Beyond Source Code

Mike Amundsen,
API Academy / CA
@mamund

Introduction


Ctrl
PgUp
Mike Amundsen
@mamund

API Design Training | API

www.apiacademy.co

API ACADEMY

Search API Academy

API Strategy | API Design | API Management | Resources | About

Register | Sign In

Your Guide to API Design & Implementation Best Practices

API Academy delivers free online lessons and in-person consulting services covering essential API techniques and tools for business managers, interface designers and enterprise architects

What is an API?

Get an overview of what an API is and what it does, to help you realize the business value of APIs

API Design Basics

Understand the API architecture process and learn basic design and implementation best practices

Web API Architectural Styles

Get a detailed overview of the main architectural styles for Web and mobile API design

Choosing a Solution

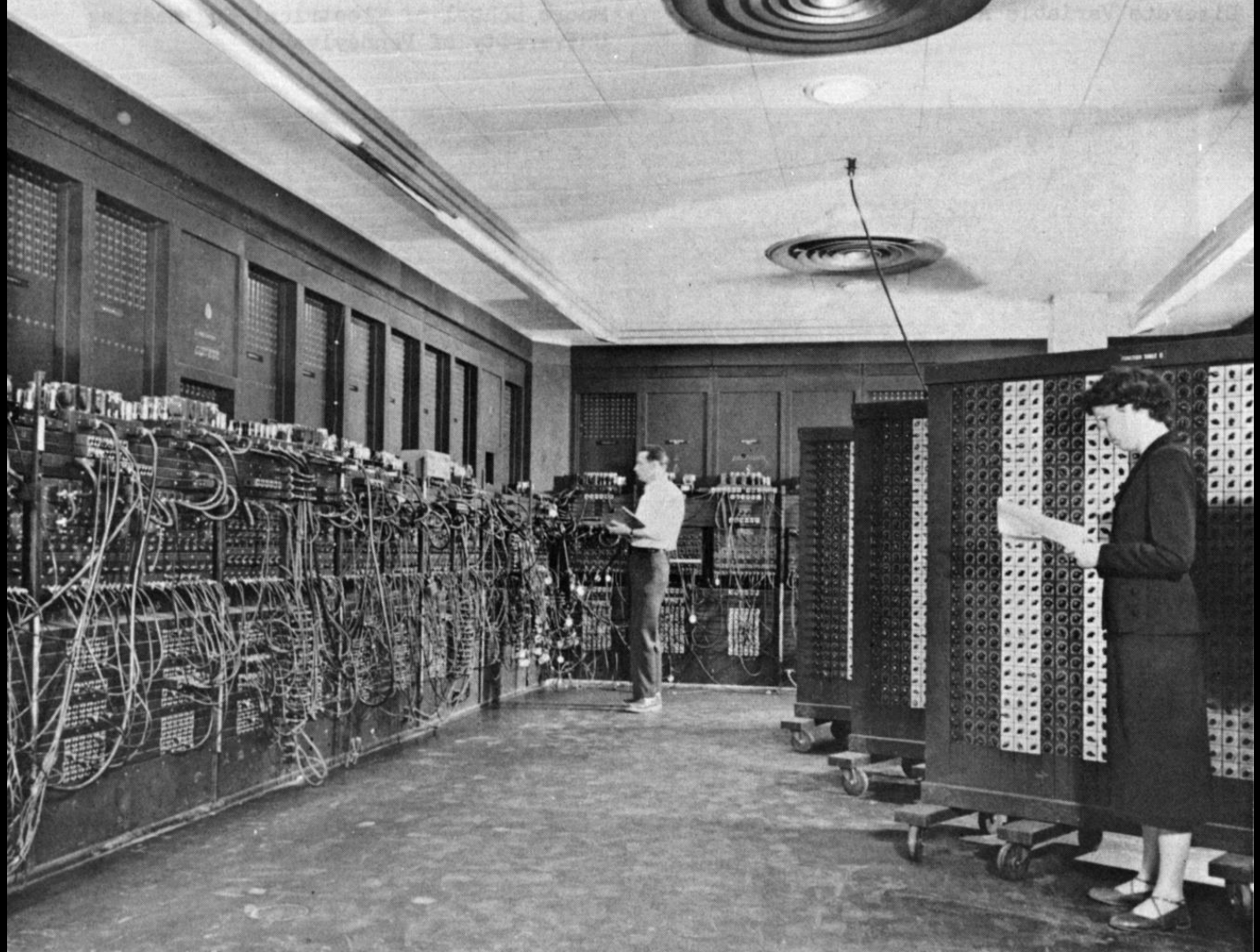
Choose between the various solutions that offer the basic components for enterprise API Management

Beyond Source Code

- Computing
- Communicating
- Scaling
- Artisinal Luddites
- The Future of Code

Computing

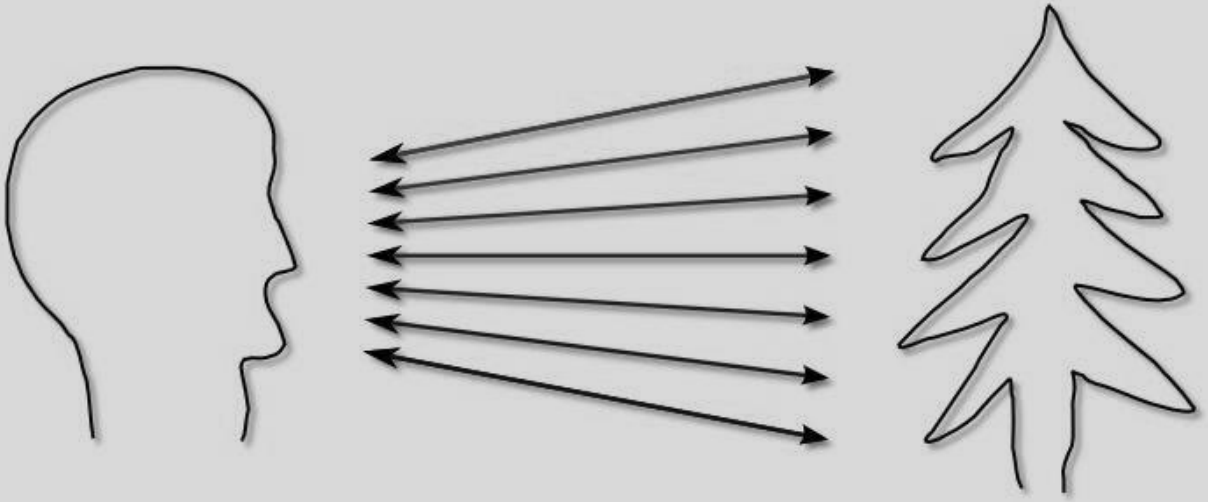

COMPUTING
DIVISION
COMPUTING
SECTION



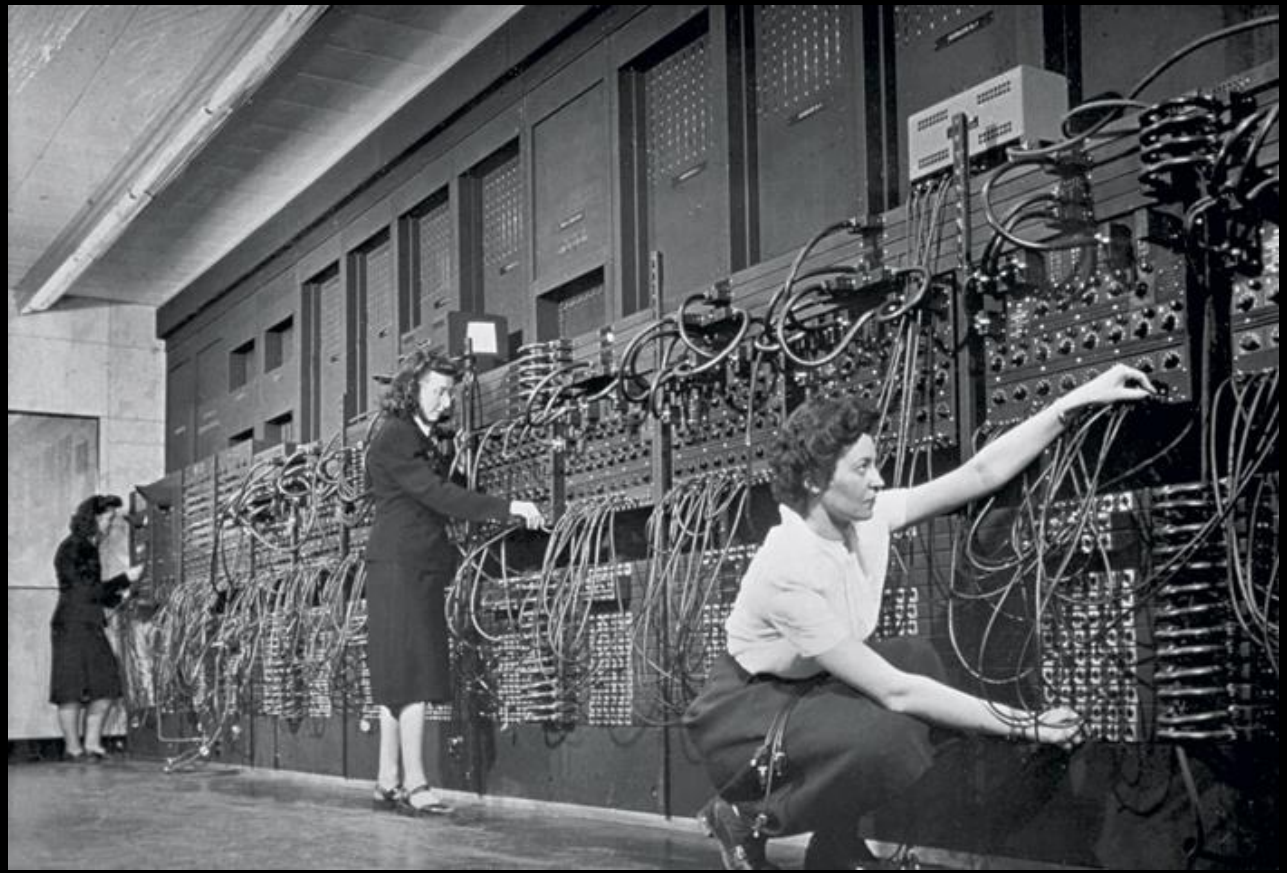
“Figure out how the machine works and then figure out how to program it.”

