processes, both material and energy. Not shown in the figure is the various communications channels that will be needed for the import/export processes to coordinate with their respective sources/sinks.

Develop the Governance Subsystem

Once the details of the economic subsystem are worked out, we can begin consideration of the governance subsystem needed to manage that economy. This, at minimum, means adding coordination-level agency for both tactical and logistical decisions. Recall that the purpose of logistical coordination is to ensure that all of the work processes work together and that there is a smooth flow of adding value to products as well as ensuring completion rates.

Figure 14.2 shows the addition of coordination level managers and some of the communications channels that allow them to regulate the work processes and cooperate with one another with respect to coordinating the flows of inputs from acquisition process and to export processes. We include an adaptation process that is capable of operating on, in this example, a material acquisition process which is being disturbed by an environmental factor, e.g. disruption to the flow of input from a source.

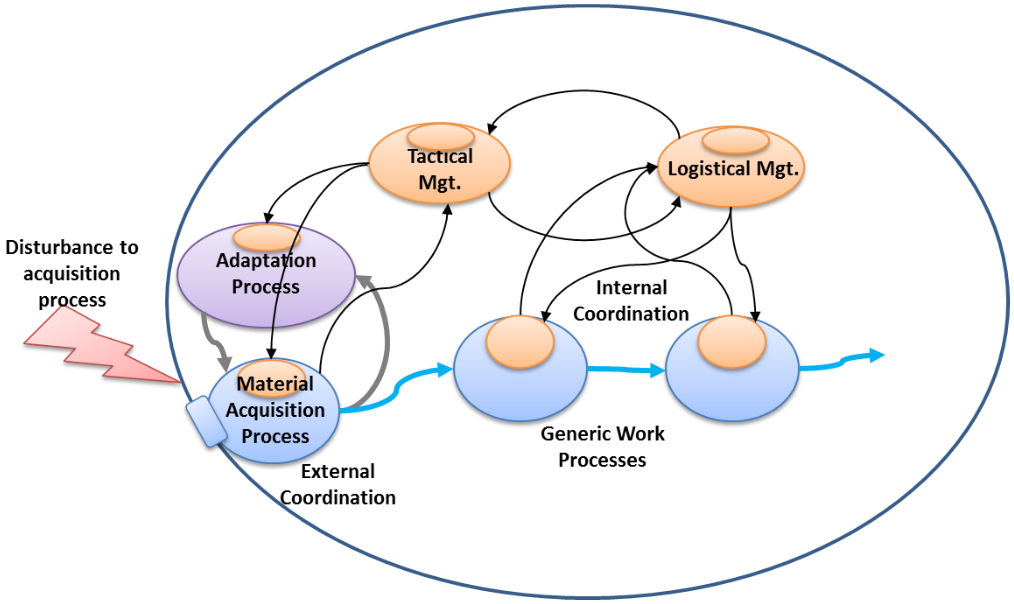

Fig. 14.2. A CAES governance archetype (without strategic management).

As with the analysis of the work processes themselves in the economic subsystem, the needs for management proceeds to determine the governance architecture, that is, the number of levels of management needed to manage the complexity of the system. The guiding principle is just that. The greater the structural complexity of the economy, the more coordinators are going to be needed and that means constructing a hierarchy of management (recall Figure 11.11).

Develop the Mechanisms of Evolvability

Evolvability can be achieved in two ways. In systems such as an ecosystem or even the Earth as a whole, where there is no explicit strategic management, evolution comes about essentially in the Darwinian way, through mutations of components followed by selection from the environment. Ecosystems are evolvable through either the evolution of member species or by invasion of new species that compete with prior members.

