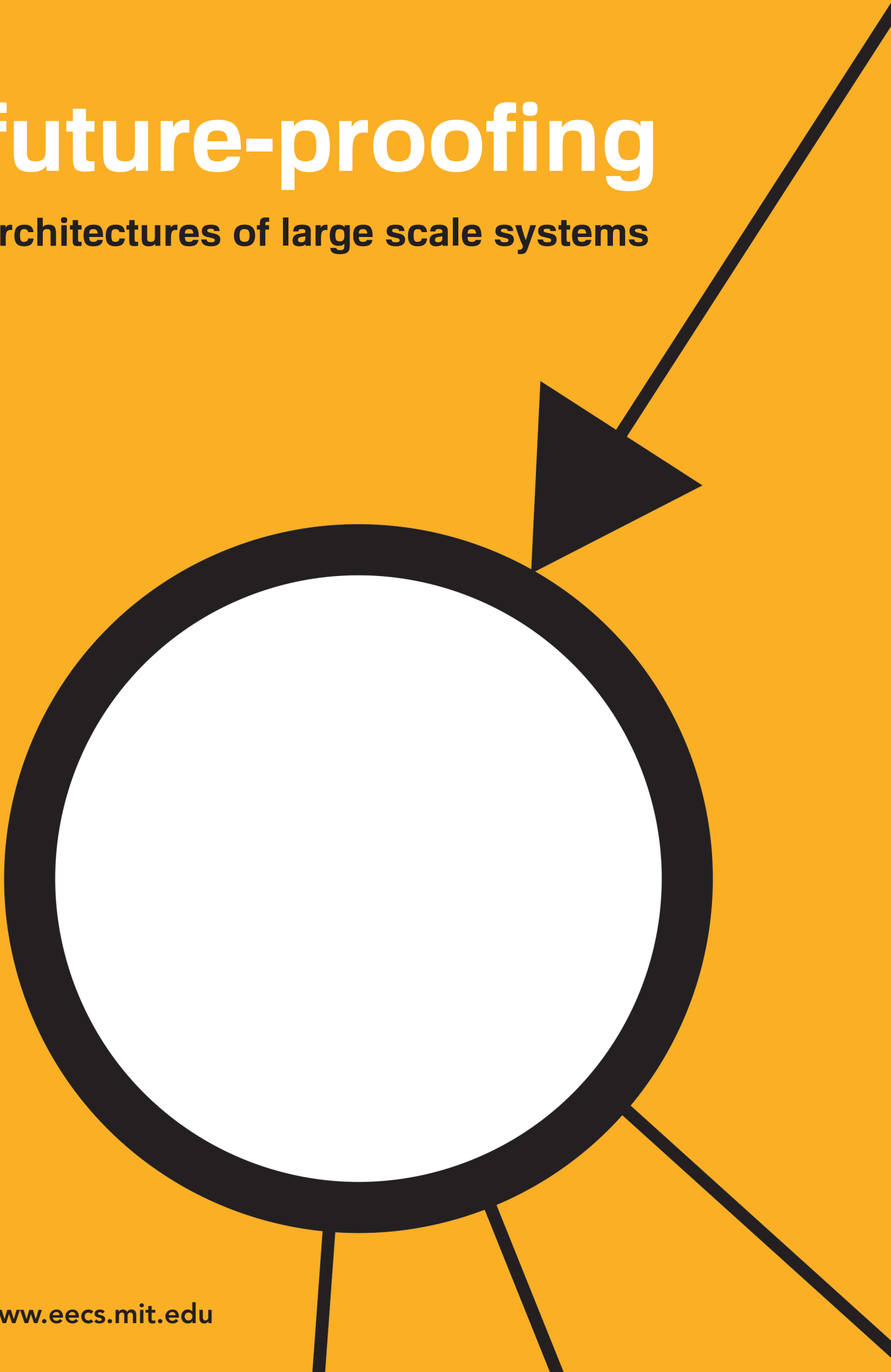
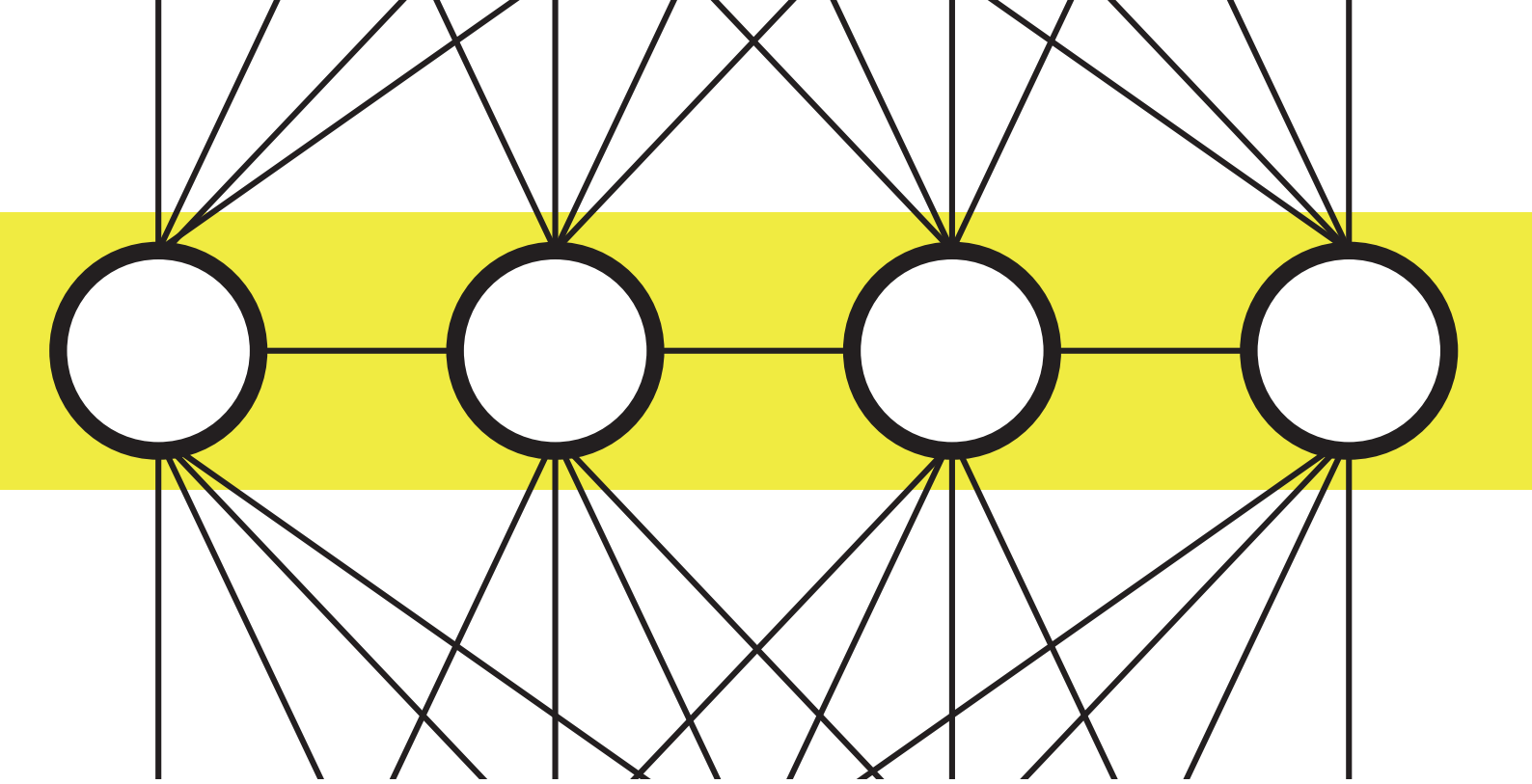
future-proofing

