The New New Enterprise Game
OOP 2025

Rules

Leadership

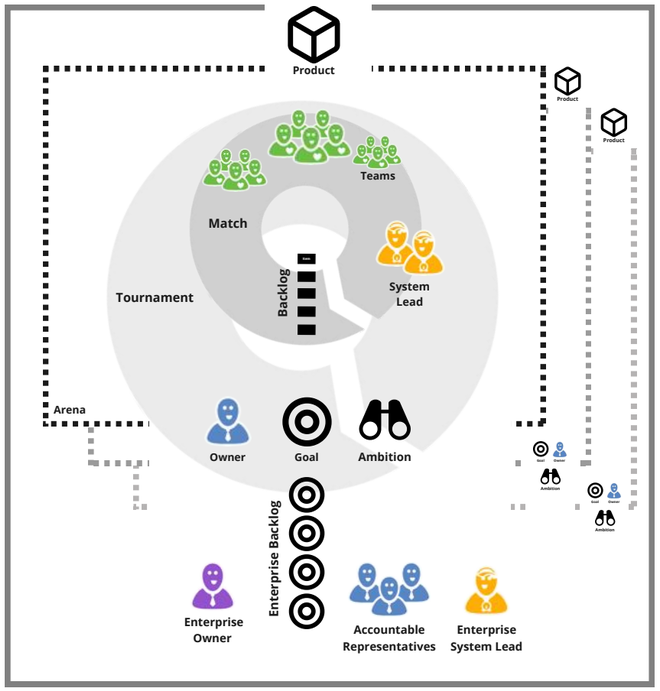
GAME 3

Strategy

DAS SCRUMTEAM

Leading the ever-faster Evolution of tomorrow's Enterprises

System

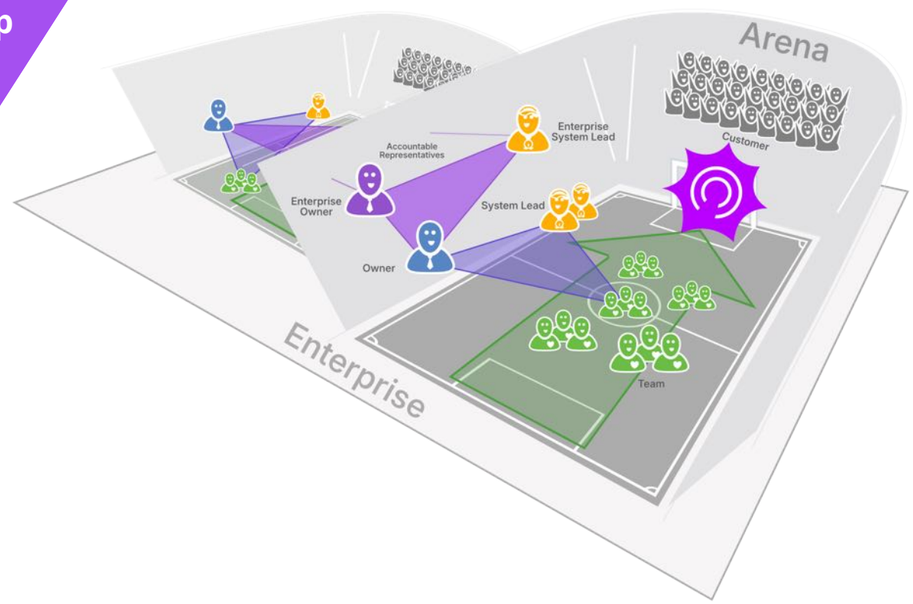


Rules

Leadership

GAME3

Strategy

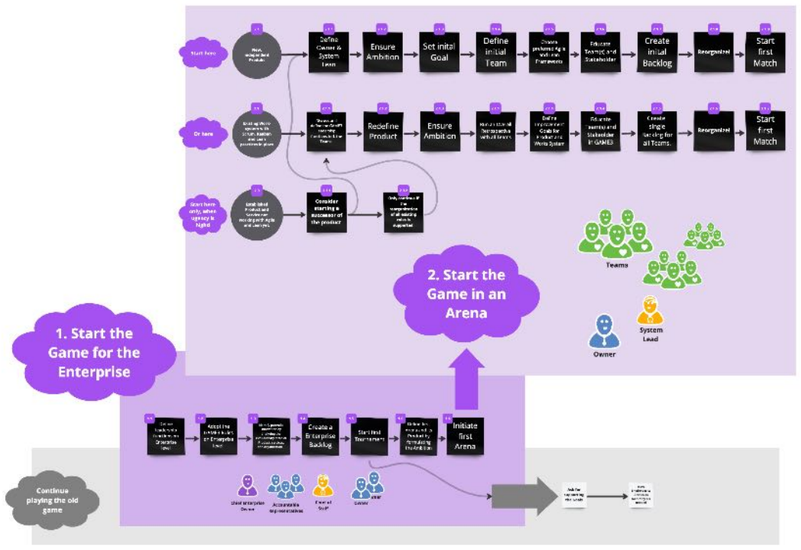


Empirical Controlled

Evolution Focused

Global Optimized

Playbook



Interplay

LeSS

Kanban

Scrum@Scale

Scrum

Wardley Mapping

Cynefin

eXtreme Programming

Product Discovery

TECHNISCHE UNIVERSITÄT DARMSTADT

WEB.DE

DAS SCRUMTEAM

scaledprinciples.org

ScALeD Agile Lean Development - The Principles

Agile methods are becoming ever more popular, and a growing number of companies has adopted agile practices on a large scale. But successfully scaling agility is challenging. As companies, projects and teams differ, there is no silver bullet solution to...