In intentional systems, human-based organizations, societies, and so on, evolvability is accomplished with intentional modifications to components in response to some alteration in the environment. This is the strategic management of such a system covered in Chapter 11. Of course, accidental modifications in components of a system can still happen and cause the system to function differently. As with genetic mutations, most of these will prove detrimental or at least reduce the efficiency/effectiveness of whatever subsystem the modification takes place. For example, when an employee tasked with a routine procedure decides to take shortcuts without realizing the consequences. They can go on for some time without anyone noticing that something subtle but vital to the health of the organization is missing. This sort of degradation of processes is, unfortunately, quite common. And the larger the organization the more it seems to happen. But every once in a while, an improvement in a process is discovered (an employee has a good idea) that boosts the overall efficiency/effectiveness.

Designing a non-intentional CAES for evolvability requires considerable anticipatory strategic thinking by the designers. The system is intended to operate autonomously and survive changes that might occur in its environment. Imagine a mobile, autonomous mining system operating on, say, Mars. The system would need to be at least adaptable; Mars has seasons. But if the system is to be long-lived it may need some capacity to evolve, say as when it discovers some new kind of ore that needs different extraction or processing techniques[9]. It would require a capability for modifying its current mechanisms. We will not dwell on the design of non-intentional CAESs. One way to think about how this is achieved is to consider the strategic management subsystem shown in Figure 14.3 below as being 'outside' the system. Our main concern now is the design of intentional systems, socio-technical artifactual systems that involve humans filling the roles of agents (or at least most of them). In this figure we show the situation when humans are in the loop and can provide strategic management.

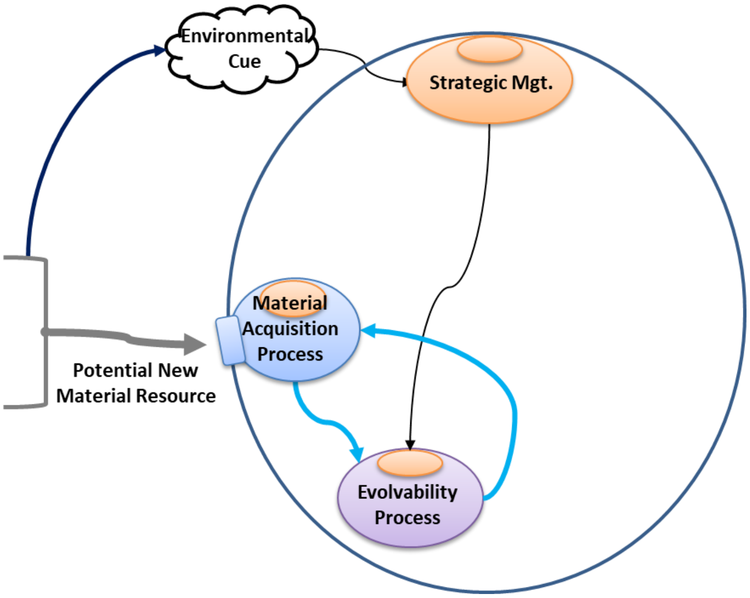

Fig. 14.3. Evolvable CAES with strategic management.

Strategic evolution requires the inclusion in the design of processes that are able to alter other processes. As shown in the figure a new material resource has been identified by the strategic agent from cues in the environment. But the current material acquisition process is not

---

[9] Least the reader thinks this example might be a stretch, please realize that the design of a true CAES has not yet been attempted. We don't really have any examples of making a non-intentional system evolvable! Putting humans in the loop (making an intentional system) provides the capacity to evolve the system in which they are embedded. The point of discussion on non-intentional but evolvable systems is to demonstrate that it is, at least in principle, possible.

designed to import that material. This requires that the process be altered so that it may import the resource. The evolvability process is a peripheral subsystem that has the capacity to operate on other processes. This is represented by the blue arrows, one from the material acquisition process to the evolvability process as the latter takes the existing former in as an input, does work on it to implement the modification, and then outputs the modified process.

Evolvability processes like this may have themselves evolved from earlier repair and maintenance processes discussed in Chapter 9. Indeed, the adaptation processes (in Figure 14.2) may have had the same origins. Repair and maintenance processes, such as autopoiesis in living systems, require there be tools and materials in reserve that can be brought to bear when needed. Many of the same mechanisms used to repair, say a machine, could also be used to build a new kind of machine.