architectures of large scale systems



www.eecs.mit.edu



The big picture: managing complexity

Studies of the properties of different architectures for large scale systems at the **Massachusetts Institute of Technology** have identified a layered structure as ideal in many contexts, avoiding chaos and offering flexibility for future change

In the field of large scale software systems, the role of the systems architect is to take a holistic view to design the most effective solution to meet known and evolving requirements. However, the requirements of a system extend beyond its business or user immediate needs, which govern its functionality from the perspective of the user. Requirements such as availability, sustainability and appropriate recovery after failure, termed non-functional requirements, are a key concern for the system architect. Among these, several in particular are crucial if the system is to be resilient to change in the future: its scalability, lifetime maintainability, controllability and its flexibility.

With the use of computing power to solve ever more ambitious problems and the advent of the Internet with its relatively rapid changes, the requirement to serve networked users in real time means that systems must be capable of fast and flexible modification without unduly growing the complexity of the system.

Dr Joel Moses is Professor at the Massachusetts Institute of Technology in the Department of Electrical Engineering and Computer Science and the Engineering Systems Division. His group developed the Macsyma system for computer algebra in the 1970's. His knowledge of the problems of handling complexity in software engineering has led him to observe a close correlation between large scale software systems and those in a wide range of other contexts, such as enterprise organisations, healthcare services, and automobile manufacturing. His central tenet is that a system's architecture is fundamental if it is to survive and succeed in a constantly changing environment: robustness and resilience must be designed into it from the start. Through mathematical abstractions and analysis, Moses has defined the characteristics of three idealised generic architectures for large scale systems, each of which excels in a particular context, delivering the correct mix of flexibility controllability, complexity and capacity to survive change.