- Kay McNulty, ENIAC Team

Me **Mental Models** **Reality**



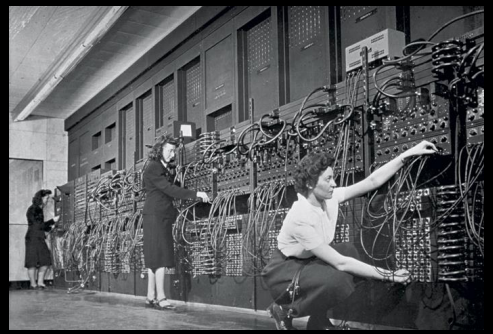
Mental Models affect the way we see and interpret reality. They are like the filters through which we see the world.

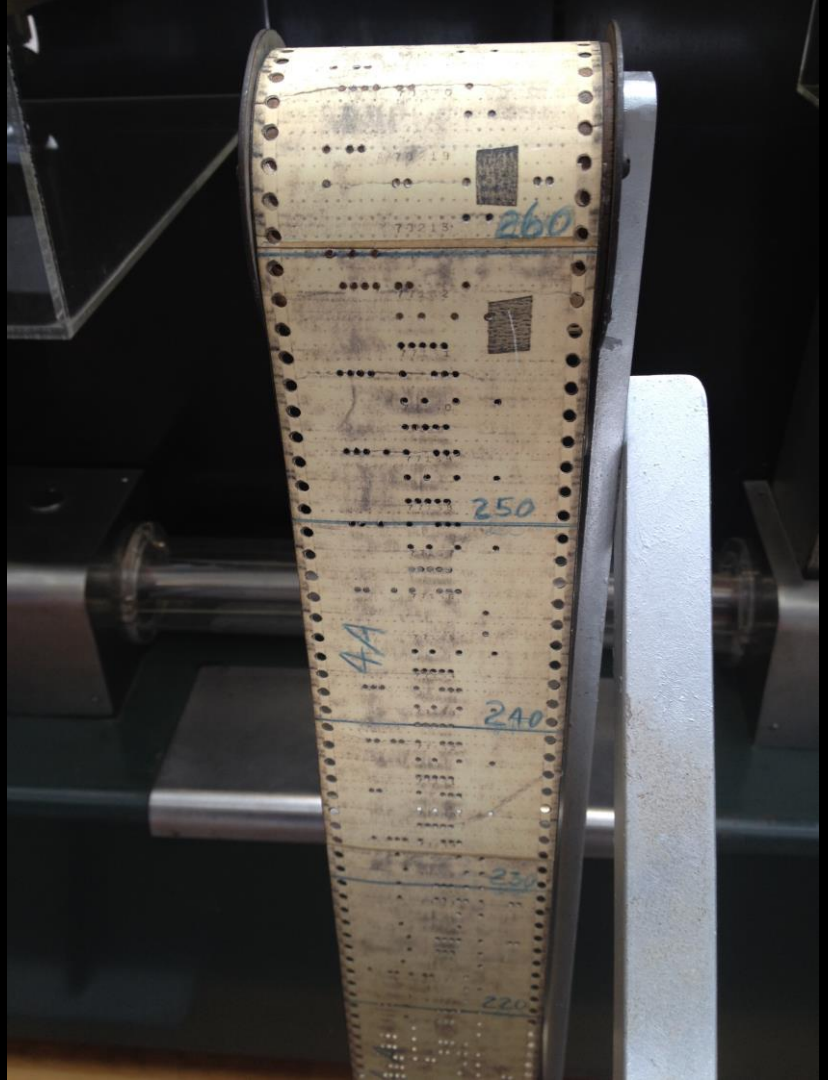


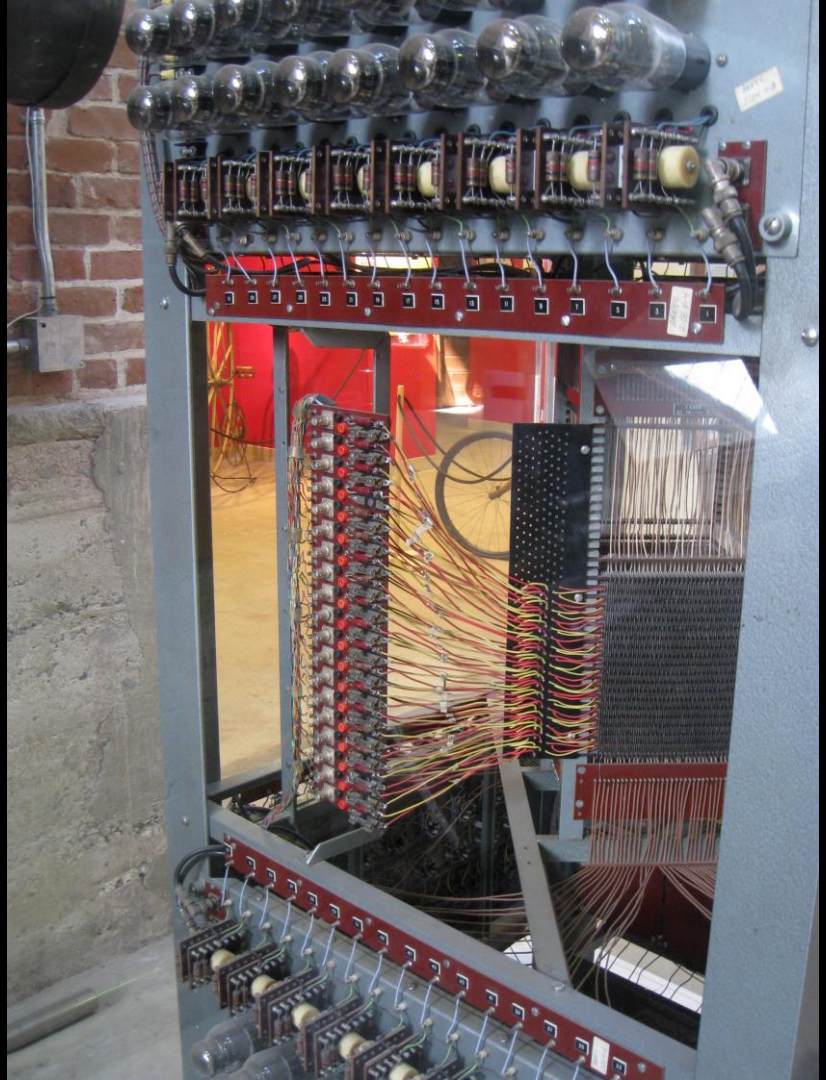
Jennings, Wescoff, & Lichterman, 1946

There was no source code…

There was no source code…
*The program was the **machine**.*

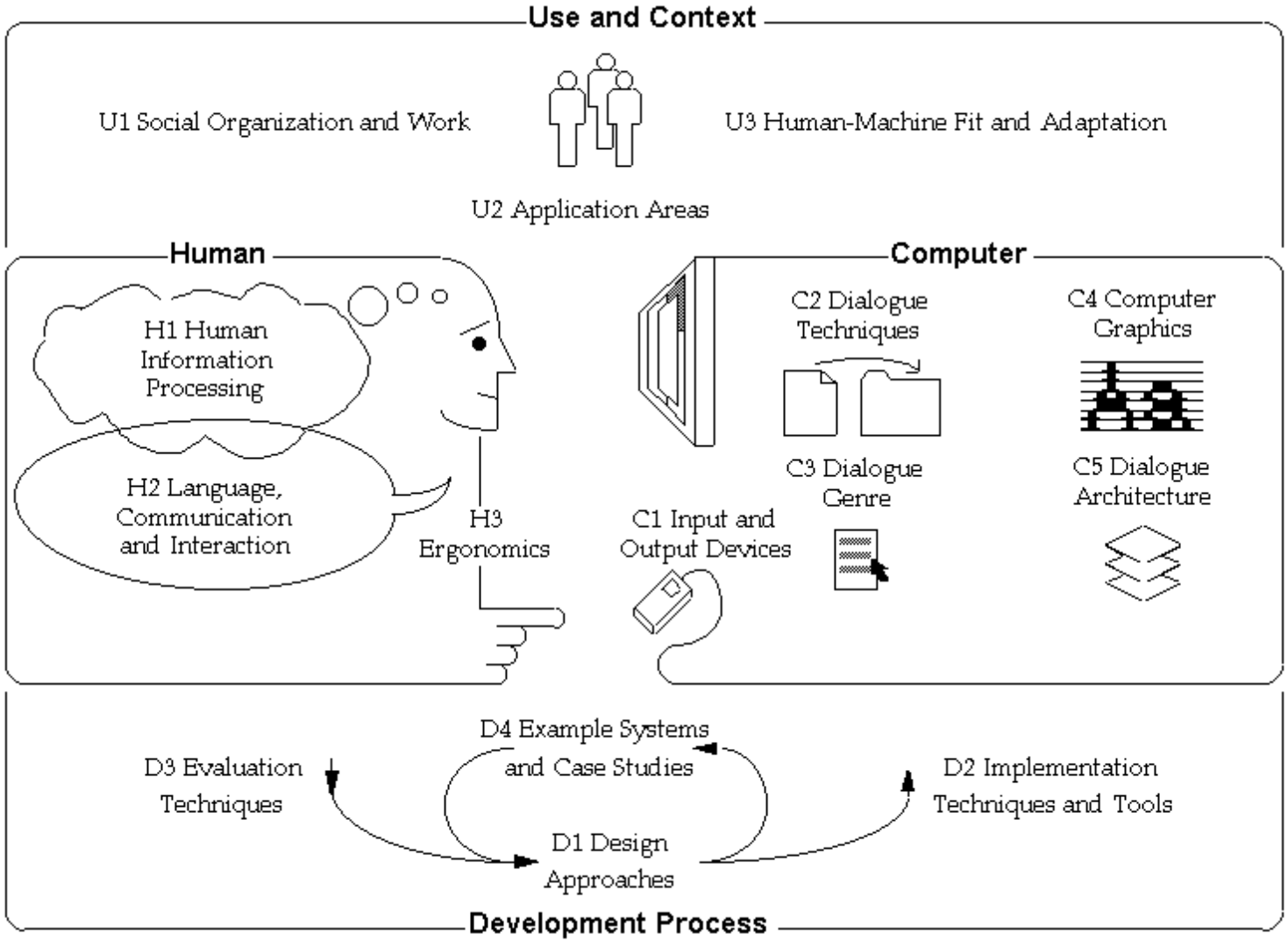



260
250
44
240

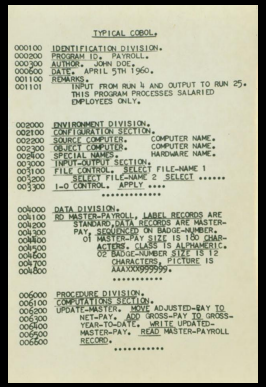


TYPICAL COBOL.

```
000100  IDENTIFICATION DIVISION.
000200  PROGRAM ID.  PAYROLL.
000300  AUTHOR.  JOHN DOE.
000600  DATE.  APRIL 5TH 1960.
001100  REMARKS.
001101      INPUT FROM RUN 4 AND OUTPUT TO RUN 25.
            THIS PROGRAM PROCESSES SALARIED
            EMPLOYEES ONLY.


002000  ENVIRONMENT DIVISION.
002100  CONFIGURATION SECTION.
002200  SOURCE COMPUTER.       COMPUTER NAME.
002300  OBJECT COMPUTER.       COMPUTER NAME.
002400  SPECIAL NAMES.         HARDWARE NAME.
003000  INPUT-OUTPUT SECTION.
003100  FILE CONTROL.  SELECT FILE-NAME 1
003200      SELECT FILE-NAME 2  SELECT ......
003300  I-O CONTROL.  APPLY ....
                   ............


004000  DATA DIVISION.
004100  RD MASTER-PAYROLL, LABEL RECORDS ARE
004200      STANDARD,DATA RECORDS ARE MASTER-
004300      PAY, SEQUENCED ON BADGE-NUMBER.
004400          01 MASTER-PAY SIZE IS 180 CHAR-
004500              ACTERS. CLASS IS ALPHAMERIC.
004600              02 BADGE-NUMBER SIZE IS 12
004700                  CHARACTERS, PICTURE IS
004800                  AAAXXX999999.
                        ..........


006000  PROCEDURE DIVISION.
006100  COMPUTATIONS SECTION.
006200  UPDATE-MASTER.  MOVE ADJUSTED-PAY TO
006300      NET-PAY.  ADD GROSS-PAY TO GROSS-
006400      YEAR-TO-DATE.  WRITE UPDATED-
006500      MASTER-PAY.  READ MASTER-PAYROLL
006600      RECORD.
                   ..........
```


Use and Context
U1 Social Organization and Work
U3 Human-Machine Fit and Adaptation
U2 Application Areas
Human
H1 Human Information Processing
H2 Language, Communication and Interaction
H3 Ergonomics