How Do We Design Complex Systems?

Conceptualization

After recognizing the nature of a complex (generally wicked) problem and understanding the context or environment in which the problem exists the designer sets about considering ways in which the problem can be solved, what sort of artifactual system might address or resolve the problem. Using the CAS/CAES archetype model helps with this activity as it is a starting place for conceptualizing the solution. We start with the ‘product’ outputs (and this means services as well). These are the outputs that interact with the entities in the environment that are parts of the problem(s) so as to change the supra-system’s behavior and reduce or eliminate the problem(s).

The designer needs to take into account concepts such as time (duration), how the environment might change over time, how those changes might affect the artifactual system, and so on. These need early consideration as they will affect how the CAS/CAES will be expanded during the analysis phase. They will determine how the subsystems need to be identified along with their relations with one another.

Architecture

Table 14.1, above, provides a rough list of the architectural features that will go into the design. Figure 9.2 (Chapter 9) provides the general layout for all of these components and their flow relations. The conceptual model of the system should have resolved the basic questions about how and how much the environment may change over the life of the system. This will determine the inclusion of the adaptivity and evolvability processes. If it is determined that the environment will remain reasonably stable over the life span of the system, then it is likely not necessary to include explicit evolvability processes. For example, in a manufacturing facility that is expected to produce the same product throughout its lifetime, provisions in the design might include adaptive capabilities to adjust for production rates if demand for the product increases or decreases. But if it is anticipated that other products might be produced in the future then the facility would include an internal design capability as well as an expanded machine shop that could design and build whatever additional equipment would be needed. In addition, the upper management of this facility would need to have all of the capabilities of a strategic management process in place to get information from the environment about possible new customers for new products and provide the tactical and logistical management layers with directions on what needs to be done to evolve the organization so as to fill that future need.

The general architecture of the system would then start with a Figure 9.2 layout giving names to the high-level processes pertinent to the particular system.

Sub-module Specification

Figure 9.2 is a very abstract representation of the CAS or CAES artifactual system to be designed. It does provide a way to focus on the different functions that need to be considered. The next step is to use the same methods described in chapters 4, 5, and 7 and above in section Start with the Economy to analyze each of the functional modules in the figure based on what they will actually do in the context of the whole system and its environment. The recursive procedure described in Chapter 5 will then allow the designers to analyze the various sub-modules down to lower levels in the organization hierarchy. This process of downward analysis produces the “design-for-free”. At the leaf nodes of the system tree, the designers are ready to develop specifications for what and how each leaf-node module should function. For the artifact portions of the system this would resemble classical engineering and require traditional specialist engineers to write specifications for the machinery, as it were.

Since we are talking about an artifactual system we need to also take into consideration the way that human beings fit into the scheme, i.e. specifying their role as sub-modules. But this is actually already done in principle because we have already identified the human role in decision-making, being agents and having agency in governance and economic subsystems. Humans are needed where automated systems are still inadequate for filling this role. These days, with advances in cyber-physical systems that make them able to handle most low-level decision processes, humans would tend to fill the logistical and tactical as well as strategic management levels. The point of analysis of all agents, cyber-physical or human, would be to understand the decision requirements and provide an adequate decision model along with the degrees of freedom or requisite variety needed to have effective agency.

If the procedures outlined in Chapter 5 are followed rigorously, and the capture of data needed to fill the knowledgebase ala Chapter 7 is seen to, then at the end of analysis the designers have enough knowledge of the system and its environment to generate functional models for the next step.

Simulation

The knowledgebase contains both the structural and functional aspects of a system. From this knowledge the designer may generate a simulation model at, in principle, any resolution and for any portion of a system. That is, the designer may call for a simulation of any subsystem at any level of the hierarchy. This is possible because the structure of the knowledgebase, derived from Equation 3.1 and its recursive use, contains not just the knowledge of the subsystems but also the knowledge of the dynamical behavior of the sources and sinks for any subsystem. Within a system level, for any subsystem chosen, all other subsystems at that level are treated as environmental entities as sources and sinks.