TREE STRUCTURE

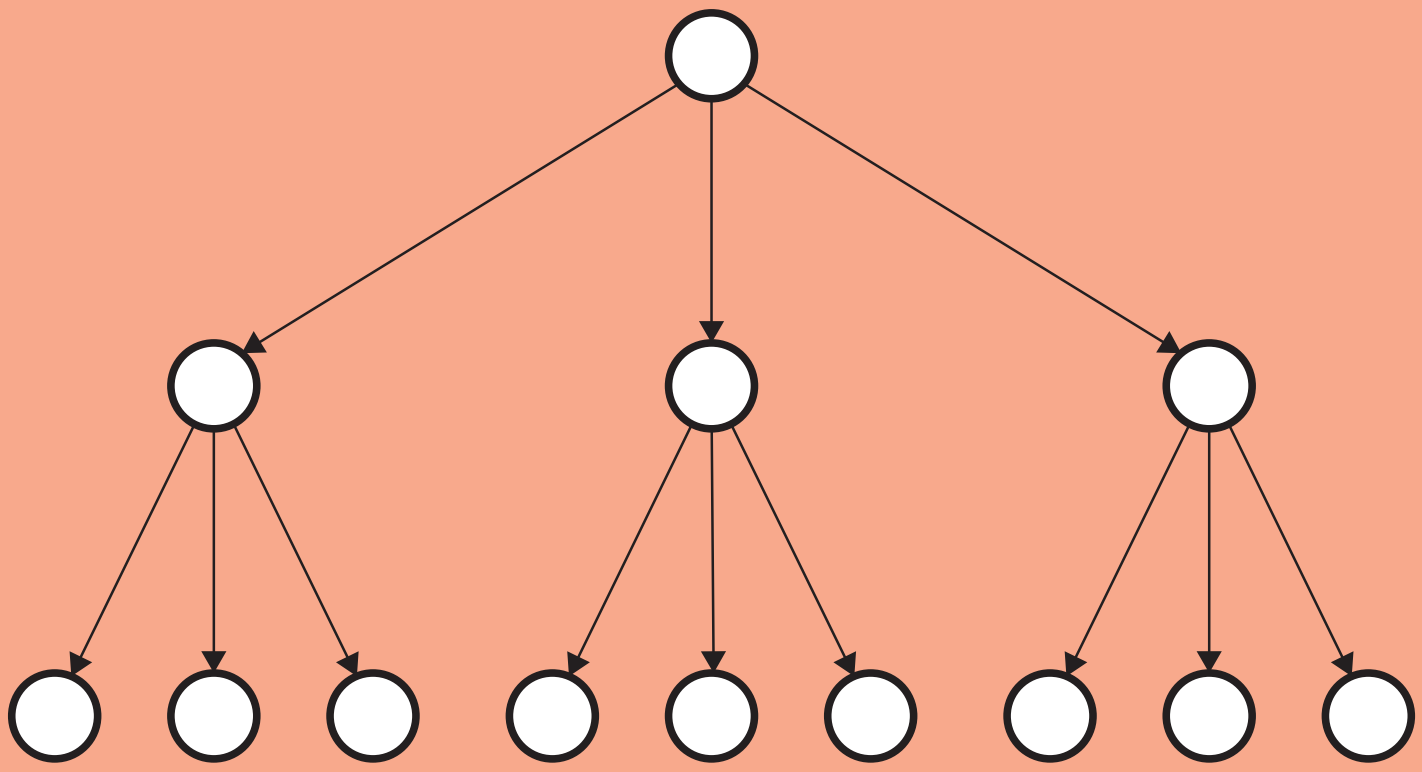


Figure 1. Architecture of a Generic Tree Structure

Moses' generic architectures

In graphical representations, architectures are depicted as a compilation of components, or nodes, and the links between them. Tree architectures, as used for conventional company diagrams describing management roles, responsibilities and reporting lines, are suited to certain classes of hierarchical systems.

The basic rules for generic tree structures are that every node, except for the topmost, has exactly one parent node and there are no horizontal connections: "The rules mean that information is exchanged only with one's supervisor and subordinates, but not with colleagues," explains Moses. This analysis shows that generic tree structures have utility for certain classes of very large systems, but are inherently inflexible. Flexibility is related to the ease of changing the control or information paths in a system. In a tree structure it is relatively easy to add nodes or interconnections to leaves of a tree. One can even merge two large tree structures. What causes problems is when one creates numerous connections between nodes that are not on the same paths. Eventually such connections will lead to exceedingly messy and structurally complex systems that will be difficult to modify further and achieve one's revised goals. Tree structures are the most difficult ones to achieve flexibility. Moses defines flexibility as the number of paths in a system, beginning with a top node and ending with a leaf node. Tree structures have the lowest ratio of the number of paths to the number of nodes to be found in a generic system architecture.