GAME 3

1. **Why**

2. **How it works**

3. **How to start**

Game 1

ABSTRACT. *The stated, accepted philosophy for systems development is that the development process is a well understood approach that can be planned, estimated, and successfully completed. This has proven incorrect in practice. SCRUM assumes that the systems development process is an unpredictable, complicated process that can only be roughly described as an overall progression. SCRUM defines the systems development process as a loose set of activities that combines known, workable tools and techniques with the best that a development team can devise to build systems. Since these activities are loose, controls to manage the process and inherent risk are used. SCRUM is an enhancement of the commonly used iterative/incremental object-oriented development cycle.*

KEY WORDS: *SCRUM SEI Capability-Maturity-Model Process Empirical*

W. Edwards Deming

Hirotaka **Takeuchi** and Ikujiro **Nonaka**

The New New Product Development Game

1986, Harvard Business Review

hbr.org

Harvard Business Publishing

The New New Product Development Game

In today's fast-paced, fiercely competitive world of commercial new product development, speed and flexibility are essential. Companies are increasingly realizing that the old, sequential approach to developing new products simply won't get the job done...

GAME3

The New New Enterprise Game - GAME3

Game 2

SCRUM Development Process

Ken Schwaber

Advanced Development Methods
131 Middlesex Turnpike Burlington, MA 01803
email virman@aol.com Fax: (617) 272-0555

***ABSTRACT**. The stated, accepted philosophy for systems development is that the development process is a well understood approach that can be planned, estimated, and successfully completed. This has proven incorrect in practice. SCRUM assumes that the systems development process is an unpredictable, complicated process that can only be roughly described as an overall progression. SCRUM defines the systems development process as a loose set of activities that combines known, workable tools and techniques with the best that a development team can devise to build systems. Since these activities are loose, controls to manage the process and inherent risk are used. SCRUM is an enhancement of the commonly used iterative/incremental object-oriented development cycle.*

***KEY WORDS:** SCRUM SEI Capability-Maturity-Model Process Empirical*

Ken Schwaber & Jeff Sutherland

The Scrum Guide

The Definitive Guide to Scrum: The Rules of the Game

SCRUM Development Process

Ken Schwaber

Advanced Development Methods
131 Middlesex Turnpike Burlington, MA 01803
email virman@aol.com Fax: (617) 272-0555

***ABSTRACT**. The stated, accepted philosophy for systems development is that the development process is a well understood approach that can be planned, estimated, and successfully completed. This has proven incorrect in practice. SCRUM assumes that the systems development process is an unpredictable, complicated process that can only be roughly described as an overall progression. SCRUM defines the systems development process as a loose set of activities that combines known, workable tools and techniques with the best that a development team can devise to build systems. Since these activities are loose, controls to manage the process and inherent risk are used. SCRUM is an enhancement of the commonly used iterative/incremental object-oriented development cycle.*

***KEY WORDS:** SCRUM SEI Capability-Maturity-Model Process Empirical*

1. Introduction

In this paper we introduce a development process, SCRUM, that treats major portions of systems development as a controlled black box. We relate this to complexity theory to show why this approach increases flexibility and produces a system that is responsive to both initial and additional requirements discovered during the ongoing development.

Numerous approaches to improving the systems development process have been tried. Each has been touted as providing "significant productivity improvements." All have failed to produce dramatic improvements.[1] As Grady Booch noted, "We often call this condition the software crisis, but frankly, a malady that has carried on this long must be called normal."[2]

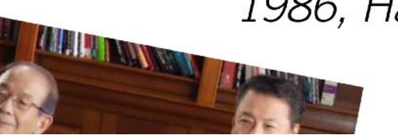
Concepts from industrial process control are applied to the field of systems development in this paper. Industrial process control defines processes as either "theoretical" (fully defined) or "empirical" (black box). When a black box process is treated as a fully

[1] Brooks, F.P. "No silver bullet—essence and accidents of software engineering," Computer 20:4:10-19, April 1987.
[2] Object Oriented Analysis and Design with Applications, p. 8, Grady Booch, The Benjamin/Cummings Publishing Company, Inc., 1994

me 1

Hirotaka **Takeuchi** and Ikujiro **Nonaka**

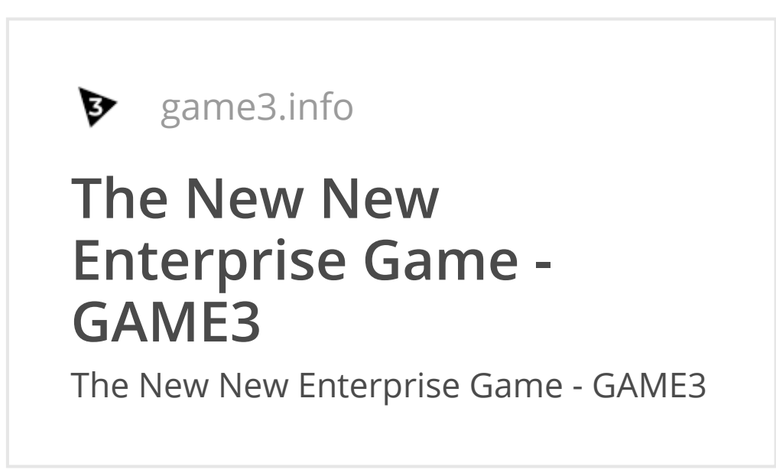
The New New Product Development Game
1986, Harvard Business Review



