Duane P. Truex, Richard Baskerville,
and Heinz Klein

Growing Systems in Emergent Organizations

As the new economic realities pressure your organization to change from stable to emergent, new practices for IT support are required.

The current frenzied pace of organizational change is being driven by the rapid development of commercial technology, global markets and reengineered, quality-oriented organizations. This constant need to change gives rise to a recognition that organizations in the present era are no longer stable, but are continuously adapting to their shifting environments. These organizations can be said to be in a state of constantly seeking stability, while never achieving it. Such organizations are said to be "emergent," and include many of today's commercial and governmental organizations.

When we refer to organizations as emergent we are saying that every feature of social organizations—culture, meaning, social relationships, decision processes and so on—are continually emergent, following no predefined pattern. These organizational features are products of constant social negotiation and consensus building. The organization itself or any of its features may exhibit temporal regularities. But such temporal regularities are recognizable only by hindsight, because organizations are always in process; they are never fully formed. We use the terms "emergent" and "emergence" rather than "emerging" because "emergent" refers to the state of being in continual process, never arriving but always in transition. "Emerging" differs from "emergent" because it gives rise to the possibility of a current state being a stage to a possible outcome and always arising from its previous history and context. So organizational emergence refers to a theory of social organization that does not assume that stable structures underpin organizations [1, 7]. This theory indicates new assumptions about the environment in which information technology (IT) must succeed. For example, in the past, IT designers strove to cre-

ate stable systems with primary goals that included low maintenance and long life spans. The relatively long life spans of stable information systems (IS) hinder organizational emergence. Having low-maintenance, stable systems means the organization is continuously battling against its constraining information systems as it adapts to an ever-changing environment. In this scenario, IS will inhibit rather than facilitate organizational change.

There have always been limited means to match IS development to a rapidly changing organization. The available means include prototyping, end-user development, and open systems connectivity. But these are inadequate because they are not connected though a coherent framework that focuses on the emergent character of organizations. If emergence, rather than stability, is taken as the dominant character of organizations, at least in some periods, there is a need to radically rethink the way in which IS are developed. Rather than viewing information systems development (ISD) as a series of projects each having a clear beginning and end, emergence calls for a continuous redevelopment perspective. A continuous redevelopment perspective implies the creation of a ISD environment that is optimized for high maintenance rather than low maintenance. Within an organization that values continual change, low maintenance is evidence of an unadaptable IT system. These systems lead to *stable systems drag*, a condition in which the organization must adapt to both to its environment and its outdated IT systems. With stable systems drag the IS actually inhibit adaptation; so organizational emergence must necessarily break free from the IS constraints. IT systems that do not produce stable systems drag are designed to adapt with an organization, shifting the organization's essential adaptation constraints to the external environment and not its own rigid internal IT framework.

Levers to Stimulate Emergence

A continuous redevelopment perspective not only involves the elimination of stable systems drag, but it also involves using IT to support and actually promote organizational emergence. In order to understand how IT can promote organizational emergence, we need to understand some of the forces behind organizational emergence. Here, we consider the three "levers of encouragement" that are known to stimulate emergent organizations.

Shared reality construction. Organizational form, structure and activity are a result of complex and continuous interactions between organizational members. What organizational members believe to be real for the organization is an outgrowth of these continuous interactions and the constant negotiation of fact, opinion, and meaning. Thus for all intents and purposes, the reality of any social organization is defined as whatever people in that organization believe is real. If the members of the organization agree that the organization is flat and lean, then the organization is flat and lean. For they act as if it were so, which in turn helps to alter all systems and social structures to conform to the shared perception of leanness and flatness [6]. This belief goes beyond individual or group delusion, and involves the construction of reality by a society [3]. Emergent organizations capitalize on this phenomenon by encouraging reality reconstruction.