The simulation models are similar in nature to a systems dynamics (SD) model except that the systems are treated as bounded (modular) processes. And their subsystems are internal bounded processes, so that a simulation of a system is a recursive simulation of lower modules each at their level's appropriate time scale. At least, again, in principle, a simulation of the entire system, along with the behaviors of its environmental entities is accomplished by recursively calling the simulations of its subsystems and they of their sub-subsystems down to the last internal node of the system tree. We say 'in principle' because the computational power needed for such simulations would be immense for extremely complex CAESs. However, as an aid to the design process and to test design operations, simulation of the pieces in a bottom up manner should help resolve issues about input/output designs (interfaces) and overall dynamical behavior of the modules. This might seem to be a retreat to reductionist approaches, but it is not. The designs fell out of a top down holistic analysis process which retained the information regarding relations and interactions of all of the modules, so the running of simulations of those modules from the bottom up is consistent with a systems approach.

We suspect that as systems design and engineering become more endowed with experience in the design approach we advocate here, designers and engineers will begin adopting so-called Big Data and distributed computing methods to apply to this approach of simulation. Since the designs are modular as a result of following the analysis methods of Chapter 5, they are naturally represented in distributed simulations.

Alter Environment

Iterate as needed.

The Food-Producing Module

Agri-Village

In the context of social modules (units) to be presented in the next chapter we consider the unit we call a "community," which is an artifactual system involving humans and their cultural artifacts. This is an aggregation of households or "domiciles" in which people reside. In keeping with the CAES model every community obtains resources and produces a product of some kind that can be used by other modules as well as internally consumed products used to maintain the integrity of that community. Our focus will be on a community that produces agricultural products to be used by, generally speaking, other nearby communities. Internally, each domicile will produce a significant amount of its own food requirements, but each will supplement their foods with products produced by the whole community (e.g. wheat flour) in sufficient excess of the community's own needs such that the product can be exported to other communities in exchange for products that those other communities produce that are not produced locally. Thus, this type of module is both semi-self-sufficient and specializes in trade. It is able to export some agricultural products. We'll call it an "agri-village" to evoke an image that is, in essence, a permaculture community[10]. The term 'permaculture' as originally coined, meant 'permanent agriculture' but has more lately been taken to mean 'permanent culture.' This meaning includes all aspects of living, including all normal activities in the domicile and communities. Much of the guiding ideas were influenced by systems ecology, particularly the work of Howard Odum (1983, 1994, & 2007; also Odum & Odum, 2001) and developed into a holistic set of design patterns (Holmgren, 2009; Mollison, 1997).

Our task is to design such a community using the methods of analysis of chapters 5 & 7 and using the CAES archetype (and its sub-models) to guide us. What follows will be a brief overview just to show the considerations that would be made in a full analysis and design. We will basically discuss the level 1 decomposition and a few instances of a deeper decomposition where it will shed further light on the process.

There are many permaculture communities in existence already around the world so the concept is well established. These communities generally follow the three core ethical tenets of permaculture[11], but can vary in terms of the degree to which they adhere. For example, the "fair share" tenet holds that individuals should take no more of the community wealth than what they need and by implication all individuals in the community have equal participation. To some this will seem like socialism and compare unfavorably to capitalism in terms of personal freedom to accumulate as much wealth as they can, and accept the notion of ownership and private property. Indeed, many of the communities operate as communes. On the other hand, most communities operate within the matrix of the larger capitalistic economies and have to interact with those economies; very few are truly self-sufficient. What we envision here is a world in which all communities are based on permaculture whether they are agri-villages or manufacturing villages.

Purposes

All CAES designs begin with a statement of the system's purpose or what function it is supposed to perform within the context of its supra-system. Only when the component system is producing something of value to the whole will the whole reciprocate with support for the component system. In the case of an agri-village, as already noted, the main export will be some

---

[10] See the Wikipedia article: https://en.wikipedia.org/wiki/Permaculture for background.