hbr.org

Harvard Business

GAME3



Generic
Adoptable
Metaframework for
Empirical-based
Enterprise
Evolution

Game 2


SCRUM Development Process
Ken Schwaber


Ken Schwaber & Jeff Sutherland
The Scrum Guide

1. **Why**
2. **How it works**
3. **How to start**

GAME3

game3.info

The New New Enterprise Game - GAME3

The New New Enterprise Game - GAME3

Generic
Adoptable
Metaframework for
Empirical-based
Enterprise
Evolution

Game 2

SCRUM Development Process

Ken Schwaber

Advanced Development Methods
131 Middlesex Turnpike Burlington, MA 01803
email virman@aol.com Fax: (617) 272-0555

***ABSTRACT**. The stated, accepted philosophy for systems development is that the development process is a well understood approach that can be planned, estimated, and successfully completed. This has proven incorrect in practice. SCRUM assumes that the systems development process is an unpredictable, complicated process that can only be roughly described as an overall progression. SCRUM defines the systems development process as a loose set of activities that combines known, workable tools and techniques with the best that a development team can devise to build systems. Since these activities are loose, controls to manage the process and inherent risk are used. SCRUM is an enhancement of the commonly used iterative/incremental object-oriented development cycle.*

***KEY WORDS:** SCRUM SEI Capability-Maturity-Model Process Empirical*

Ken Schwaber & Jeff Sutherland

The Scrum Guide

The Definitive Guide to Scrum: The Rules of the Game

Game 1

W. Edwards Deming

Hirotaka **Takeuchi** and Ikujiro **Nonaka**

The New New Product Development Game
1986, Harvard Business Review

hbr.org

Harvard Business Publishing

The New New Product Development Game

In today's fast-paced, fiercely competitive world of commercial new product development, speed and flexibility are essential. Companies are increasingly realizing that the old, sequential approach to developing new products simply won't get the job done....

1. Start the Game for the Enterprise

- 1.1 Define leadership functions on Enterprise level
- 1.2 Adopt the GAME3 Rules on Enterprise level
- 1.3 Identify potential innovation by analyzing the evolutionary state of Product, services, and organization.
- 1.4 Create a Enterprise Backlog
- 1.5 Start first Tournament
- 1.6 Define first Arena and its Product by formulating the Ambition
- 1.6 Initiate first Arena

Chief Enterprise Owner

Accountable Representatives

Chief of Staff

Owner

Ask for supporting the Goals

Move Employees to Arenas as soon they are needed

Continue playing the old game

2. Start the Game in an Arena

Start here

2.1 New, independent Produkt

- 2.1.1 Define Owner & System Lean
- 2.1.2 Ensure Ambition
- 2.1.3 Set inital Goal
- 2.1.4 Define initial Team
- 2.1.5 Choose preferred Agile and Lean Frameworks
- 2.1.6 Educate Team(s) and Stakeholder
- 2.1.7 Create inital Backlog
- 2.1.8 **Reorganize!**
- 2.1.9 Start first Match

Or here

2.1 Existing Work-system with Scrum, Kanban and Lean practices in place

- 2.1.1 Discuss and define the GAME3 leadership functions incl. the Teams.
- 2.1.2 Redefine Product
- 2.1.3 Ensure Ambition
- 2.1.1 Run an Overall Retrospective with all Teams
- 2.1.1 Define Improvement Goals for Product and Works System
- 2.1.4 Educate Team(s) and Stakeholder in GAME3
- 2.1.5 Create single Backlog for all Teams.
- 2.1.6 **Reorganize!**
- 2.1.7 Start first Match

Start here only, when ugency is hight!

2.3 Established Product and Service not working with Agile and Lean yet.

- 2.3.1 **Consider starting a successor of the product**
- 2.3.2 **Only continue if the reorganization of all existing roles is supported!**

Teams

System Lead

Owner

GAME 3

Leadership

System Leads lead to an **effective work system.** They serve the enterprise by:

- Developing competencies of Teams and people.
- Facilitating decision-making.
- Sustaining a continuous cycle of Planning, Execution, Inspection, and Adaptation.

Owners lead to the **success of the product and services.** She serves the enterprise by:

- Balancing opportunities and risks.
- Focusing the organization to increase effectiveness.
- Ensuring decisions are made.