Self-reference and organizational identity. An organization uses its own identity as the primary point of reference when it reconstructs itself [5, 8]. This means that the socially constructed realities of an organization form the basis for the next version of the organization. As an organization adjusts and changes it does so with reference to its former self in a more or less constant mode of self-reproduction. That is, in continously reproducing itself the organization must do so with constant reference to itself, its past practices, values, decisions, contracts, and commitments. It is self-referential. Emergence theory covers both self-reference and reproduction, with the caveat that nothing is ever reproduced in quite the same way. When the organization possesses a narrow identity, a reconstruction of the organization will be very much like the original version. Such organizations may emerge rapidly, but with minor changes. An organization with a broad identity may emerge slowly, but with major changes. Self-reference and self-reproduction concern the degree of change over a unit of time and are called autopoiesis [4]. The important idea here is that the organization is in a continuous state of adjustment. While these are often very subtle adjustments, the process is one that is very hard to stop. And, like the tectonic plates beneath the earth's surface, the motion is continuous no matter what the surface appearance may be.

The dialectics of organizational autopoiesis. Autopoiesis is the process by which organizations emerge. This lever comprises the individual relationships between members of the organization. Dialectical engagement is the way organizational meaning (and hence self-reference and constant change) is negotiated. The dialectics of this process center politics, conflict and struggle between social forces in the organization [2]. These processes affect shared reality construction and self-reference. Conflicts can arise when there are multiple versions of reality floating around an organization. By nurturing such conflicting versions of reality, reality becomes easier to recon-

struct, and organizational emergence is encouraged. Similarly, conflict is also important for self-reference, since conflict can create multiple identities and thereby decrease the similarity between a reconstructed organization and its previous version.

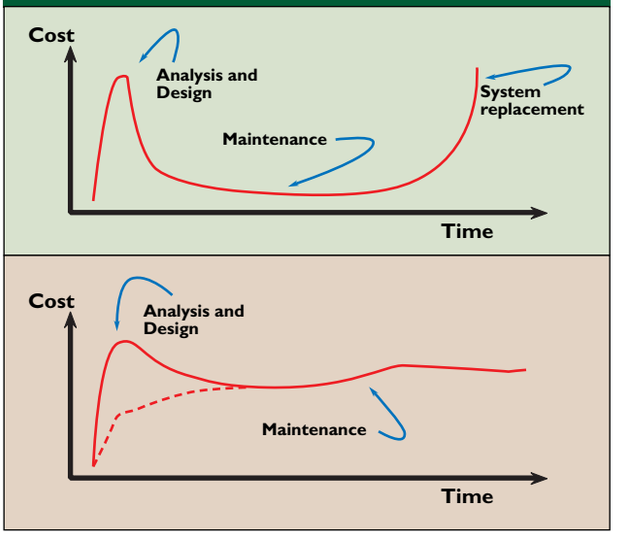
Revoking Traditional ISD Goals

Emergent organizations place less value on organizational stability. Removing stability as an organizational presumption affects the ISD goal set. At least five of the central goals of the 1960s–1990s ISD process become obsolete upon adopting the emergent systems viewpoint. These central goals were highly valued by IS managers and developers, but are inappropriate for emergent organizations. These now obsolete goals of IS development include: 1) proper IS analysis and design requires formal, often lengthy, analysis and design activities in order to minimize maintenance activities; 2) one must achieve user satisfaction; 3) one can and must create a reasonably complete and traceable set of abstract requirements; 4) complete specifications can and should be derived from these abstract requirements; and, 5) ISD requires rigorous advance planning. These goals are interrelated, and form a cohesive goal set that dissolves as a whole when notions of organizational stability are removed from predominance in the organizational goal set.

Lengthy analysis and design are poor investments. One obsolete ISD goal is a systems life span that centralizes large-scale analysis and design activities in order to minimize maintenance activities. If the organization is very stable, precisely designed systems may satisfactorily operate with minimal changes for long periods. This was very important in the early history of IS because computer hardware and custom software were extremely expensive to obtain and maintain. Applications had to be stable and operationally inexpensive over long periods to economically justify the initial system costs. Target applications were of low volatility, such as transaction processing and database management. The ISD mindset presumed that a large investment in systems analysis and design was recouped over the long period of low-cost operation and maintenance. This is sometimes represented graphically in a manner similar to the top diagram in Figure 1. A high-cost analysis is justified by a long low-cost operation terminated when the maintenance costs rise exponentially (thus justifying a replacement system). However, many essential applications in emergent organizations need higher constant volatility.