[11] See the Wikipedia article: https://en.wikipedia.org/wiki/Permaculture#Three_Foundational_Ethics for background. Accessed 2/28/2020.

foodstuff that is produced in excess of the needs of the community. However, the purpose is more complex than just producing food. As will be discussed in the next chapter (which describes the larger social system that is part of the agri-village's supra-system) every community contributes to the collective knowledge and to the governance of the whole social system. Thus, we can characterize the purpose in terms of producing a useful material/energy product as well as meaningful messages.

In addition to these productions, the community must replenish its own human and physical capital. As will become clear in the next chapter this does not mean 'growing' the population or increasing individual consumption. Indeed, for the human social system to be sustainable in the very long run, it will be necessary to maintain a stable population across the various module types and hierarchical levels. We will not speculate on the nature of the mechanism for establishing and managing a stable population size; many such mechanisms are well known and we will assume that societies will choose those most in consonance with its group values.

Below we will identify the product outputs that define the overall purposes of the system. We will consider the transformation function (section 3.3.3.4. Transformations) for the whole SOI but in general terms. We need to determine the inputs needed in order for the system to produce these transformations. One important difference between the use of the top-down decomposition process for design as opposed to the analysis of an existing system is that the transformation function in the latter can readily be approximated from obtaining input/output data over some time frame and then using machine deep learning (or preferably causal learning, Mobus, 1994) provide a starting point for modeling the SOI, noting that this approximation will be refined during the process of decomposition and the refinement of transformation functions at lower levels in the organization hierarchy. In design of a new SOI the transformation function is to be approximated by early specifications given the understood needs of the supra-system (the products to be exported to sink entities). These specifications may change as the environment (i.e. level -1) becomes better understood. But these specifications provide requirements and constraints on the subsequent design of the internal modules (level 1) once the flows into and out of the SOI (level 0) are better understood.

The whole process of design is generally an iterated recursive 'dive' into the SOI and its subsystems making refinements and adjustments to the assumptions made in the specifications.

Environment – Level -1

In keeping with the methods of Chapter 5, we begin our analysis of the system of interest with an examination of the environment or context in which the SOI operates, followed by analysis of the inputs and outputs crossing the boundary. The former will establish how the system will serve its purposes by identifying the relevant source and sink entities and the milieu in which the system operates.

Milieu

For an agri-village module there are several important milieu factors operating on different scales of time and space and having their own dynamics.

Climate

One of the more relevant factors conditioning the long-term (and thus evolutionary) aspects of an agri-village will be the climate of the region, which determines set of feasible crops that can be grown, for example. At the time of this writing the climate is being rapidly driven from the norms of the last ten thousand years (or so) through anthropogenic warming of the atmosphere and hydrosphere, with the acidification of the later (oceans) with increased CO_2 injections.

Climate considerations will condition all aspects of the design of an agri-village, down to the details of building construction and soil management. The reason is that the climate conditions the weather and as the former is trending toward a warmer world the weather patterns (already dynamically chaotic) are becoming increasingly unpredictable.

Seasonal Variations

As the climate warms, we are now beginning to experience a number of changes in weather patterns. Among the effects we are seeing are increasing variances in seasonal conditions such as average diurnal temperatures. These changes are having an effect on the kinds of crops a particular region can grow. There are also changes in the timing of plant growth cycles. Agri-village designs will need to take this into consideration as the system may not be able to merely adapt to intermittent variations, but evolve to trending variations.

Storms and Other Disturbances

The climate change that is developing has also had an impact on the severity and frequency of storms such as hurricanes, floods, and tornados. Designs of buildings, as one example, should take this into consideration. It is unfortunate that the most productive soils are found in flood plains. Some designs may be based on

Entities, Sources, and Sinks

There are several principle input and output requirements for an agri-village. Perhaps the most important is solar energy, both real-time, that is daily, and accumulated embodied energy in the form of fibers for various purposes, and wood for timber and heating. As well, the rainfall in the watershed is produced by solar energy evaporation of water. The input of solar energy depends on the ‘aperture’ or square area of land where plants are grown. Below we address the issue of this land area in terms of how much is needed to support a community.