Computer
C2 Dialogue Techniques
C4 Computer Graphics
C3 Dialogue Genre
C5 Dialogue Architecture
C1 Input and Output Devices
D4 Example Systems and Case Studies
D3 Evaluation Techniques
D2 Implementation Techniques and Tools
D1 Design Approaches
Development Process

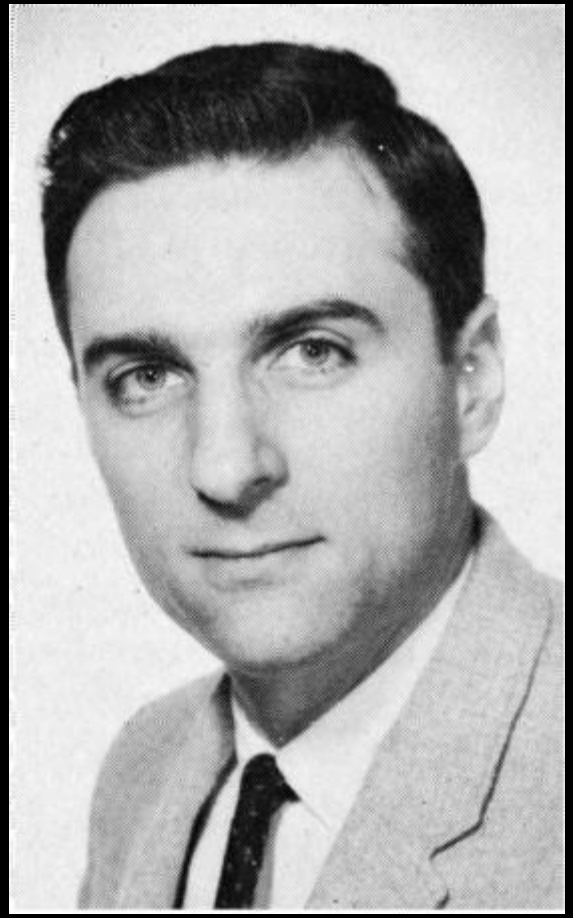
Now there is source code…
*The machine is the **code**.*



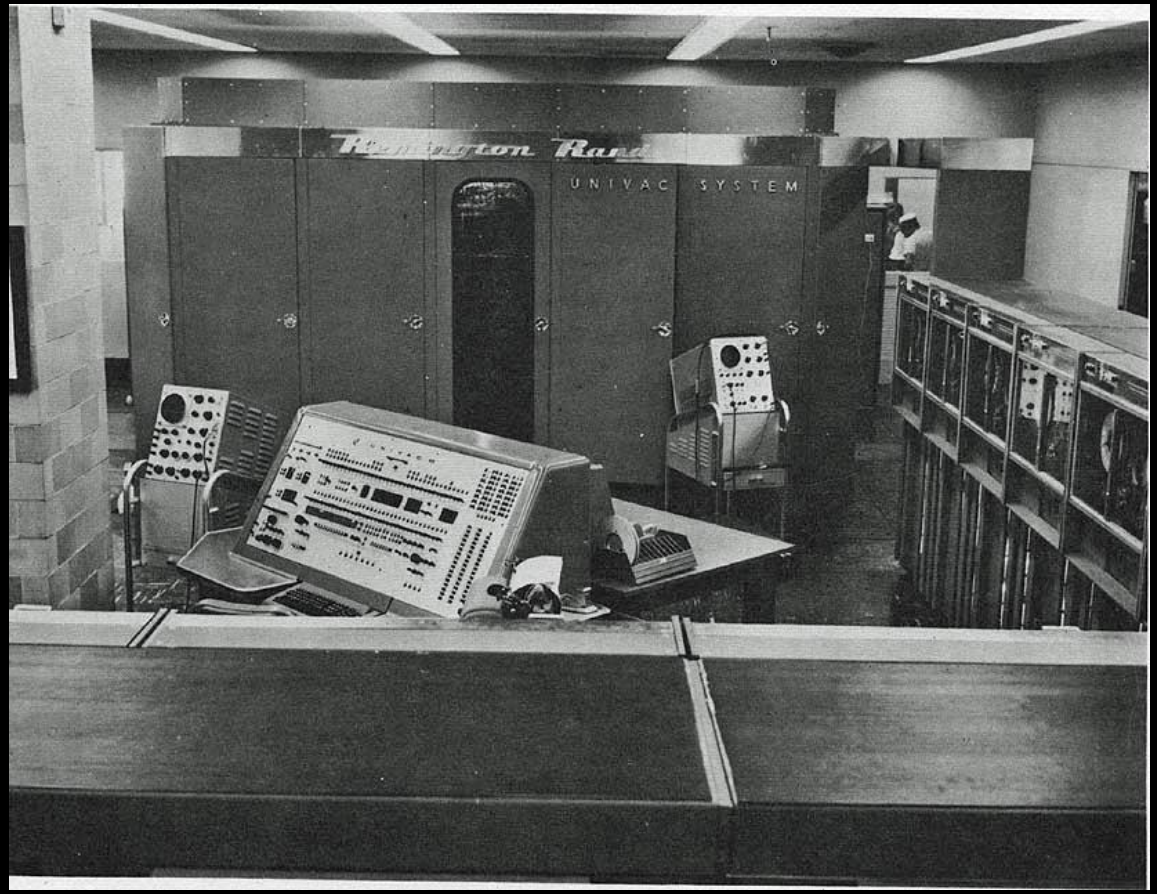
TYPICAL COBOL.

```
000100  IDENTIFICATION DIVISION.
000200  PROGRAM ID.  PAYROLL.
000300  AUTHOR.  JOHN DOE.
000400  DATE.  APRIL 5TH 1960.
001100  REMARKS.
001101      INPUT FROM RUN 4 AND OUTPUT TO RUN 25.
            THIS PROGRAM PROCESSES SALARIED
            EMPLOYEES ONLY.

002000  ENVIRONMENT DIVISION.
002100  CONFIGURATION SECTION.
002200  SOURCE-COMPUTER.      COMPUTER NAME.
002300  OBJECT-COMPUTER.      COMPUTER NAME.
002400  SPECIAL NAMES.        HARDWARE NAME.
002900  INPUT-OUTPUT SECTION.
003100  FILE CONTROL.  SELECT FILE-NAME 1
003200      SELECT FILE-NAME 2  SELECT ******
003300  I-O CONTROL.  APPLY ****.
                  ***********

004000  DATA DIVISION.
004100  FD MASTER-PAYROLL, LABEL RECORDS ARE
004200      STANDARD, DATA RECORDS ARE MASTER-
004300      PAY, SEQUENCED ON BADGE-NUMBER.
004400      01 MASTER-PAY. SIZE IS 100 CHAR-
004500          ACTERS. CLASS IS ALPHAMERIC.
004600          05 BADGE-NUMBER SIZE IS 12
004700              CHARACTERS. PICTURE IS
004800              AAAXXX999999.
                  ***********

006000  PROCEDURE DIVISION.
006100  COMPUTATIONS SECTION.
006200  UPDATE-MASTER.  MOVE ADJUSTED-PAY TO
006300      NET-PAY.  ADD GROSS-PAY TO GROSS-
006400      YEAR-TO-DATE.  WRITE UPDATED-
006500      MASTER-PAY.  READ MASTER-PAYROLL
006600      RECORD.
                  ***********
```

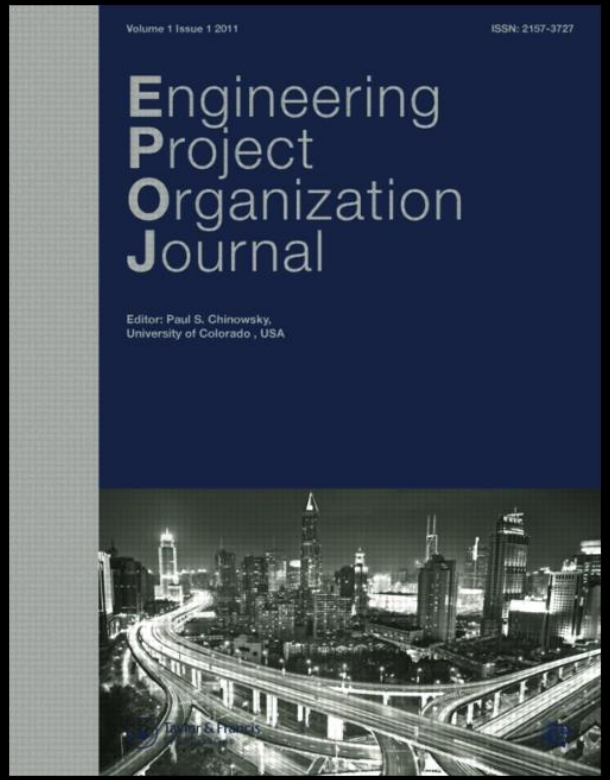
Communicating



Melvin Conway


Remington Rand
UNIVAC SYSTEM

Project-Based Organizations



“Project-based organizations revolve around the concept that a group of individuals or firms join together with the explicit purpose of producing a tangible set of outputs”

-- Paul Chinowsky, EPOJ 2011

“How Do Committees Invent?”