GRID NETWORK

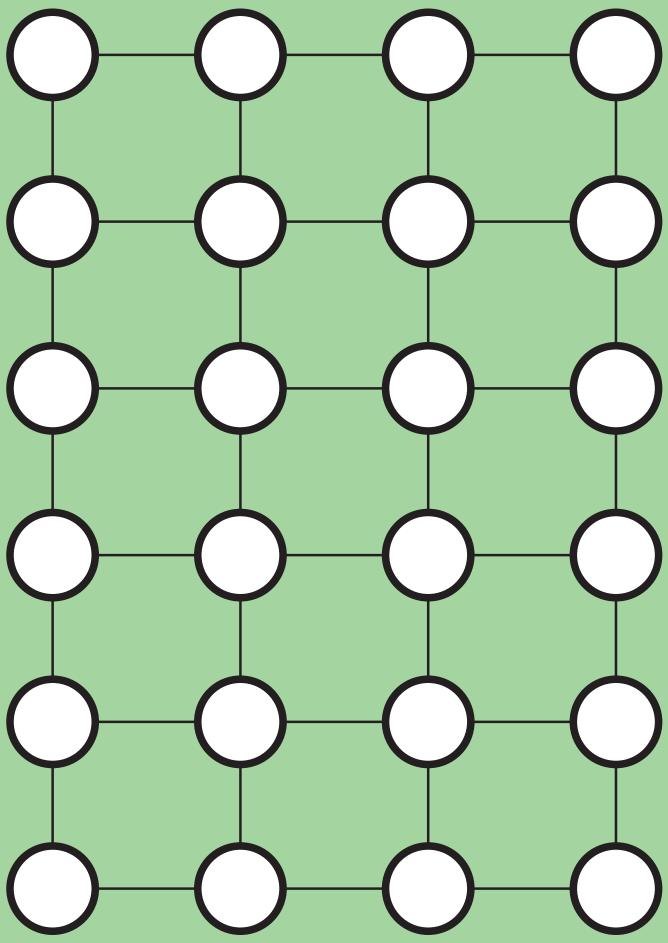


Figure 2. Architecture of a Generic Grid Network

Moses' second generic architecture is the grid network, in which each node is connected to its neighbouring nodes only, so its structure is flat and not hierarchial.

Grid networks can model enormous, complex systems – Moses' example of the application of the grid network architecture is a computer-based weather modelling system: "Grid networks have potential for great flexibility since there are exponentially many paths, as a function of the number of nodes," details Moses. "Unfortunately, such systems can exhibit chaotic behaviour, as the weather system shows." One cannot accurately predict the weather in general using computer models for more than a couple of weeks.

With some properties of both tree and grid architectures, Moses' third generic architecture is the layered structure.

Layered architectures can model certain classes of hierarchical systems, with the difference between layered structures and generic tree structures being that nodes at the same layer can be connected to each other, and each node can have multiple parent nodes in the layer immediately above it. Like grid configurations, layered architectures can have many standard connections between adjacent nodes, the difference being that these connections can be used both vertically and horizontally. "Layered structures can handle many classes of changes with relatively great flexibility and no undue increase in complexity: adding a new horizontal or vertical link to the next layer will not change the structure's generic architecture," notes Moses. Moses describes the architecture of the Internet as based on a three-layered design.