In a permaculture-based agri-village the forest areas are included in the landscape and are, to some degree, cultivated or at least replanted as wood is harvested. So the forest along with the food crop fields are the main input interfaces with the sun as the source of energy. There are additional ways to capture solar energy as will be discussed below.

In addition to local solar influx, communities are expected to obtain specialized resources from other communities such as metals or manufactured goods.

Finally, agri-villages and other kinds of communities may be expected to obtain resources from ‘nature.’ The example of rainfall is a good one. The moisture in air that condenses as precipitation in the watershed will come from more than just the area of the community territory. Similarly, the food supply may be supplemented with meat from hunting. The populations of prey animals are the sources to consider.

The principle material output from the agri-village is the surplus foodstuffs it exports to customer modules, such as the one described above. In this instance the customer is also a source for the metals and clays used in the agri-village, part of the economy of the tribal module that encompasses the communities.

Of equal importance in designing a module is the minimization of waste products that cannot be used internally and have to be exported to the environment. Every effort must go into the design of processes, the outputs of which, can be recycled internal to the module. But when this is not possible then the design needs to consider sinks in the environment that could reasonably handle the effluents. For example, composting toilets can recycle much of human waste as fertilizer thus keeping it out of the output stream (i.e. the water flowing through the community). Animal wastes can be similarly managed and recycled as fertilizer too. Solid wastes should be minimized. We imagine that the use of plastics that cannot be immediately reused or recycled will be abandoned. There is probably no way to make such plastics in any case since they require oil derivatives. In fact, about the only waste product that cannot be internally recycled (or at least not easily) is the carbon dioxide exhaled by the people and animals. If the human social system is in balance with the Ecos, this will be recycled through plant photosynthesis. The ideal design will have almost no need for waste sink services.

Boundary Conditions and Interfaces

In section 3.3.3.3 Boundary, we provide guidance in how to consider the boundary of an SOI by examining the physical aspects of the boundary itself, e.g. a physical container or internal constraints such as stronger bonds between components than between components and external entities. This also includes the degree to which the boundary is able to exclude penetrations by substances (porosity[12]) and control inputs and outputs through ‘formal’ interfaces.

---

[12] An important aspect of porosity for human communities is the infiltration by pathogens. See the section below on the repair and maintenance functions of the SOI for a short discussion on the health care and public health functions.

For purposes of description of the design process we will make the assumption that an agri-village as envisioned will not be enclosed in a physical boundary. The nature of a permaculture design covers extended territory across many different landscapes such as fields, forests, watersheds, etc. Rather we imagine the boundary as being the territorial extent of the community, that is, the amount of surface area, regardless of topographical features, possessed by the people in the community. We say possessed as opposed to ‘owned’ since this surface area may actually need to change with changing conditions. There should probably be a wide buffer zone between neighboring communities that allow for shifts growing outward or shrinking inward. In all cases the area possessed should never be more than a community needs to maintain self-sufficiency. Territorial recognition by neighboring modules (see the governance architecture discussion in the next chapter) and on-going cooperation in management of the buffer commons[13] is essential.

There are at least two ways to think about the boundary of an agri-village. One way is to consider a boundary around the ‘village’ itself, excluding the resources that the village needs to import. Alternatively, we might consider the ‘territory’ including the sources of resources that are obtained by the village directly, as opposed to inputs from other villages. In the former case we need to consider those sources, the area of land and water that is ‘controlled’ by the village. This is no easy problem. Many resources spread over a wide area may be shared by multiple villages (a commons) without conflict as long as there is monitoring of the usage rates and a coordination function making sure none of the communities participating take more than their fair share. The details will likely depend on the distribution of resources within a particular landscape. For this exercise we will consider a boundary around a particular community and define (design) specific external sources (and sinks) and interfaces with this larger supra-system entities (sources and sinks). In this case what is inside the system is the village, its cultivated lands, fields used for livestock, and water reservoirs directly used by the village. Thus, all other flows are inputs to or outputs from the community system.