Harvard Business Review


REJECTED

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:

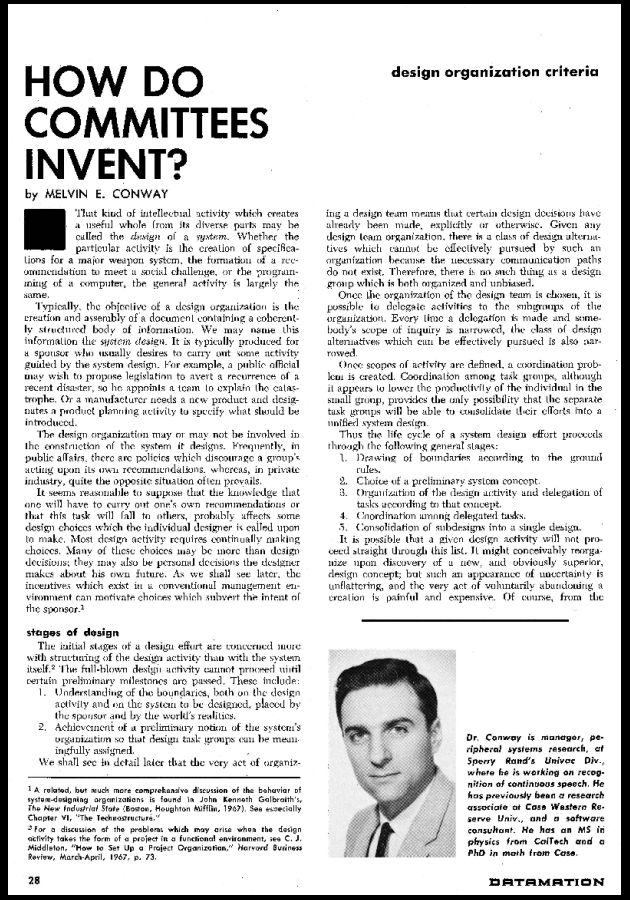
1. Drawing of boundaries according to the ground rules.
2. Choice of a preliminary system concept.
3. Organization of the design activity and delegation of tasks according to that concept.
4. Coordination among delegated tasks.
5. Consolidation of subdesigns into a single design.

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

[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.

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.



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.

“Any organization that designs a system (defined more broadly here than just information systems) will inevitably produce a design whose structure is a copy of the organization's communication structure.”

-- Mel Conway, 1967

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.

A system's design is a copy of the organization's communication structure.

-- Mel Conway, 1967

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.

Communication dictates design.

-- Mel Conway, 1967

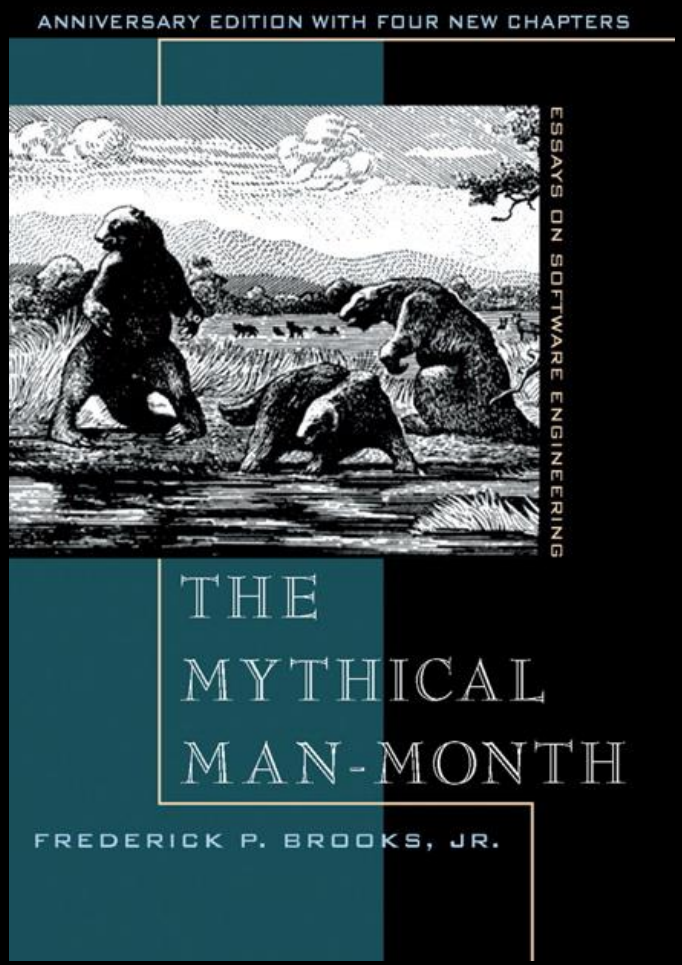
Conway's Law

ANNIVERSARY EDITION WITH FOUR NEW CHAPTERS

ESSAYS ON SOFTWARE ENGINEERING

THE MYTHICAL MAN-MONTH

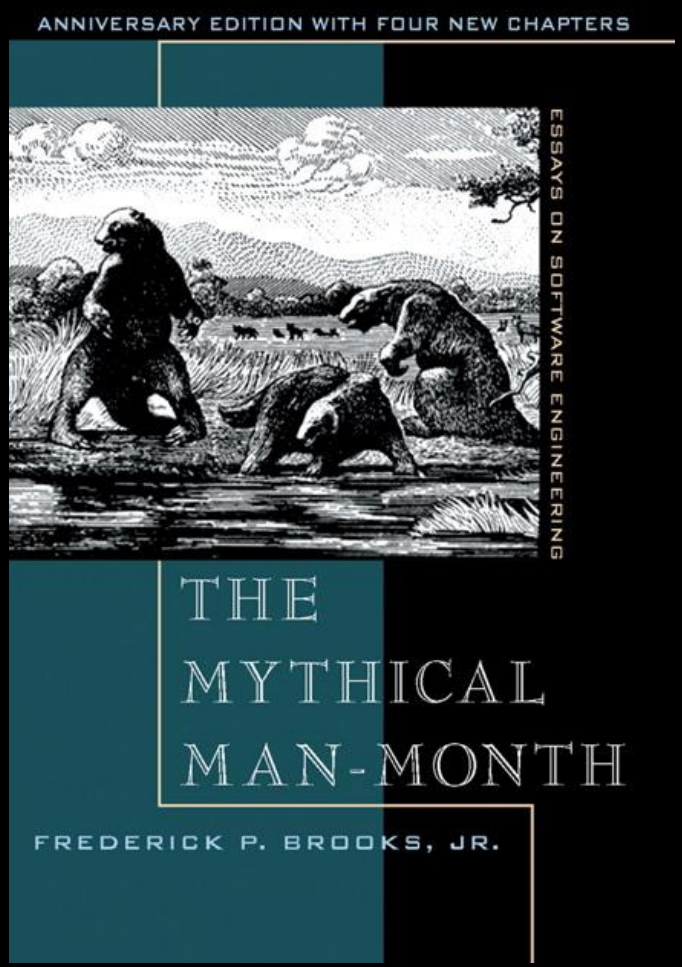
FREDERICK P. BROOKS, JR.



Brooks' Law

"*Adding manpower to a late software project makes it later*."

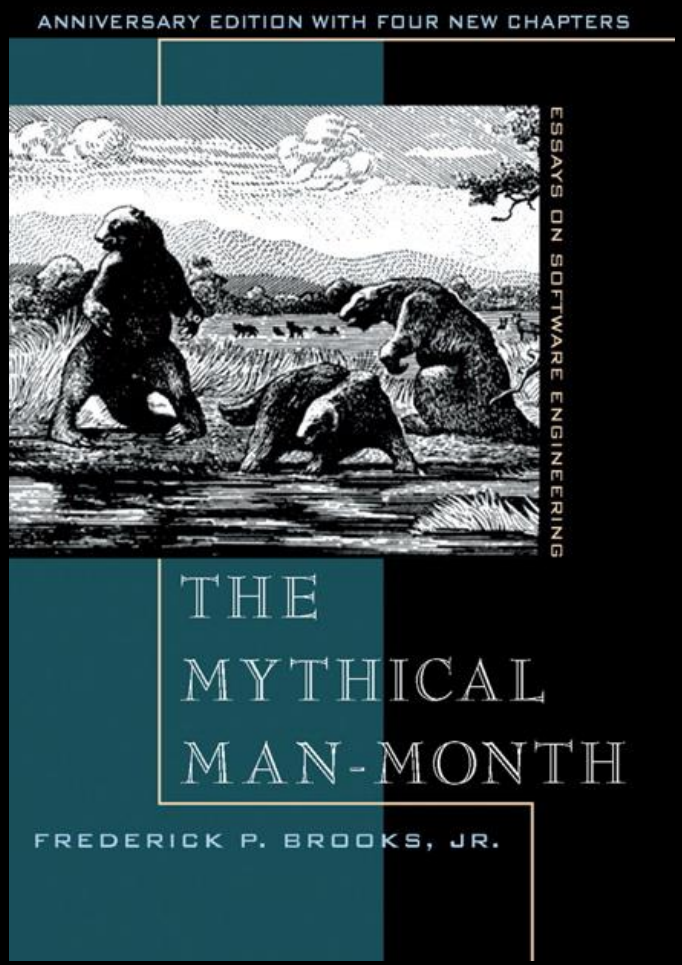
-- Fred Brooks, 1975



Intercommunication formula

$$n(n - 1) / 2$$

-- Fred Brooks, 1975



Intercommunication formula

```
5*(5–1)/2 = 10
15*(15–1)/2 = 105
50*(50–1)/2 = 1,225
150*(150–1)/2 = 11,175
```