LAYERED SYSTEM

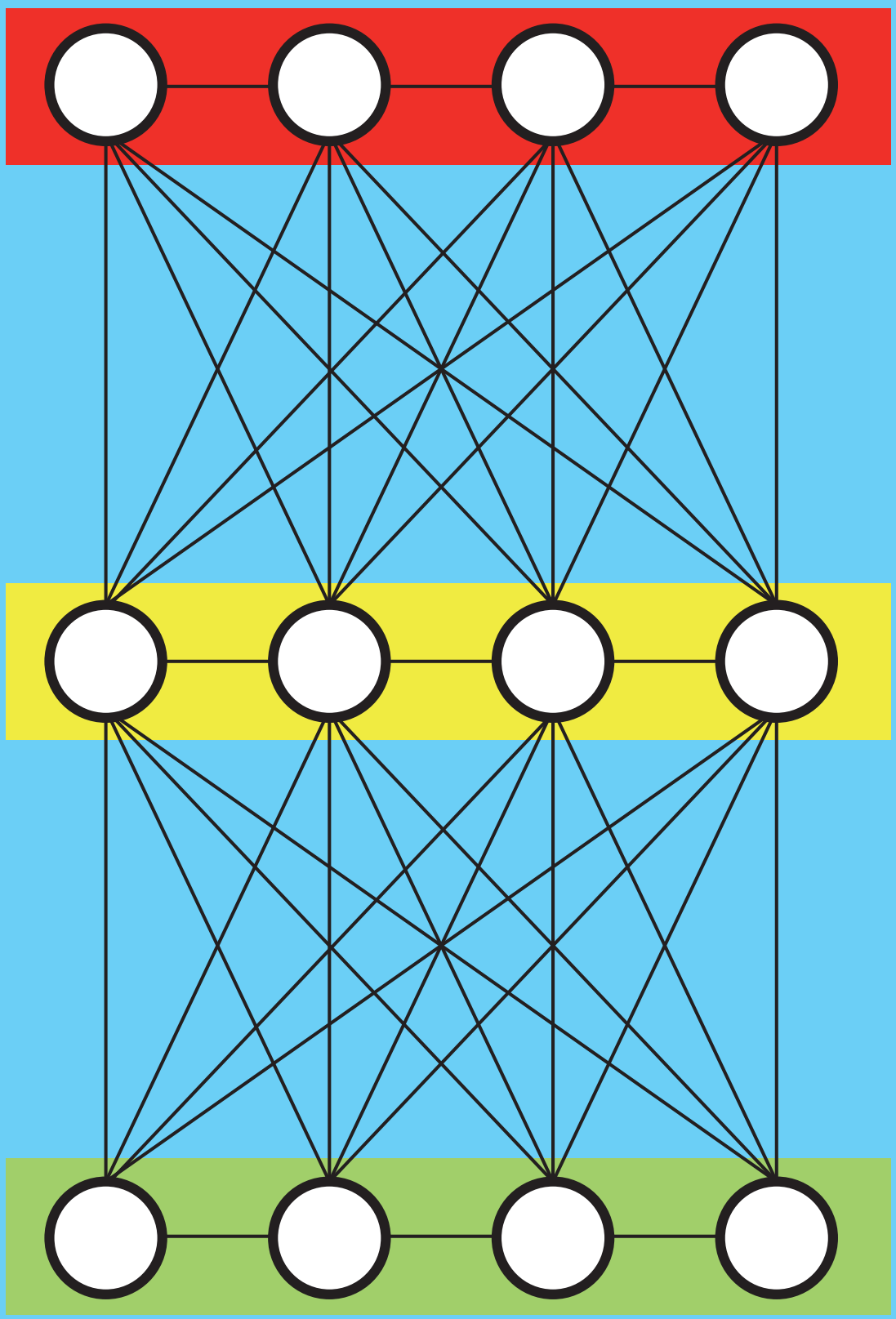


Figure 3. Architecture of a Generic Layered System

GENERIC TEAM

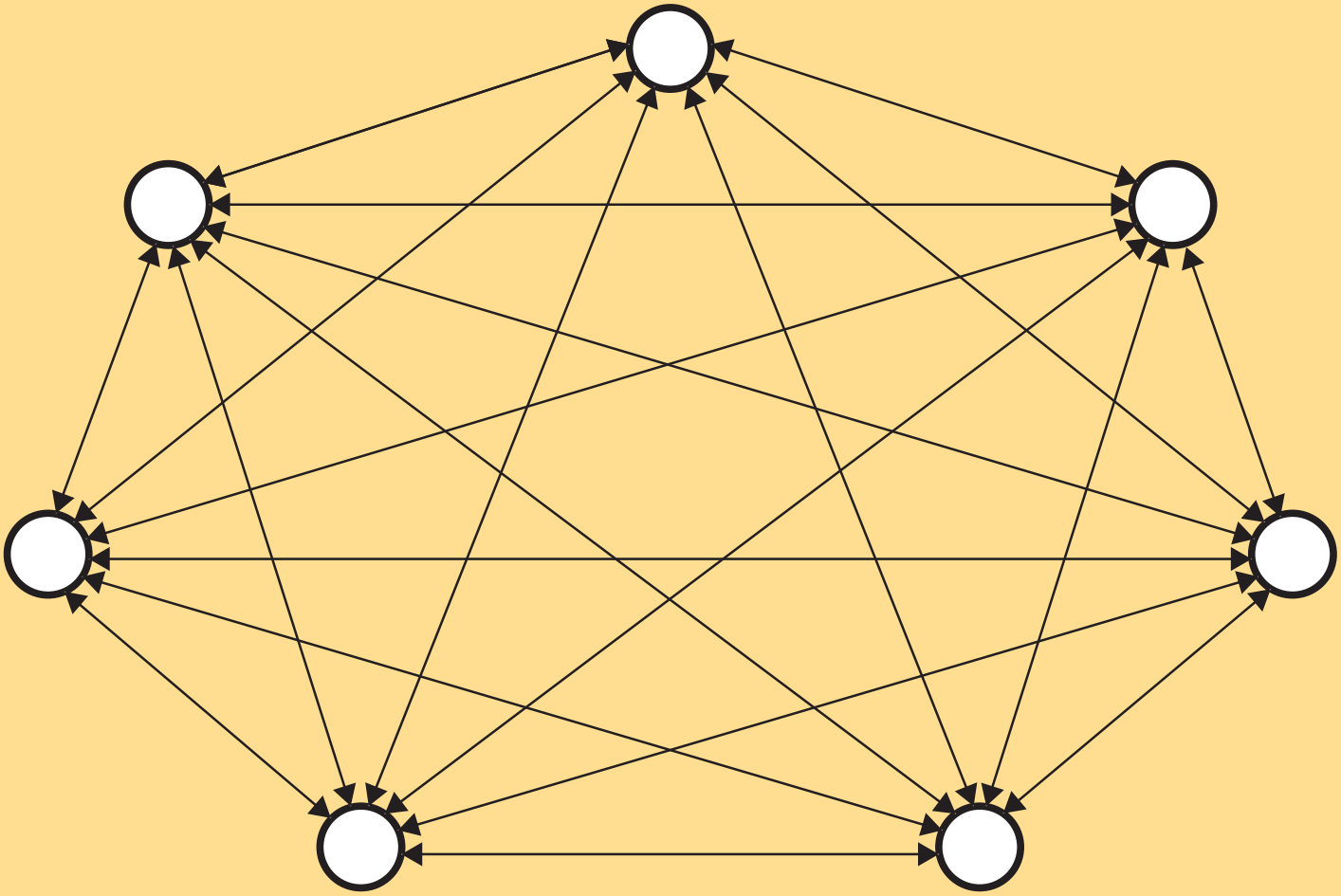


Figure 4. Architecture of Generic Team

In Computer Science layered architectures are associated with the concept of levels of abstraction where each layer is an abstraction of the layer below it. Abstract algebra in mathematics uses similar concepts.

The final generic architecture is that of a team. Within the team, each node is connected to all of the others, which adds a relatively high degree of structural complexity. While a benefit is that the structure can cope with high rates of change, there is a disadvantage in that it is impossible for humans to maintain close interactions with a very large number of others, which thus limits the number of nodes, or members, in a team of humans.

Towards understanding the architectures of complex systems

Health Care Delivery as an Example of Layered Systems

The so-called Obamacare Act deals mainly with providing health care insurance for many millions of Americans who do not currently have such insurance. This is an important change for the U.S. Less attention is paid in the bill to the overall national cost of health care delivery in the U.S. This cost can be nearly a trillion dollars per year higher than that of France, for example, when one considers the relative national health care cost between the U.S. and France, as a function of the relative national GDP. Here we point out ways in which system architectures can reduce the overall cost of health care delivery in the U.S.