Systems that are forced to observe long periods of low volatility and minimum maintenance increase the stable systems drag on the emergent organization.

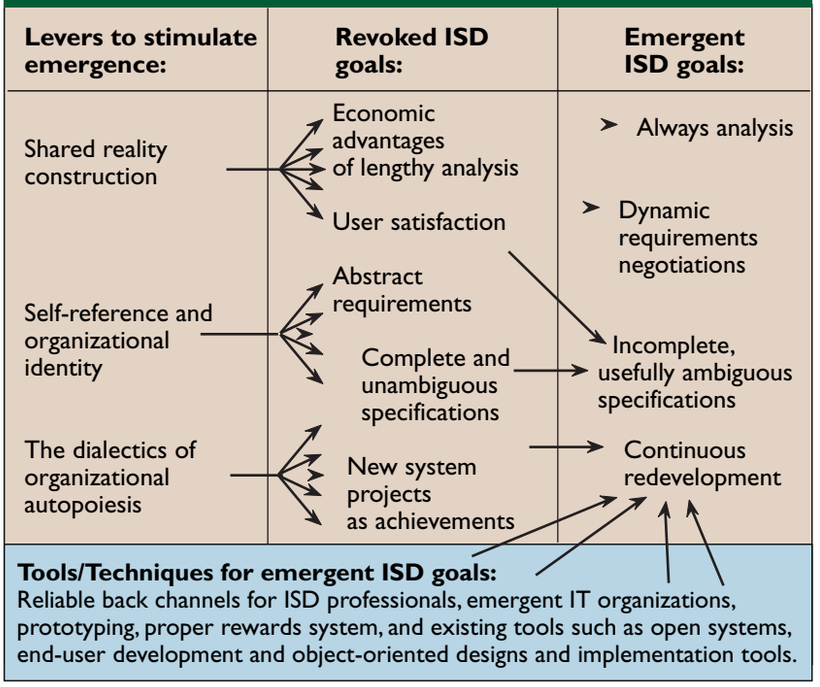
Figure 1. Alternative lifespan economies.



Maintenance of such systems is expensive, because the implementation technology is typically cheap to operate, but expensive to adapt (for example, custom C++ programs and centralized database management). In order to adapt such high-cost systems, high maintenance costs inevitably characterize the maintenance period. This is shown as the solid line in the lower diagram of Figure 1. In fact, the high analysis and design costs do not bring long-term, low-cost maintenance to emergent organizations, but long-term, high-cost maintenance. A shorter and less intensive analysis and design effort would probably result in the same high-cost maintenance levels (illustrated by the dotted line in the lower diagram of Figure 1). Since large-scale analysis and design projects lose their economic value in emergent organizations, the related ISD goals are obsolescent.

User satisfaction is improbable. The second obsolete ISD goal is user satisfaction. The stable systems ISD mindset enrolls users as active consumers of the IS product. User participation and acceptance of new systems is of central importance under stable system thinking because users are assumed to understand their own current and future needs. Under emergent organization assumptions, user needs may unfold rapidly in directions that are poorly understood by the users themselves. Since the users' needs are evolving, even during requirements determination activities, users become frustrated and trapped by the system they are helping to shape. In emergent ISD, user participation purposely exposes the specification process to the conflicts in the user world. Users can never be satisfied in emergent organizations, because their needs are always changing. The user-systems analysis relationship is characterized by continuing conflict

Figure 2. Mapping old assumptions to new perspectives.



and dialectic that stimulates change in the IS. The emergent ISD mindset centralizes the process of the user-systems analysis dialectic that advances emergence. This ISD goal does not seek the delivery of a stable IS product to the users, rather it is related to the delivery of the ISD dialectic service that continuously adapts the existing IS.