Teams lead to **customer satisfaction.** They serve the enterprise by:

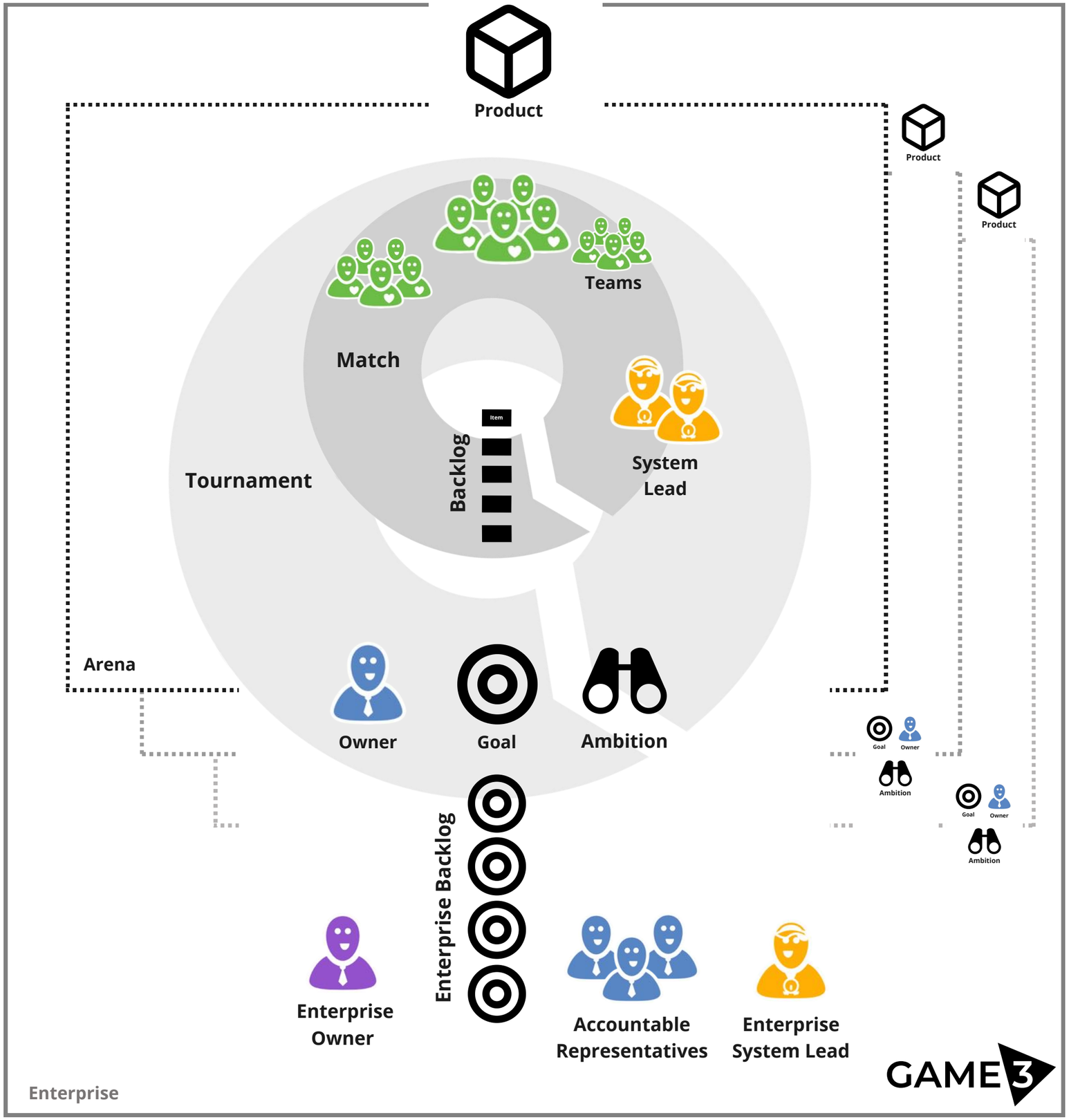
- Managing and executing the work.
- Creating value and ensuring quality.
- Identifying opportunities for improvement in products, services, and work systems.