The Health Care Delivery system in the U.S. can be viewed as a layered system with three layers. The middle layer is composed of General Practitioners and their staff – nurses and physicians' assistants. At the top layer we have specialists and their staff, and hospitals associated with them. The bottom layer has numerous facilities, such as drug stores and community clinics. A key role of GPs is to make sure that the patient's care is provided in a coherent, consistent and efficient manner. The patient should not fall between the cracks of the various specialists and lower layer providers. The current name for this role of the General Practitioners in the middle layer is 'patient-centered medical home.'

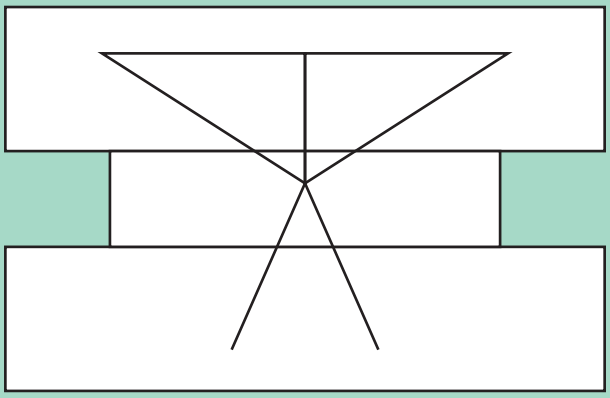


Figure 5. Architecture of a Medical Home

The figure shows three layers of coordinated health care providers: specialists, primary care physicians and their staff, nurses and other health professionals.

An Accountable Care Organisation is a relatively large health care organisation that provides medical care to a large set of patients. A group of health care providers (e.g. GPs, specialists, hospitals) make up the ACO. They often get paid by insurers a fixed amount based on the number of patients (global payments). In return the ACO is accountable for the quality of health care provided to the entire set of patients. The insurers usually create incentives to the ACO for reducing the overall health care costs for the entire set of patients. Figure 6 below indicates the layered structure of an ACO, which is usually within a geographic region. The top layer has hospitals and specialists. The middle layer has GPs in a medical home. In contrast to our medical home model above, the ACO will have a sizable number of GP practices, as shown in the figure.