Abstract requirements are largely imaginary. A third obsolete goal relates to the central value of abstract requirements determination. Stable systems thinking presumes that a stable set of abstract requirements awaits discovery by talented analysts. The abstractions are useful for raising the requirements process out of the turmoil of daily activities. Emergent systems thinking assumes that day-to-day turmoil is central to IS requirements, and that requirements are always in motion, unfrozen, and negotiable. Any distinctions between IS requirements and post-acceptance (future) enhancements are artificial IS project devices that excuse the delivery of an obsolescent IS. The diminishment of the requirements goal relates to the obsolescence of large-scale analysis and user satisfaction goals because the major analytic target (abstract requirements) is unsuitable for emergent organizations. A labor-intensive review of the current situation is little more than a history lesson in past organizational states, and future requirements are abstractions of obscure user guesswork about future organizational states. Even if analysis is lifted from its dependence on user guesswork, the unpredictable directions of self-referencing emergence make concise analysis of the distant future improbable.

Complete and unambiguous specifications are ineffectual. A fourth obsolete goal in ISD is a complete and unambiguous specification. Stable systems thinking presumes that the organization will "hold still" long enough for specification and implementation. This goal has always been difficult, and the concept of the frozen specification has been discredited. Achievement of this goal burdens ISD with parallel analysis, specification and implementation rework as the organization emerges out from under the planned IS. This burden also contributes to the problems of abstract analysis and user dissatisfaction by increasing the front-end expense of ISD projects and increasing the complexity of user-systems analysis interactions.

New system projects denote ISD failure. The fifth obsolete ISD goal is the importance of new-systems project planning. This goal is a holdover from the early ISD projects that replaced manual IS with computer-based IS. This replacement-mentality created a new-systems project orientation in ISD that presumes every IS has a limited lifespan. Emergent IS thinking accepts that every system must evolve continuously, and that all systems must be adapted regularly to their changing environments. A new ISD project arises only from the utter failure of an existing computer-based IS. Under stable systems assumptions, the high value placed on new ISD over maintenance paradoxically implied a high value on the ultimate failure of every IS. (The low value placed on maintenance is most evident in university IS training. Typically, only new ISD is taught.)

ISD Goals for Emergent Organizations

The preceding list of obsolete ISD goals under assumptions of emergent organizations implies that an alternative goal set arises from the alternative assumption set. We will consider four distinct goals that arise from emergent assumptions. This alternative goal set is implied by the assumption set (see Figure 2). In the new goal set the first, second, and last items stand in contrast to the first, second, and fifth items from the revoked set of traditional ISD goals. And the new

third goal is an implied response to the revocation of third and fourth items in the old goal set.

Always analysis. Under emergent assumptions, the analysis of IS applications must be continuous. Since the organization is emerging, the fundamental IS must continuously change and adapt. In order to implement this adaptation, requirements and specifications are constantly renegotiated. Analysis activities are no longer captured within the early stages of a system's life cycle. Instead, these activities are an ongoing service of the organizational ISD group. It is important to realize that this ongoing service must not be cyclical (periods of analysis followed by periods of implementation), but is generally a constant ISD activity in parallel with systems operation and maintenance. The results of this ongoing analysis are continuously fed into the maintenance activities. Because of organizational emergence, the underlying ISD service continuously monitors and reappraises the IS support for every business process and organizational activity. Under this goal, analysis is not a component of an ISD project, but an ongoing ISD organizational maintenance activity.

SYSTEMS SHOULD BE UNDER CONSTANT DEVELOPMENT, CAN NEVER BE FULLY SPECIFIED AND ARE SUBJECT TO CONSTANT ADJUSTMENT AND ADAPTATION.

Dynamic requirements negotiations. Because the organization is emerging around the users, IS requirements can never be fully specified because users are always in conflict with them. Thus user satisfaction is improbable. Indeed, under this assumption, a setting where users are fully satisfied would be an alarming anomaly. Requirements are no longer determined as part of a project, but become a negotiated outcome of the changing characteristics of an emergent organization and the resources for enhancing or altering the existing IS. An emergent ISD goal is not user satisfaction, but a "healthy" degree of conflict between users and their IS. As requirements conflicts rise, increased negotiation and IS enhancement activities are prescribed. As requirements conflicts fall, ISD activities are decreased. The conflict, negotiation and enhancement are continuous service activities provided to support ongoing business processes. These activities are not necessarily associated with any ISD project.