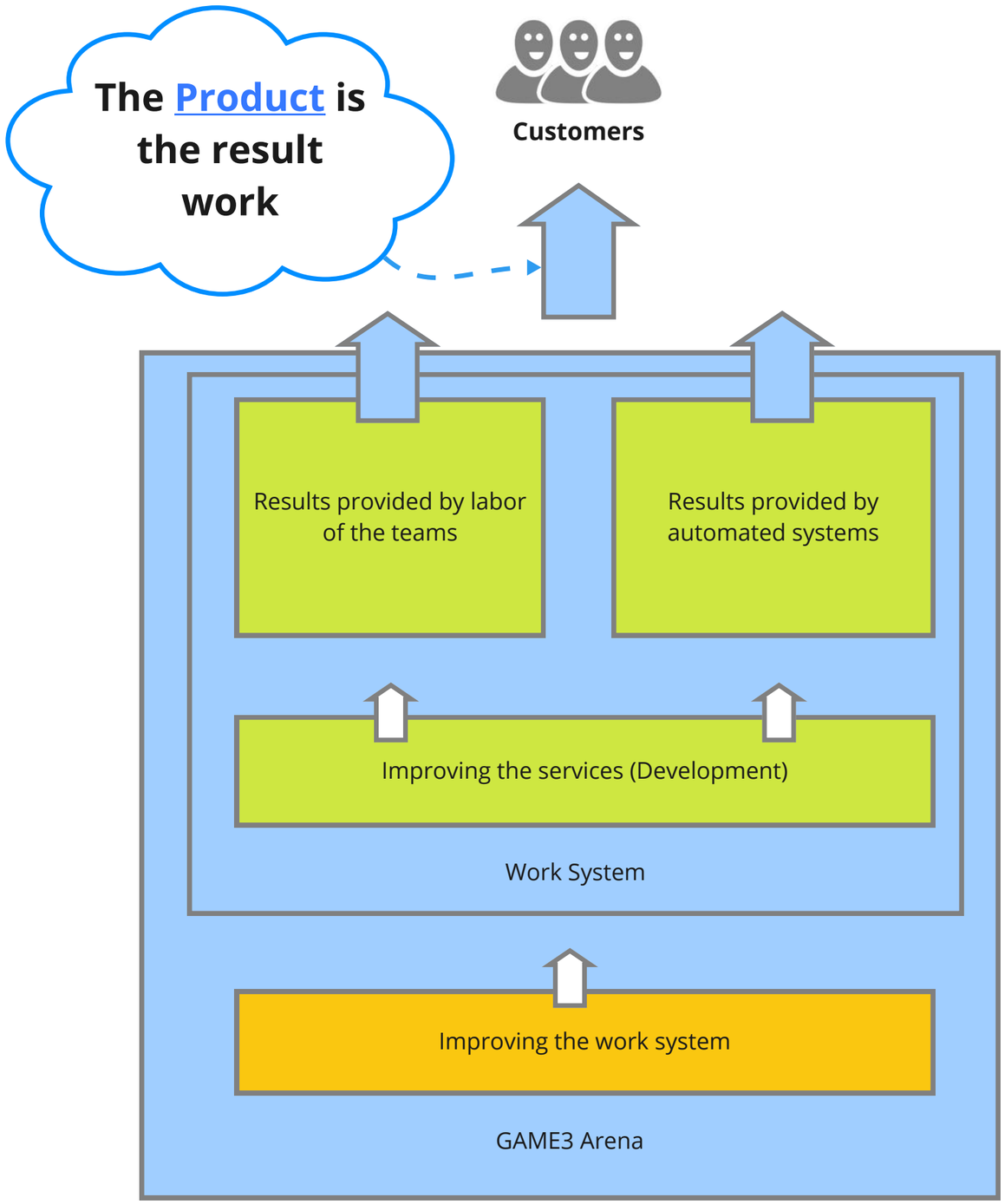
GAME3

game3.info

Rules - GAME3

Rules - GAME3




The Product is the result work
Customers
Results provided by labor of the teams
Results provided by automated systems
Improving the services (Development)
Work System
Improving the work system
GAME3 Arena

Conway's law

HOW DO COMMITTEES INVENT?

by MELVIN E. CONWAY

That kind of intellectual activity which creates a useful whole from its diverse parts may be called the *design* of a *system*. Whether the particular activity is the creation of specifications for a major weapon system, the formation of a recommendation to meet a social challenge, or the programming of a computer, the general activity is largely the same.

Typically, the objective of a design organization is the creation and assembly of a document containing a coherently structured body of information. We may name this information the *system design*. It is typically produced for a sponsor who usually desires to carry out some activity guided by the system design. For example, a public official may wish to propose legislation to avert a recurrence of a recent disaster, so he appoints a team to explain the catastrophe. Or a manufacturer needs a new product and designates a product planning activity to specify what should be introduced.

The design organization may or may not be involved in the construction of the system it designs. Frequently, in public affairs, there are policies which discourage a group's acting upon its own recommendations, whereas, in private industry, quite the opposite situation often prevails.

It seems reasonable to suppose that the knowledge that one will have to carry out one's own recommendations or that this task will fall to others, probably affects some design choices which the individual designer is called upon to make. Most design activity requires continually making choices. Many of these choices may be more than design decisions; they may also be personal decisions the designer makes about his own future. As we shall see later, the incentives which exist in a conventional management environment can motivate choices which subvert the intent of the sponsor.[1]

stages of design

The initial stages of a design effort are concerned more with structuring of the design activity than with the system itself.[2] The full-blown design activity cannot proceed until certain preliminary milestones are passed. These include:

1. Understanding of the boundaries, both on the design activity and on the system to be designed, placed by the sponsor and by the world's realities.
2. Achievement of a preliminary notion of the system's organization so that design task groups can be meaningfully assigned.

We shall see in detail later that the very act of organizing a design team means that certain design decisions have already been made, explicitly or otherwise. Given any design team organization, there is a class of design alternatives which cannot be effectively pursued by such an organization because the necessary communication paths do not exist. Therefore, there is no such thing as a design group which is both organized and unbiased.

Once the organization of the design team is chosen, it is possible to delegate activities to the subgroups of the organization. Every time a delegation is made and somebody's scope of inquiry is narrowed, the class of design alternatives which can be effectively pursued is also narrowed.

Once scopes of activity are defined, a coordination problem is created. Coordination among task groups, although it appears to lower the productivity of the individual in the small group, provides the only possibility that the separate task groups will be able to consolidate their efforts into a unified system design.

Thus the life cycle of a system design effort proceeds through the following general stages:

1. Drawing of boundaries according to the ground rules.
2. Choice of a preliminary system concept.
3. Organization of the design activity and delegation of tasks according to that concept.
4. Coordination among delegated tasks.
5. Consolidation of subdesigns into a single design.