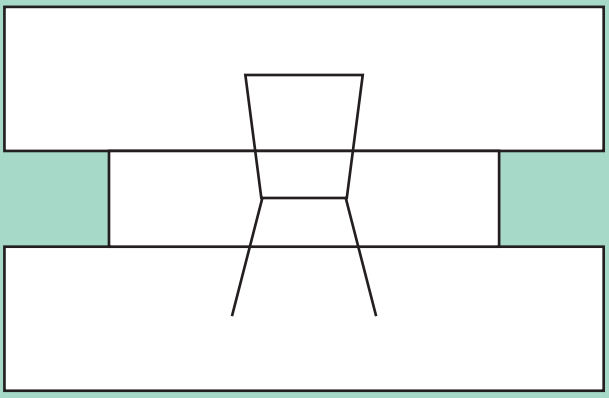


Figure 6. Architecture of an Accountable Care Organisation

The previous figures (5 and 6) indicate an advantage of layered architectures in that they provide relatively simple alternative models for the structure of certain socio-technical systems, such as the U.S. health care delivery system.

We also propose a three layer structure for tertiary or teaching hospitals. The lowest layer of these tertiary hospitals would have an emergency room and some general medical services, such as x-rays. The middle layer would have several specialty subhospitals for issues such as cancer and diabetes. These would share some medical services (e.g., anaesthesiology). The top layer of these tertiary hospitals would handle cases that have been difficult to diagnose. The top layer would also include the use of very complex or experimental procedures. Master diagnosticians or teams of diagnosticians would practice in this layer. Master diagnosticians need to be both deep and broad in their knowledge base of medicine. Systems thinking will likely play an important role in their analysis of complex medical cases.

A key advantage of specialised subhospitals is that with much practice teams of doctors, nurses and other staff members can continually improve their ability to treat a class of patients with high quality at relatively low cost. Physicians in the specialised subhospitals may need to play multiple roles. They are, of course, members of specialised teams in a subhospital. They are also specialists who may need to consult on cases occurring in the emergency department or in the top layer of the hospital. Some specialists, such as pathologists or radiologists, may need to consult in several subhospitals as well as the other two layers of a tertiary hospital. A given tertiary hospital need not have all possible specialised subhospitals. Some specialised subhospitals simply would not have the volume that justifies their presence in numerous tertiary hospitals. The competition between subhospitals in the same specialty in a given geographic area should lead to some having large volume based on cost, quality and general reputation, thus using competition to drive out some other subhospitals in the same specialty.

Clay Christensen [The Innovator's Prescription] suggests that there ought to be a clear difference in how specialised hospitals are paid from the way members of the top layer or bottom layer in a tertiary hospital are paid. The specialised subhospitals ought to be paid a fixed price for their usual procedures. If there are complications associated with these procedures in a given patient, the specialised subhospital will have to take care of them without additional payments. This will place pressure on the subhospital to continually improve its patient outcomes.

Diagnosticians at the top layer of the tertiary hospital should be paid by the hour for diagnosing a complex medical case. Reliance on teams of diagnosticians is an approach that is used in the Mayo Clinic and at few other places. Paying by the hour for the top layer is an approach that results from the difficulty of diagnosing these patients. We should be willing to pay by the hour for teams of master diagnosticians since a good diagnosis in complex cases will often greatly reduce overall costs as well as save lives. Neither payment approach is a fee-for-service approach, which is the current major approach in the U.S. Fee-for-service creates a tendency to use too many procedures, since doctors and hospitals are usually paid for each procedure, even if, for example, some of the procedures are needed to deal with avoidable complications caused by prior procedures.

SOLID
LIQUID
GAS

An Analogy to Major Phases of Matter

In a recent collaboration with Prof David Broniatowski, Moses explores an analogy between the ease of making changes to connection patterns in the three main generic architectures of large scale systems and of making changes to the three major phases of matter: solids, liquids and gases. Like a generic tree structure, making adjustments to the internal configuration of a solid is difficult and can eventually add greatly to the complexity; like the grid structure, a gas is relatively easy to modify, but when the environmental rate of change is extremely high, it can exhibit chaotic behaviour. Finally, liquids, like layered architectures, are intermediate between gas and solid phases: liquids of different compositions, such as oil and water, can form layers, and while these can be changed readily, the rate is slower than with a gas. Ultimately, the analogy shows that responding to a high degree of change can transform the architecture of a phase of matter from one to another type – from liquid to gas, for example. Such transformations can also occur in systems using the generic architectures.

Dr Stuart A Kauffman's work [The Origin of Order] into the analogy of complex systems to solids, liquids and gas phases is of significance to Moses and Broniatowski's research. Kauffman emphasises the self-organisation of systems rather than changes made in those systems by outside designers, and it is this approach that is important to the work of Moses and Broniatowski. "This design perspective is central to our work and may be said to differentiate an engineering approach from a natural science approach," Moses elaborates.