Interfaces

The boundary is defined, in part, by a list of interfaces for these inputs and outputs. Here we mention a few examples. These segregate, of course, into input and output categories, which will be discussed below, but also into classes of each category. For example, input interfaces will include the obvious subcategories, energy, material, and messages and output interfaces will include the subcategories of product, material wastes, and waste heat. In this last case we recognize that waste heat will radiate or be removed by convection (air currents) from every subsystem so we need not consider a special heat removal process. Each of the subcategories of

---

[13] In truth, management isn’t quite the right term here. Buffer zones would not really be ‘commons’ in the usual sense, but rather a general kind of asset (nature) that profits the surrounding communities simply by existing and regulating its own self in the way ecosystems do.

inputs and outputs will further be differentiated by types of energy, types of material, types of messages, types of products, and types of wastes.

For example, if a village receives a kind of food staple from a different village, one the receiving village does not itself produce, then there must be an interface for either picking up the food from the producing village, or receiving a shipment from that village (who does the transporting?). Food is an interesting case where both material and energy are in a coupled state as they enter the system and both must be accounted for[14]. The energy is used to support the living entities and the waste products, CO_2, urine, and manure (and eventually the no longer living biomass of the deceased) have to be considered. All but the first should be considered as recyclable.

Other interfaces for receiving or exporting materials may include, for example, a saw mill for receiving and processing wood products obtained from a forest.

Input/Output – Level 0

Export Processes,

Main Product – Calories Out

The main product of a food-producing modular community would nutritious calories, meaning vital nutrients and bioenergy. It is expected that to some degree or another every community in the future world will be responsible for some aspects of food production (and preserving) but some communities may be tasked with other forms of production (such as making tools) and have a lesser capacity to be completely self-sufficient. They will be customers of excess production by communities that are primarily food producing.

The design would call for careful consideration of the methods and timing of exporting food products to other communities. The interfaces could be in the form of markets as is the case today, but markets depend on a certain amount of competition which would defeat the purpose of establishing a more global system of trade. More appropriately, the design might include direct trade agreements (protocols) between communities, where the food-producing community agreed to produce certain food products at a certain rate and deliver those products at agreed upon times and places.

---

[14] This is not much different for the case of systems that, for example, burn fossil fuels to obtain energy. The outputs of oxidized carbon and hydrogen (along with things like sulfur dioxide and trace metal pollutants) and leftover ash, if any, are waste materials. The inputs are the fuels as material substances that contain the sought energies in the form of exothermic covalent bonds as well as oxygen and some initial energy to start the combustion.

Waste Products

Import Processes, Interfaces, and Sources

Energy

Materials

Messages

Internal Processes – Level 1

Work Processes and Economy

Most of the work processes at level 1 will be the domiciles and the agricultural process. However, the community must also be capable of producing clothing and other artifacts as needed. These will be described under the Peripheral Processes below.

Agriculture involves

Management Processes and Governance

In the following chapter we describe the basic management/governance architecture for the social modules. This architecture implements the governance archetype model specifically for human organizations and society in general. Here we will only consider the level 1 requirements in terms of what needs to be managed and governed within the agri-village module.

Agriculture production

Domicile Coordination

Peripheral Processes

Adaptation Processes

Evolvability Processes

Repair and Maintenance Processes

Recycle Processes

References and Further Reading

Volk, T (1995) *Metapatterns Across Space, Time, and Mind*, Columbia University Press, New York

Mobus, GE (1994) "Toward a theory of learning and representing causal inferences in neural networks", in Levine, D.S. and Aparicio, M. (Eds.), *Neural Networks for Knowledge Representation and Inference*, Lawrence Erlbaum Associates Associates (now Taylor & Francis), New York