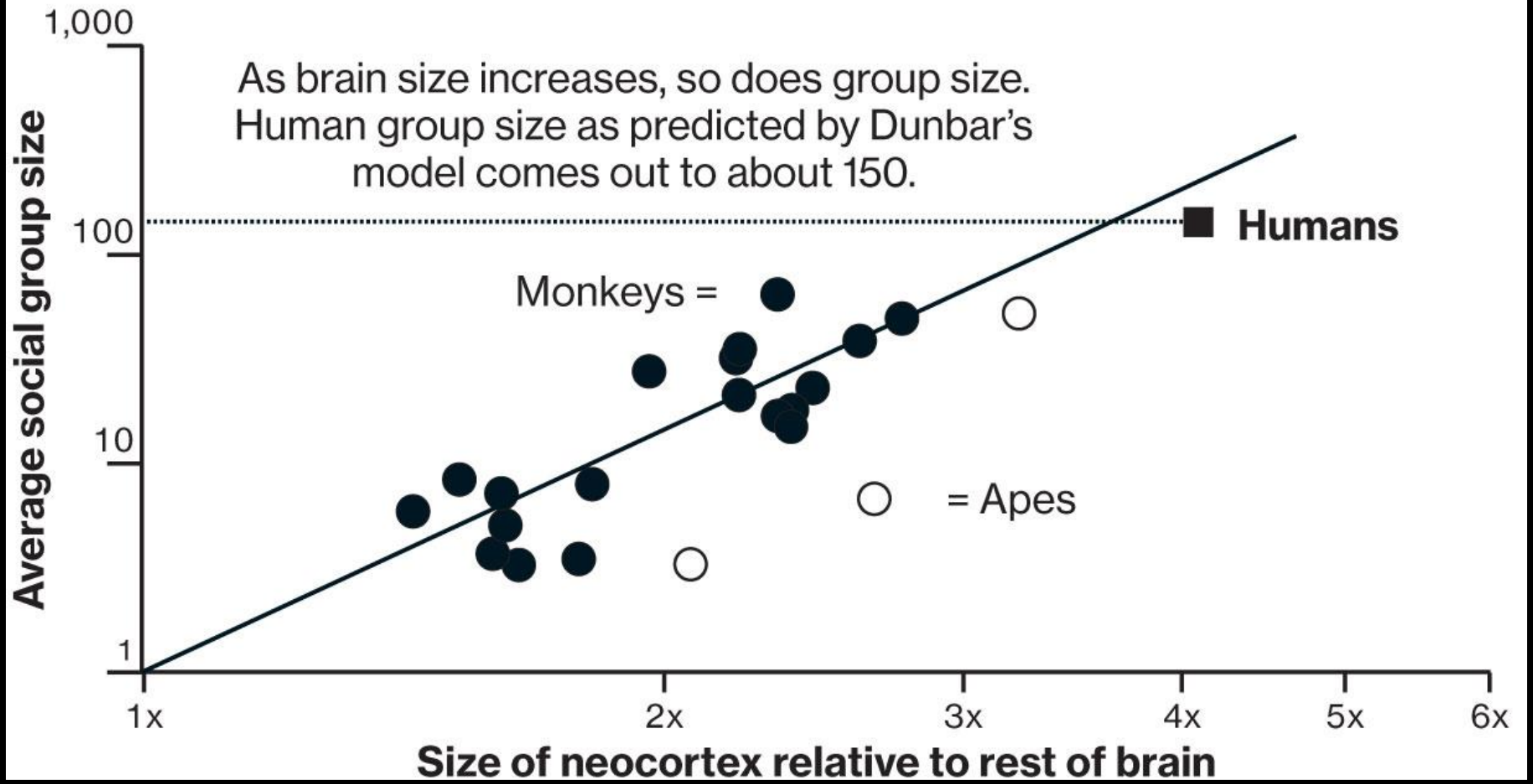
-- Fred Brooks, 1975

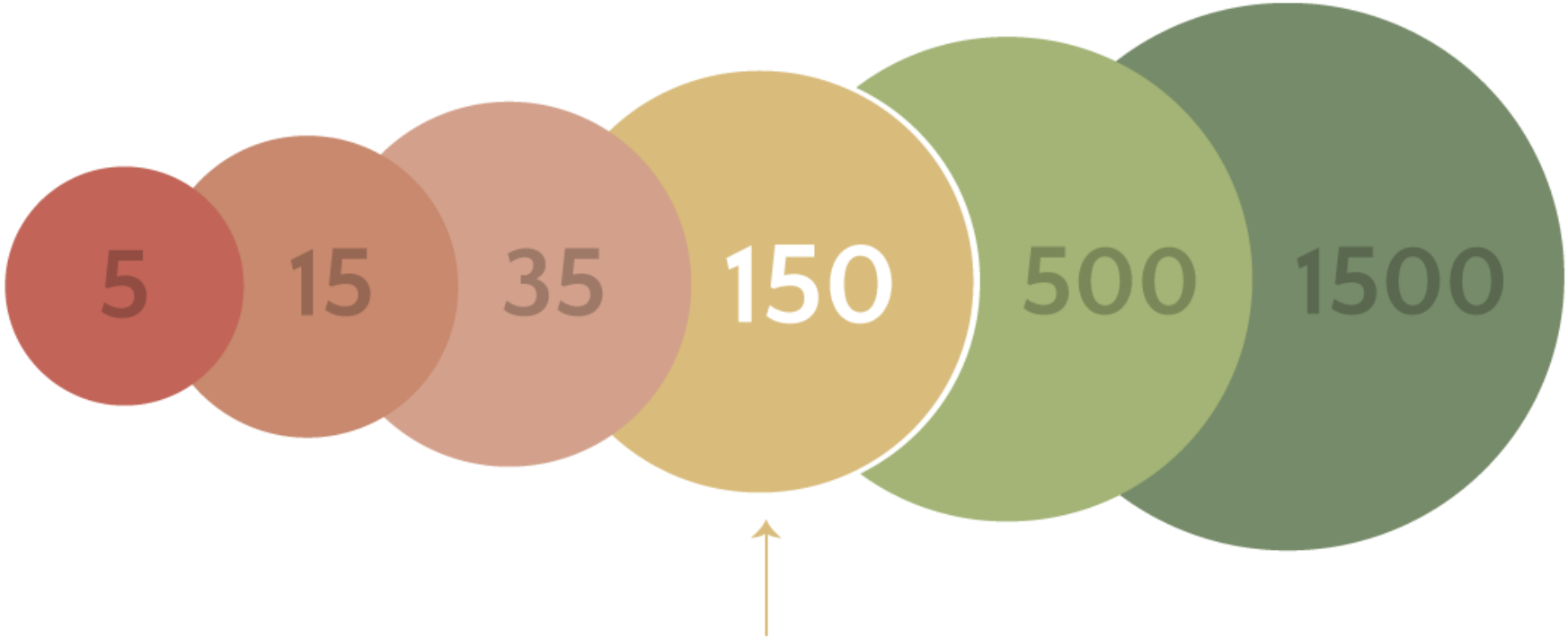
Dunbar's Number

A measurement of the "cognitive limit to the number of individuals with whom any one person can maintain stable relationships."

-- Robin Dunbar, 1992

The Social Cortex





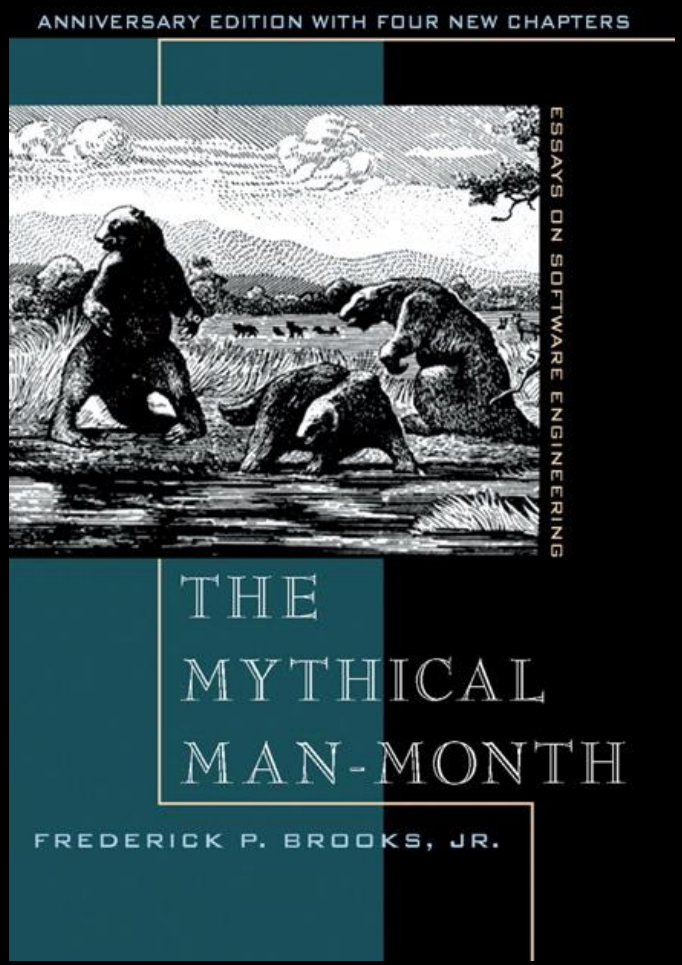
Dunbar's Number

the max number of relationships a person can maintain

Dunbar Groups

Intimate friends: 5
Trusted friends: 15
Close friends: 35
Casual friends: 150

-- Robin Dunbar, 1992



Intercommunication formula

```
5*(5–1)/2 = 10
15*(15–1)/2 = 105
50*(50–1)/2 = 1,225
150*(150–1)/2 = 11,175
```

-- Fred Brooks, 1975

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.