Systems for the long-haul

Looking ahead, Moses and his colleagues are confident that their research and approach to system design will provide an important set of concepts for system architects for the foreseeable future. He concludes: "Architectures of real systems differ from the generic and ideal types we discuss, but we expect that a deeper understanding of the relationships between complexity, flexibility, communicability and generic architectures will help system architects make trade-offs that will allow their systems to cope with changes in their environment for a long time."

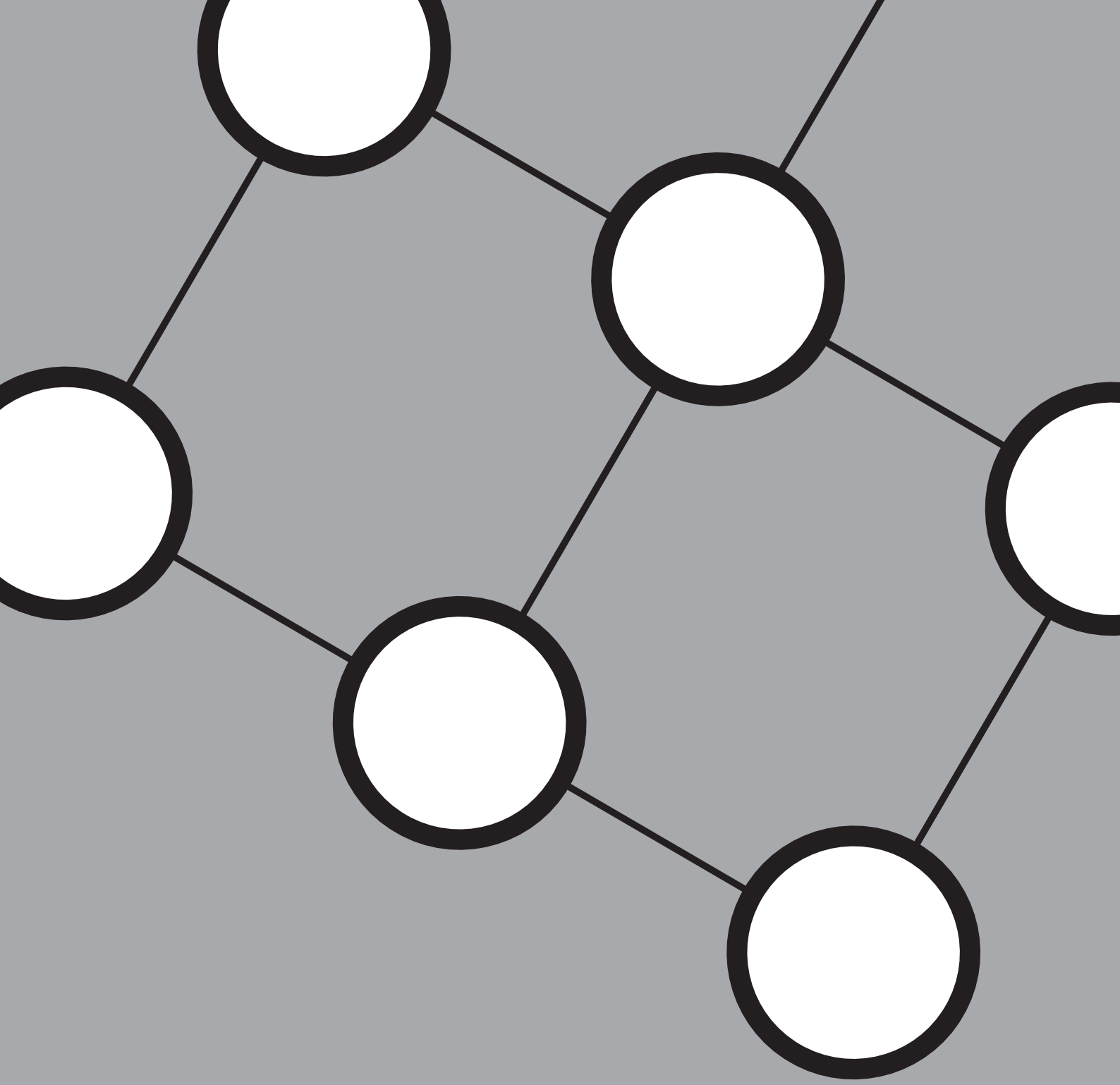
Moses central tenet is that a system's architecture is fundamental if the system is to survive and succeed in a constantly changing environment

Key Collaborator: David A. Broniatowski. Assistant Professor in the Department of Engineering Management and Systems Engineering at George Washington University.

Principal Investigator: Joel Moses, Institute Professor, MIT, Room 32-249, 77 Massachusetts Avenue, Cambridge Mass. 02139

This article is based in part on an article in International Innovation, North America, November 2013, pp. 97-99

Funding: Natiomal Science Foundation – grant no. 1023152



www.eecs.mit.edu