It is possible that a given design activity will not proceed straight through this list. It might conceivably reorganize upon discovery of a new, and obviously superior, design concept; but such an appearance of uncertainty is unflattering, and the very act of voluntarily abandoning a creation is painful and expensive. Of course, from the

Dr. Conway is manager, peripheral systems research, at Sperry Rand's Univac Div., where he is working on recognition of continuous speech. He has previously been a research associate at Case Western Reserve Univ., and a software consultant. He has an MS in physics from CalTech and a PhD in math from Case.

[1] A related, but much more comprehensive discussion of the behavior of system-designing organizations is found in John Kenneth Galbraith's, *The New Industrial State* (Boston, Houghton Mifflin, 1967). See especially Chapter VI, "The Technostructure."

[2] For a discussion of the problems which may arise when the design activity takes the form of a project in a functional environment, see C. J. Middleton, "How to Set Up a Project Organization," *Harvard Business Review*, March-April, 1967, p. 73.

manufacturers whose programmers and engineers bear a similar relationship.)

system management

The structures of large systems tend to disintegrate during development, qualitatively more so than with small systems. This observation is strikingly evident when applied to the large military information systems of the last dozen years; these are some of the most complex objects devised by the mind of man. An activity called "system management" has sprung up partially in response to this tendency of systems to disintegrate. Let us examine the utility to system management of the concepts we have developed here.

Why do large systems disintegrate? The process seems to occur in three steps, the first two of which are controllable and the third of which is a direct result of our homomorphism.

First, the realization by the initial designers that the system will be large, together with certain pressures in their organization, make irresistible the temptation to assign too many people to a design effort.

Second, application of the conventional wisdom of management to a large design organization causes its communication structure to disintegrate.

Third, the homomorphism insures that the structure of the system will reflect the disintegration which has occurred in the design organization.

Let us first examine the tendency to overpopulate a design effort. It is a natural temptation of the initial designer—the one whose preliminary design concepts influence the organization of the design effort—to delegate tasks when the apparent complexity of the system approaches his limits of comprehension. This is the turning point in the course of the design. Either he struggles to reduce the system to comprehensibility and wins, or else he loses control of it. The outcome is almost predictable if there is schedule pressure and a budget to be managed.

A manager knows that he will be vulnerable to the charge of mismanagement if he misses his schedule without having applied all his resources. This knowledge creates a strong pressure on the initial designer who might prefer to wrestle with the design rather than fragment it by delegation, but he is made to feel that the cost of risk is too high to take the chance. Therefore, he is forced to delegate in order to bring more resources to bear.

The following case illustrates another but related way in which the environment of the manager can be in conflict with the integrity of the system being designed.

A manager must subcontract a crucial and difficult design task. He has a choice of two contractors, a small new organization which proposes an intuitively appealing approach for much less money than is budgeted, and an established but conventional outfit which is asking a more "realistic" fee. He knows that if the bright young organization fails to produce adequate results, he will be accused of mismanagement, whereas if the established outfit fails, it will be evidence that the problem is indeed a difficult one.

What is the difficulty here? A large part of it relates to the kind of reasoning about measurement of resources which arises from conventional accounting theory. According to this theory, the unit of resource is the dollar, and all resources must be measured using units of measurement which are convertible to the dollar. If the resource is human effort, the unit of measurement is the number of hours worked by each man times his hourly cost, summed up for the whole working force.

One fallacy behind this calculation is the property of *linearity* which says that two men working for a year or one hundred men working for a week (at the same hourly cost per man) are resources of equal value. Assuming that two men and one hundred men cannot work in the same organizational structure (this is intuitively evident and will be discussed below) our homomorphism says that they will not design similar systems; therefore the value of their efforts may not even be comparable. From experience we know that the two men, if they are well chosen and survive the experience, will give us a better system. Assumptions which may be adequate for peeling potatoes and erecting brick walls fail for designing systems.

Parkinson's Law[3] plays an important role in the overassignment of design effort. As long as the manager's prestige and power are tied to the size of his budget, he will be motivated to expand his organization. This is an inappropriate motive in the management of a system design activity. Once the organization exists, of course, it will be used. Probably the greatest single common factor behind many poorly designed systems now in existence has been the availability of a design organization in need of work.

The second step in the disintegration of a system design—the fragmentation of the design organization's communication structure—begins as soon as delegation has started. Elementary probability theory tells us that the number of possible communication paths in an organization is approximately half the square of the number of people in the organization. Even in a moderately small organization it becomes necessary to restrict communication in order that people can get some "work" done. Research which leads to techniques permitting more efficient communication among designers will play an extremely important role in the technology of system management.

Common management practice places certain numerical constraints on the complexity of the linear graph which represents the administrative structure of a military-style organization. Specifically, each individual must have at most one superior and at most approximately seven subordinates. To the extent that organizational protocol restricts communication along lines of command, the communication structure of an organization will resemble its administrative structure. This is one reason why military-style organizations design systems which look like their organization charts.

conclusion

The basic thesis of this article is that organizations which design systems (in the broad sense used here) are constrained to produce designs which are copies of the communication structures of these organizations. We have seen that this fact has important implications for the management of system design. Primarily, we have found a criterion for the structuring of design organizations: a design effort should be organized according to the need for communication.