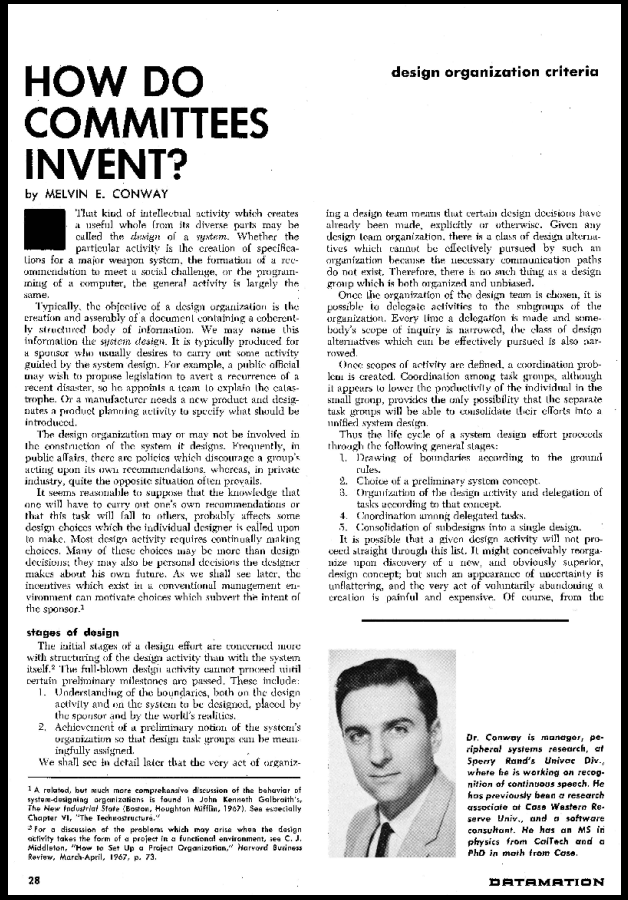
Communication dictates design.

-- Mel Conway, 1967

Conway's (first) Law

So… what about other Conway Laws?

Conway's Second 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.

Doing it Over

"There is never enough time to do something right, but there is always enough time to do it over."

-- Mel Conway, 1967

Trade Offs

Efficiency-Effectiveness Trade Offs (ETTOs)

The ETTO Principle:

Efficiency-Thoroughness Trade-Off

Why Things That Go Right Sometimes Go Wrong

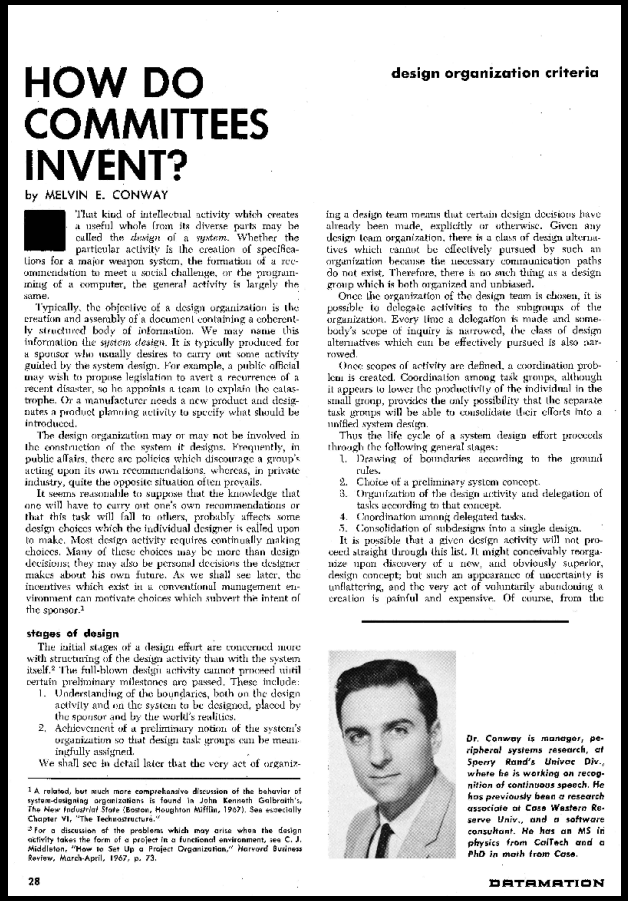
ERIK HOLLNAGEL

Conway's Third Law

Homomorphism

"There is a homomorphism from the linear graph of a system to the linear graph of its design organization"

-- Mel Conway, 1967



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."

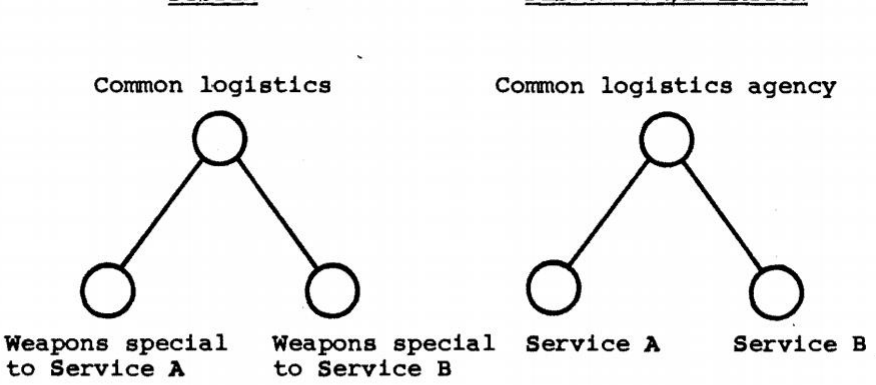
[2] For a discussion of the problems which may arise when the design activity takes the form of a project in a functional environment, see C. J. Middleton, "How to Set Up a Project Organization," *Harvard Business Review*, March-April, 1967, p. 73.

ho·mo·mor·phism

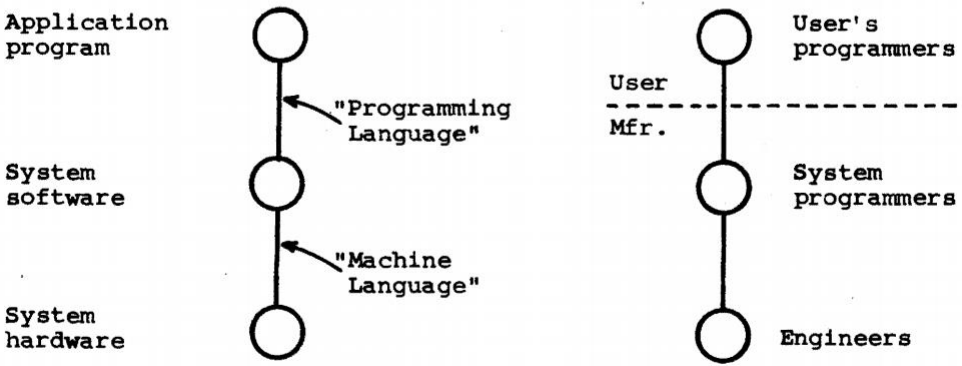
/ˌhōməˈmôrfizəm/

noun MATHEMATICS

a transformation of one set into another that preserves in the second set the relations between elements of the first.

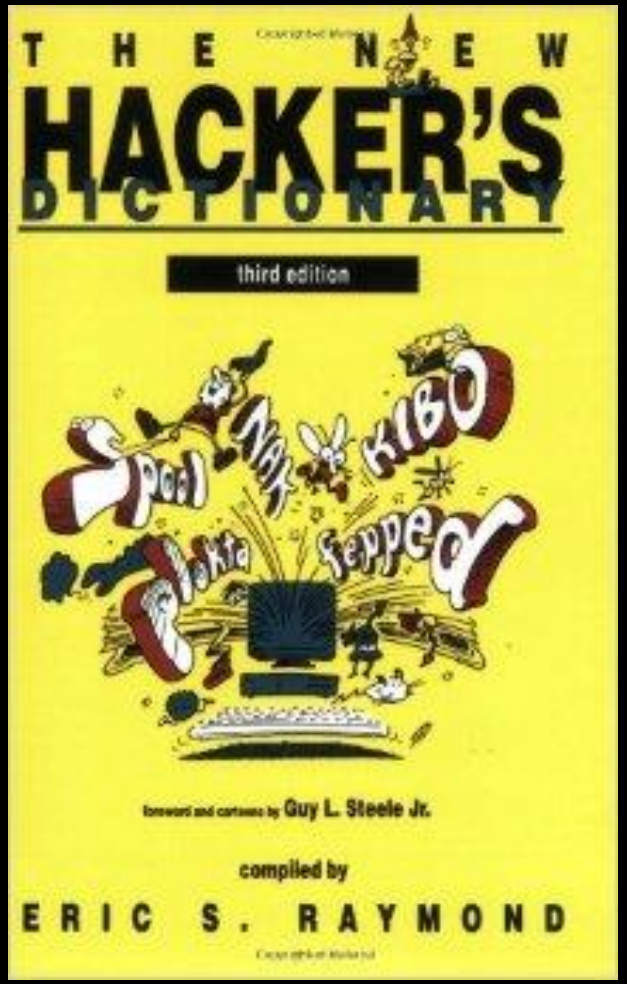


3a. A Weapon System



3b. A Computer System

Figure 3 Two examples of identity of structure between a system and its design organization.

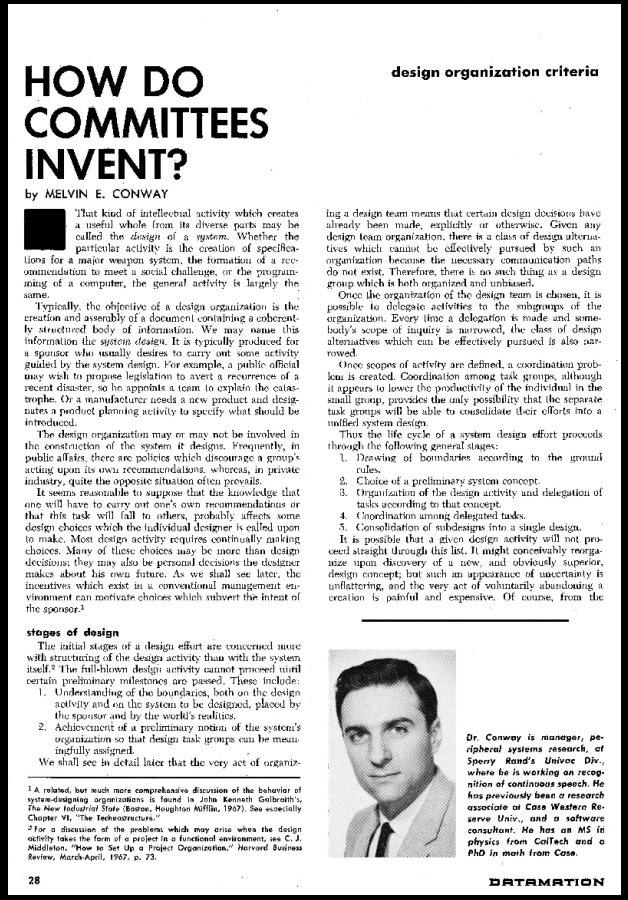


Homomorphism

“If you have four groups working on a compiler, you'll get a 4-pass compiler.”

- Eric S. Raymond, 1991

Conway’s Fourth 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.

Disintegration

"The structures of large systems tend to disintegrate during development, qualitatively more so than with small systems."