Incomplete and usefully ambiguous specifications. If abstract requirements are largely imaginary, and unambiguous specifications are ineffectual, analysts must come to terms with ambiguity. Because the requirements are in motion, specifications must be kept in a state in which these can be easily adapted for enhancing or modifying the existing system. The goal is a set of specifications wherein each specification is open-ended and easily modified. Complete and unambiguous specifications are only possible for organizations that are totally stable, and waste valuable resources in an emergent setting. System enhancement and modification activities begin to be undertaken even though the specifications are incomplete and ambiguous. These activities "succeed" because they are themselves never completed (the organization is likely to emerge from under the planned enhancements or modifications). Traditionally, the IS is a consequence of the specification. Under the emergent view, the specification is just as equally a consequence of the IS emergence. This parallel emergence leads to both an IS and an ISD process that are incomplete and usefully ambiguous. These last two characteristics represent an excellent foundation for further organizational emergence.

Continuous redevelopment. Under emergent assumptions, this goal supplants the current ISD project mentality under which all systems terminate at their obsolescence point. The goal of ISD is to preserve all existing IS applications by continuously enhancing and modifying these to match organizational requirements. The goal of ISD is to prevent system obsolescence and thereby eliminate system termination (and the implied new ISD project). The U.S. railroad system provides a metaphor to illustrate how this ISD approach operates. Today's railroad systems no longer resemble the railroads of a century ago. The engines, rolling stock, tracks, stations, and signaling have all been replaced with modern elements. There has not been a nationwide development project to replace the entire railroad system. Instead, the railroad system has emerged to match the needs of the nation and the limits of the technology. This emergence is a consequence of continuous enhancements: new tracks added in some areas and new rolling stock purchased when needed, for example. The net effect is an adaptive railroad system. Continuous redevelopment implies that information systems are continuously enhanced and modified such that they are never totally outdated and irreparable.

There are two interesting implications of continuous redevelopment. The first implication arises from the viewpoint of life-cycle termination as an anomaly. When an IS becomes too expensive to maintain and must be replaced, there is an implied failure on the part of ISD management. ISD management failed to keep the IS maintained in a state that permitted its further redevelopment. In other words, the IS was allowed to decay beyond its economic rescue point. In an emergent setting, the decayed IS probably imposed a long period of rising stable systems drag that limited the organizational ability to emerge. Had the system been continuously redeveloped, the drag would have been reduced and the system life span extended indefinitely. In most traditional ISD organizations, the resources that might be used for continuous redevelopment are paradoxically occupied with system replacement projects.

The second interesting implication regards legacy systems and the infamous Y2K problem. These two interconnected problems have risen in importance over the last decade. To a degree, both of these result from the preservation of the 1960s and 1970s ISD project mentality into the 1980s and 1990s. The new systems projects consumed the resources that might have otherwise been applied in gradually redeveloping, enhancing, and modifying these old systems. Under continuous redevelopment, these systems, like the national railroad system, could not be legacy systems. Over the 1980s and 1990s, these legacy systems should have evolved, but didn't. Today's ISD managers are now confronted with (and blamed for) the failures of their predecessors.

Adaptability orientation. The essential impact of the emergent goal set on ISD relates to the adaptability of IS. Recognizing that IS must undergo continuous redevelopment, the ISD approach and the underlying IS architecture must be conducive to redevelopment. Ease of modification must be deeply embedded in every IS. This easy modification implies that every system includes explicit ISD mechanisms by which the system can adapt. An interesting implication of this goal is the merger of IS and ISD. Development of an IS is exactly the same activity as maintenance, and is equally an essential component of IS operation. The distinction between IS and ISD disappears because the emergence of IS is embodied by the goal set of emergent ISD—an emergent IS is ISD.

Ways of Supporting the New ISD Goals

The existing vehicles for supporting an effort to reach the emergent organization goals include easily maintainable specifications, open systems interconnection architectures, prototyping, and end-user development. Easily maintainable specifications, like object-oriented designs, make it easier and cheaper to respecify IT systems when change is needed. Open systems architectures enable IT components to be easily rearranged and incorporated with newly developed components. Prototypes, particularly operational prototypes, are typically built with tools that enable easy changes. End-user development uses productivity tools to create inexpensive applications that can be thought of as disposable systems. These existing tools have a role in supporting emergent organizations, but these alone do not go far enough. Several IT organizational capabilities can also help.