This criterion creates problems because the need to communicate at any time depends on the system concept in effect at that time. Because the design which occurs first is almost never the best possible, the prevailing system concept may need to change. Therefore, flexibility of organization is important to effective design.

Ways must be found to reward design managers for keeping their organizations lean and flexible. There is need for a philosophy of system design management which is not based on the assumption that adding manpower simply adds to productivity. The development of such a philosophy promises to unearth basic questions about value of resources and techniques of communication which will need to be answered before our system-building technology can proceed with confidence. ■

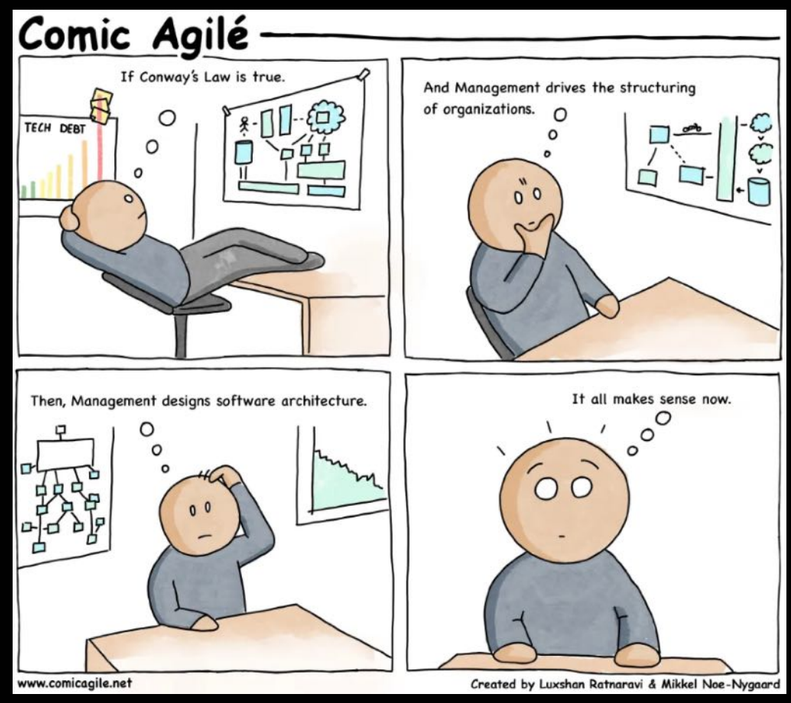
[3] C. Northcote Parkinson, *Parkinson's Law and Other Studies in Administration* (Boston, Houghton Mifflin, 1957).

Conway's law

Organisationen, die Systeme entwerfen, sind darauf beschränkt, Entwürfe zu produzieren, die Kopien der Kommunikationsstruktur dieser Organisationen sind.

Conway's law

"organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations."



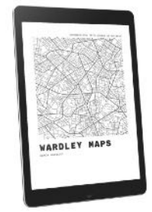
Simon Wardley

@swardley

LEADING EDGE FORUM
ADVANTAGE THROUGH INSIGHT

leadingedgeforum.com

Wardley Mapping



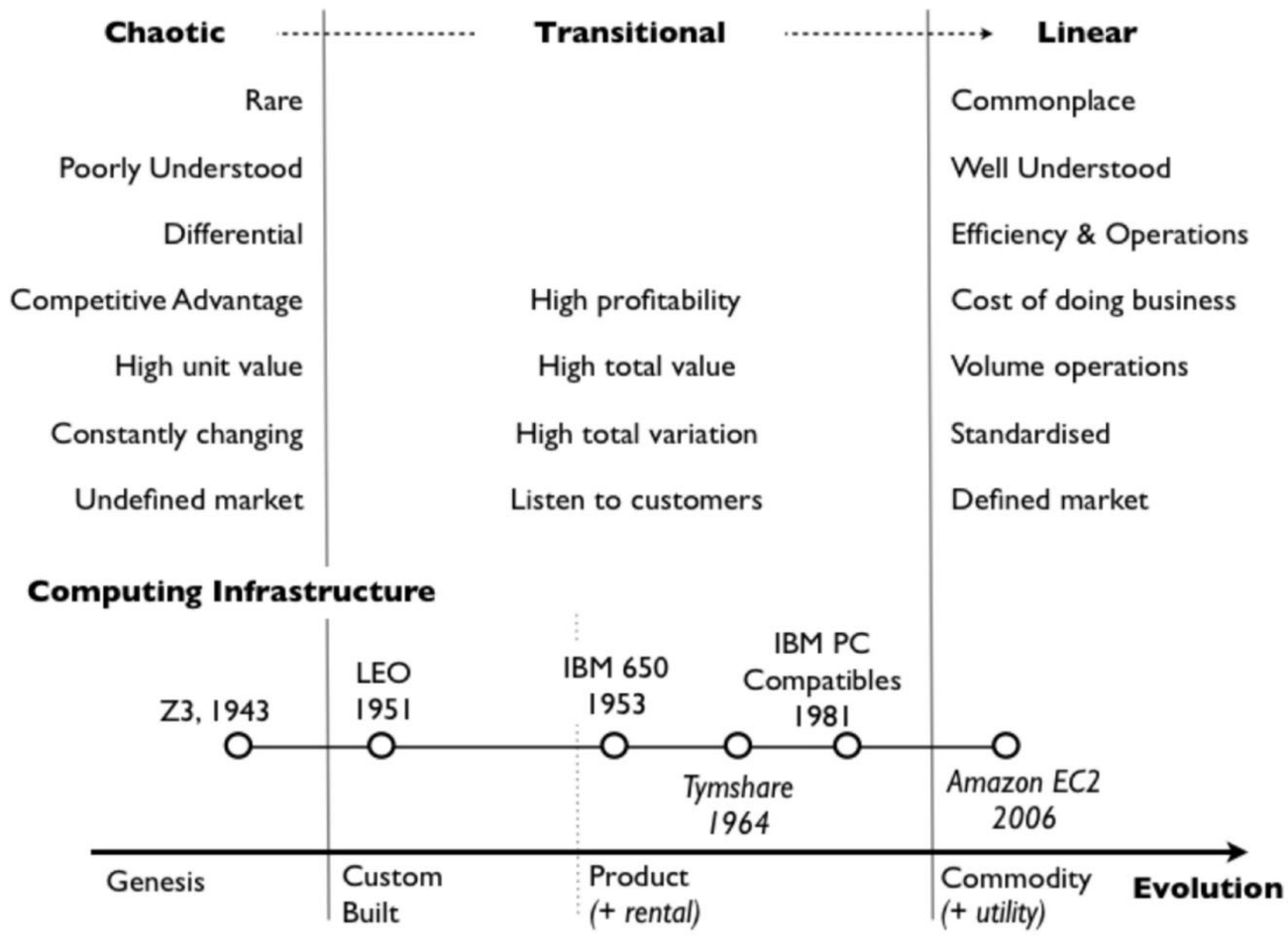
"This is the story of my journey, from a bumbling and confused CEO lost in the headlights of change to having a vague idea of what I was doing."