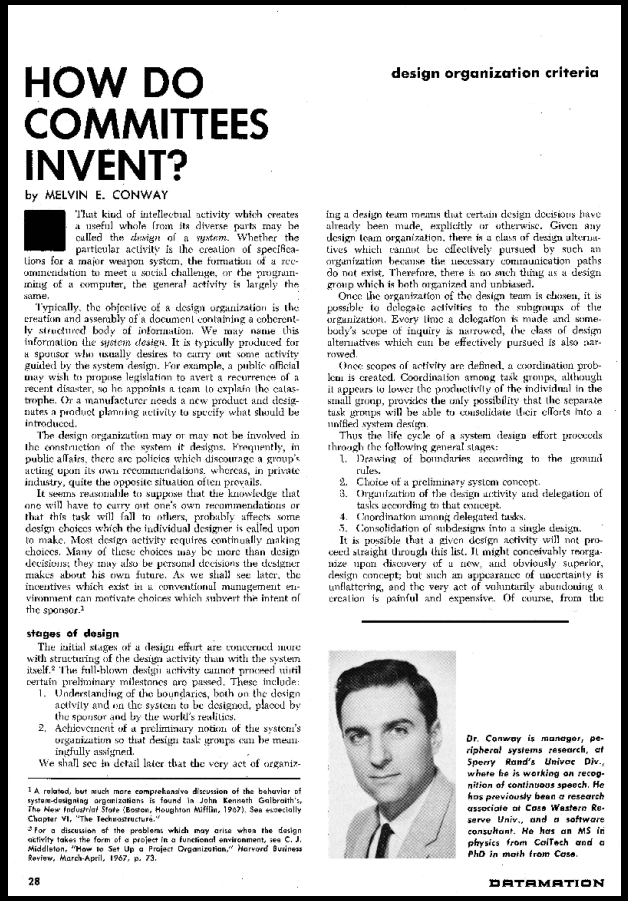
-- Mel Conway, 1967

Three reasons Disintegration occurs…

Disintegration: Reason #1

“The realization that the system will be large, together with organization pressures, make irresistible the temptation to assign too many people to a design effort”

-- Mel Conway, 1967



design organization criteria

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:

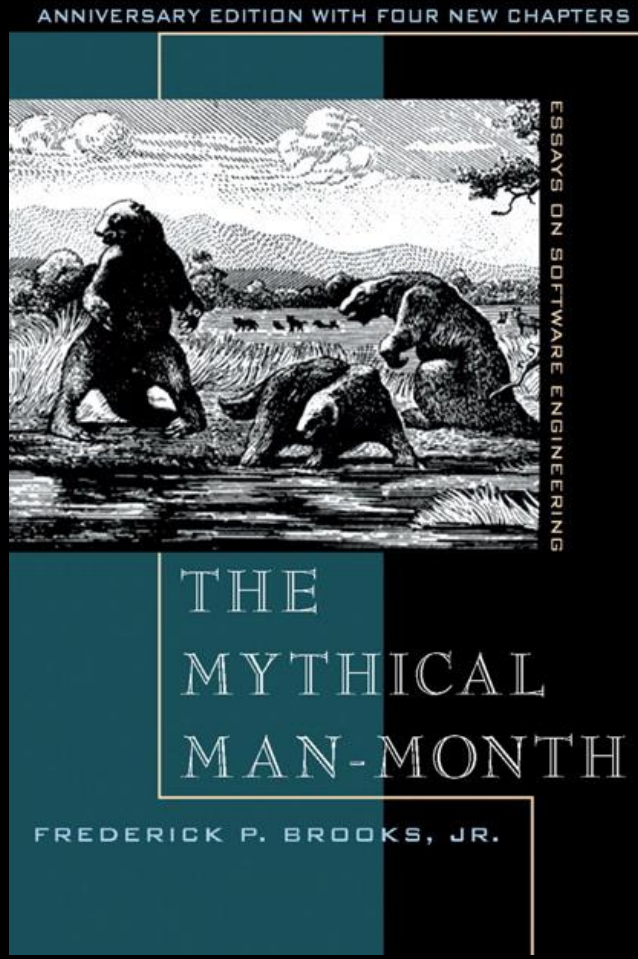
1. Drawing of boundaries according to the ground rules.
2. Choice of a preliminary system concept.
3. Organization of the design activity and delegation of tasks according to that concept.
4. Coordination among delegated tasks.
5. Consolidation of subdesigns into a single design.

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.

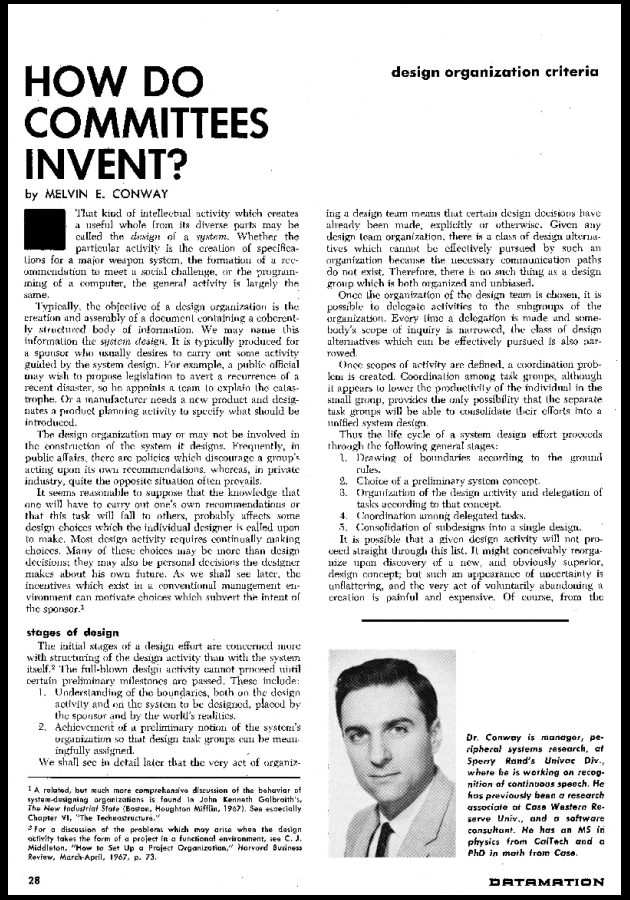


Brooks' Law

Adding manpower to a late software project makes it later.

-- Fred Brooks, 1975

Disintegration: Reason #2



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.

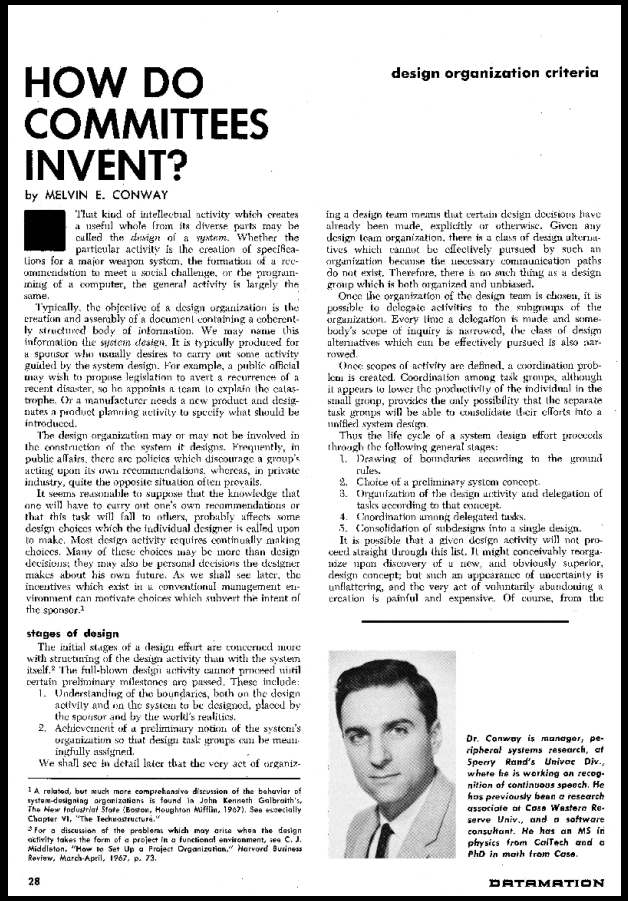
"Application of the conventional wisdom of management to a large design organization causes its communication structure to disintegrate."

-- Mel Conway, 1967

Dunbar's Number

A measurement of the "cognitive limit to the number of individuals with whom any one person can maintain stable relationships."

-- Robin Dunbar, 1992



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.

Disintegration: Reason #3

"Homomorphism insures that the structure of the system will reflect the disintegration which has occurred in the design organization."

-- Mel Conway, 1967

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.

Communication dictates design.

-- Mel Conway, 1967

*The machine is the **organization**.*

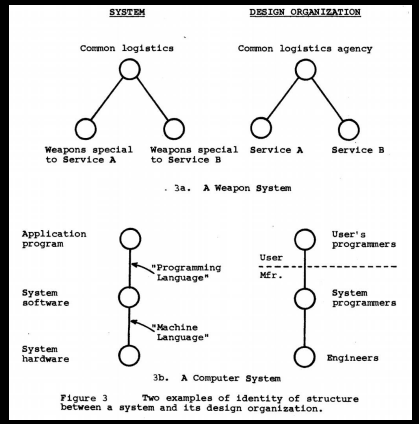


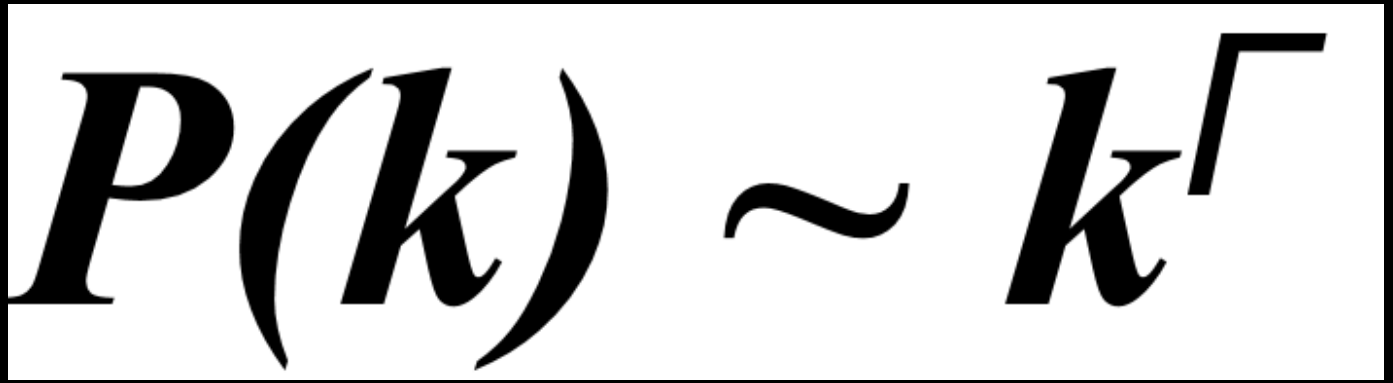
Figure 3 Two examples of identity of structure between a system and its design organization.