Back channel communications for ISD professionals. Back channels, such as guaranteed privacy for email, chat rooms, and groupware, permit developers to establish versions of the organizational identity or reality that conflict with other versions. This conflict is important for autopoiesis and emergence. These channels should extend beyond the ISD group and into the users with whom they may interact in order to continuously redevelop systems.

Emergent IT organizations. The IT organization itself must be highly emergent. One element that can promote this emergence is virtual teams that extend to include users. These teams lack the history that confines their adaptation, and eliminate the boundary between user and developer. Another important element is the elimination of the "project" as the primary means of organizing IT activities. An emergent IT organization replaces projects with "streams" of redevelopment activity that are continuous as long as the particular IT system requirement is present. A new project represents the failure of the IT organization to properly adapt the systems in its charge.

Proper rewards system. The IT organization that supports emergent organizations must value system adaptation. Initially developing adaptable systems is important. However, most of the organization's important development activities are merged with its maintenance activities. Maintenance needs to become innovative and linked to the changing goal set of the organization. This shifting set of values recognizes high-maintenance activities as the mark of an excellent system for an emergent organization.

Summary

Blame for the systems development crisis has been laid at the feet of the creators of development methods, tool builders, analysts, designers and implementers. But we suggest that the problem may, instead, lie in an incorrect goal set that we all have accepted from the outset that is the idea that systems

should support organizational stability and structure, should be low maintenance, and should strive for high degrees of user acceptance. We propose an alternative view that assumes systems should be under constant development, can never be fully specified and, like the organizations for which they are built, are subject to constant adjustment and adaptation.

Since organizational change has become so important to organizational survival, IT systems must also incorporate continuous change. This incorporation goes beyond adaptable systems, and includes creating support for organizations that cannot help but emerge. Continuous change implies replacement of traditional ISD values. These outmoded values include long IT system life spans, dependence on user acceptance, concise specifications, and complete systems analysis. Emergent IT organizations value continuous analysis, negotiated requirements, and a large portfolio of continuous maintenance activities. C

References

1. Baskerville, R., Travis, J., and Truex, D.P. Systems without method: The impact of new technologies on information systems development projects. In K.E. Kendall, K. Lyytinen, and J.I. DeGross, Eds., *Transactions on the Impact of Computer Supported Technologies in Information Systems Development*. Elsevier, Amsterdam, 1992.
2. Benson, J.K. Organizations: A dialectical view. *Administrative Science Quarterly 22*, (Mar. 1977).
3. Berger, P.L. and Luckmann, T. *The Social Construction of Reality: A Treatise in the Sociology of Knowledge*. Anchor Press, NY, 1966.
4. Jantsch, E. *The Self Organizing Universe: Scientific and Human Implications of the Emerging Paradigm of Evolution.* Pergamon Press, NY, 1980.
5. Luhmann, N. The autopoiesis of social systems, in F. Geyer and J. van der Zouwen, Eds., *Sociocybernetic Paradoxes*, Sage Publications, London, 1986.
6. Orlikowski, W.J. Improvising organizational transformation over time: A situated change perspective. *Information Systems Research 7*, 1 (1996), 63–92.
7. Truex, D.P. and Klein, H.K. A rejection of structure as a basis for information systems development. In R.K. Stamper, P. Kerola, R. Lee, and K. Lyytinen, Eds., *Collaborative Work, Social Communications, and Information Systems*. Elsevier, Amsterdam, 1991.
8. von Foerster, H. Principles of self-organization in a socio-managerial context. In H. Ulrich and G.J.B. Probst, Eds., *Self-Organization and Management of Social Systems*, Springer-Verlag, Berlin, 1984.

Duane P. Truex (dtruex@gsu.edu) is an assistant professor in the Computer Information Systems Department at Georgia State University.
Richard Baskerville (baskerville@acm.org) is an associate professor in the Computer Information Systems Department at Georgia State University.
Heinz Klein (hkklein@binghamton.edu) is an associate professor in the School of Management at Binghamton University, NY.