- Simon Wardley

learnwardleymapping.com

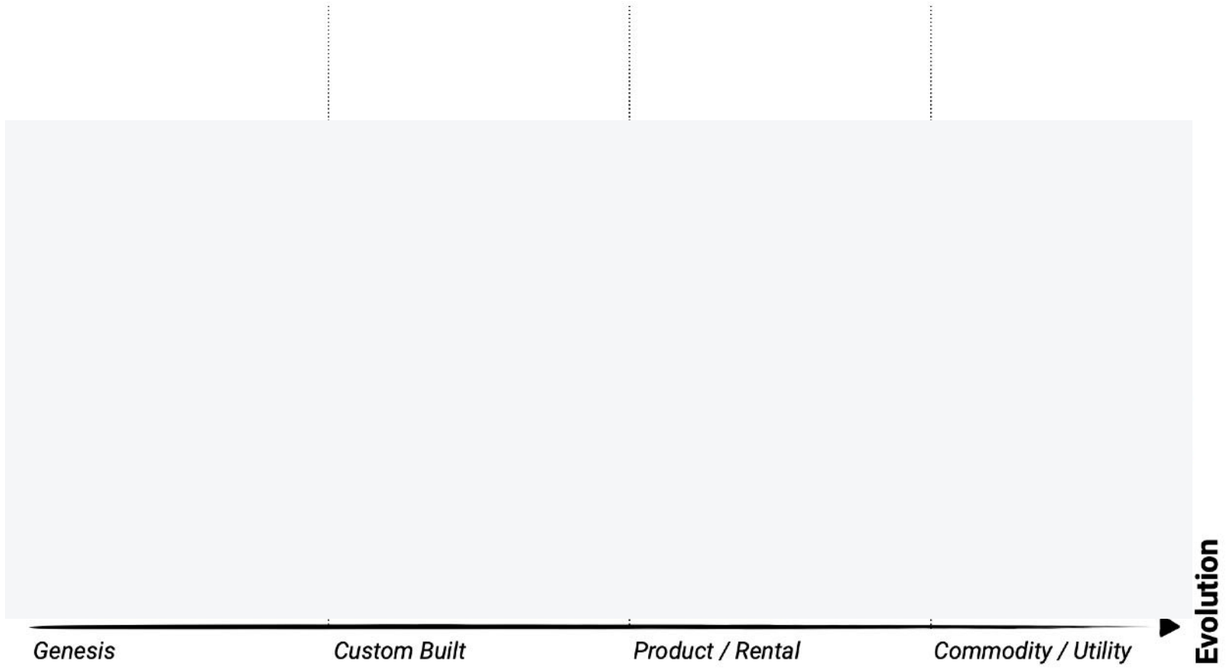
The Book

This is the story of Simon Wardley. Follow his journey from bumbling and confused CEO lost in the headlights of change to someone with a vague idea of what they're doing.



GAME 3

Strategy



1. Die Entscheidungen, die wir heute für unsere Produkte und Dienstleistungen treffen, schaffen die Probleme von morgen.

2. Evolution ist kein konstanter Fluss.

3. Evolution schreitet im Durchschnitt immer schneller voran.

4. Wir können die Evolution nicht aufhalten. Das liegt in der Natur des Menschen. Aber wir können sie **führen**.

Evolution Focused *First*!

Second: **Agile** **Efficiency**