Scaling

“Free” as in “Scale-Free”

*“A **scale-free network** is a network whose degree distribution follows a power law."*

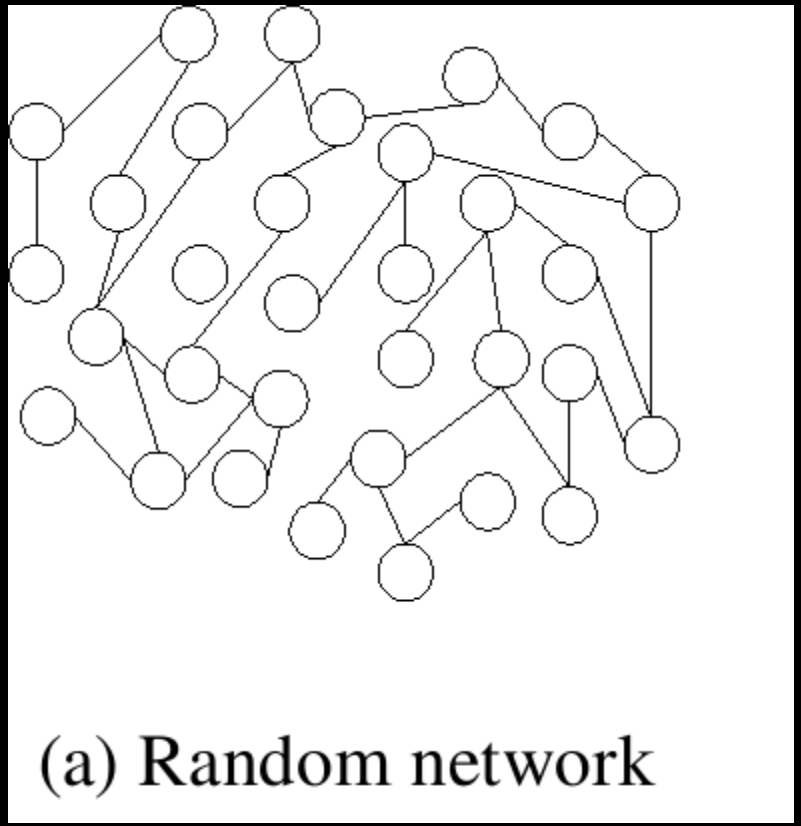
*“A **scale-free network** is a network whose degree distribution follows a **power law**."*


P(k) ~ k^Γ

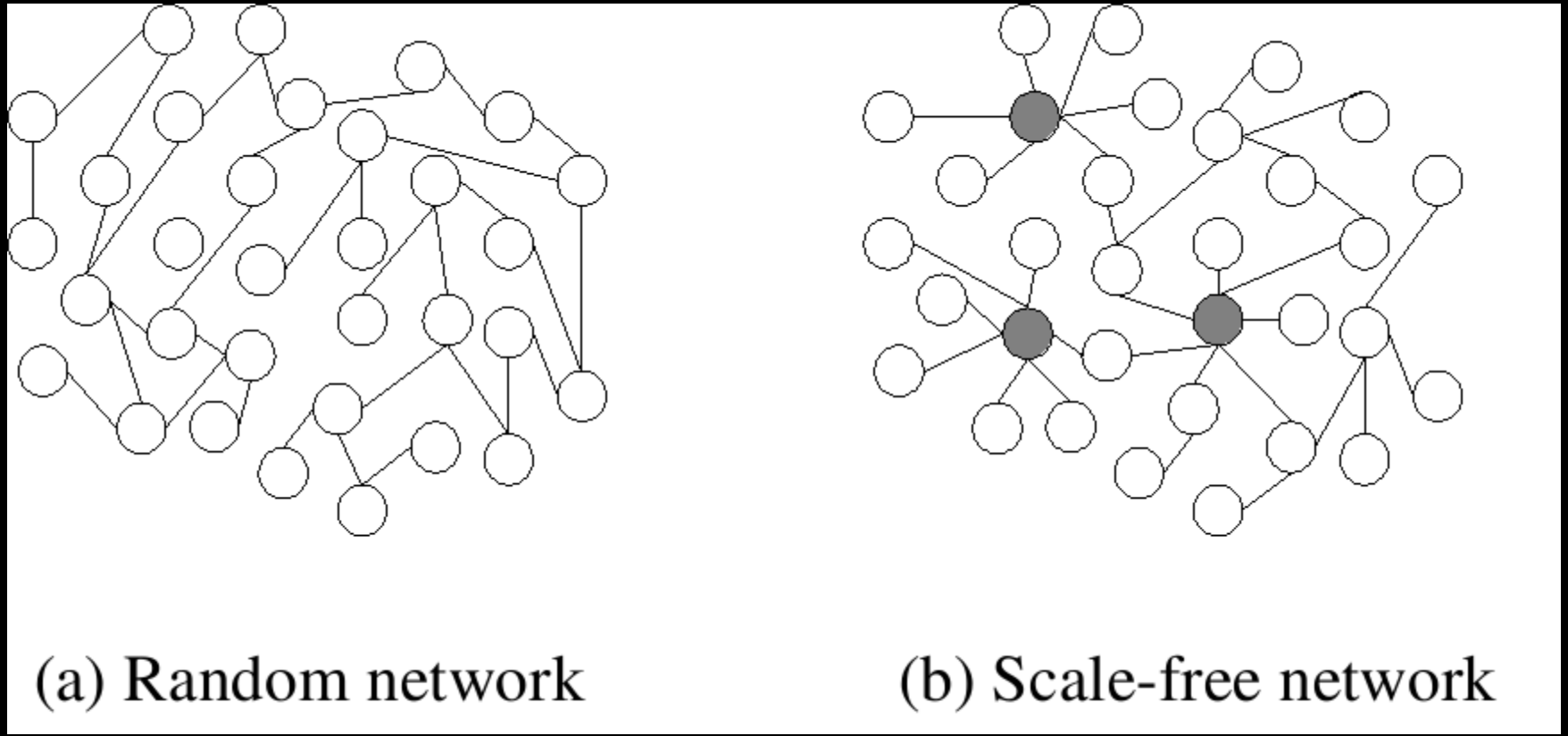

ABA
GPA1
AGB1
target genes

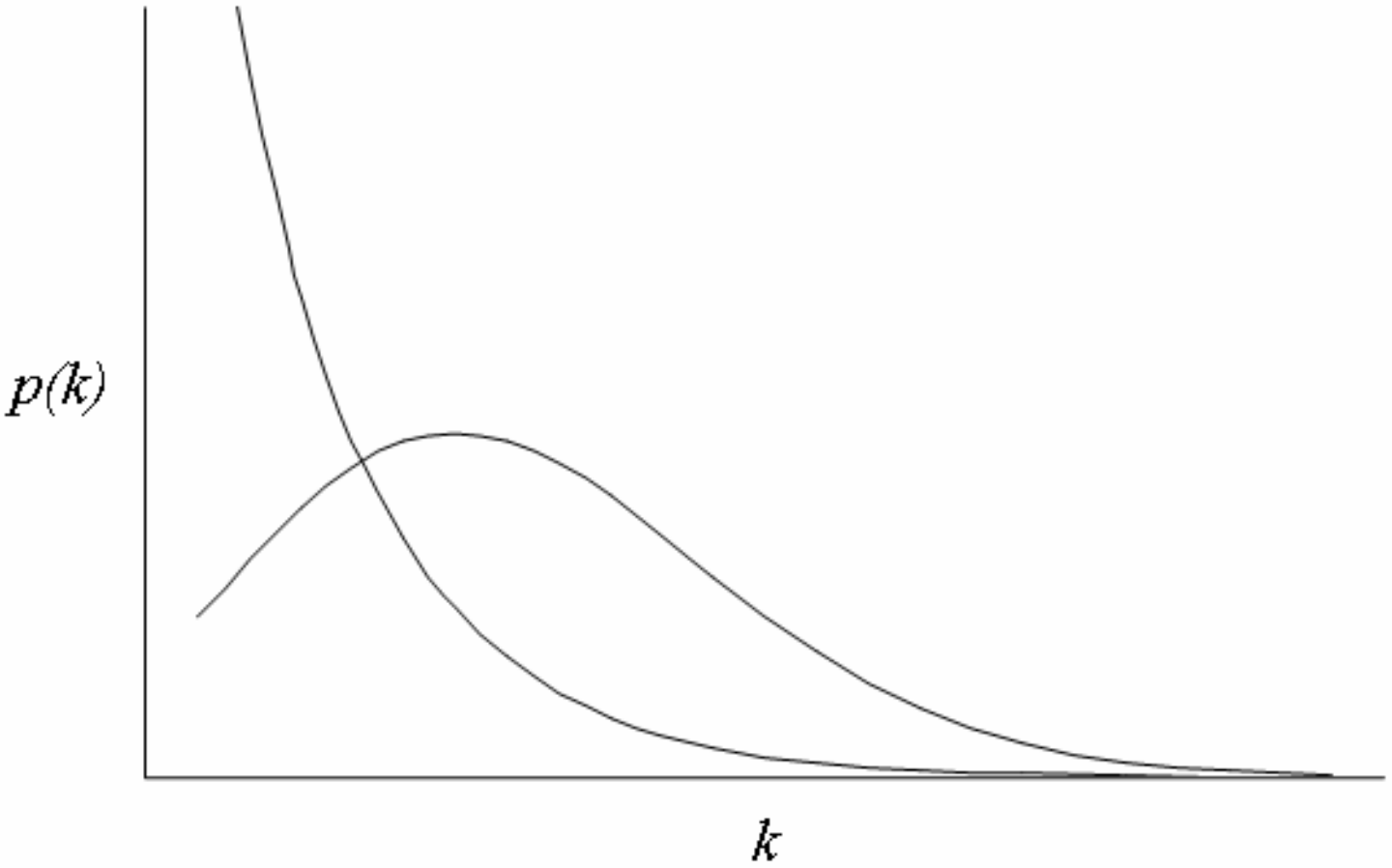

LAMONT
LIBRARY

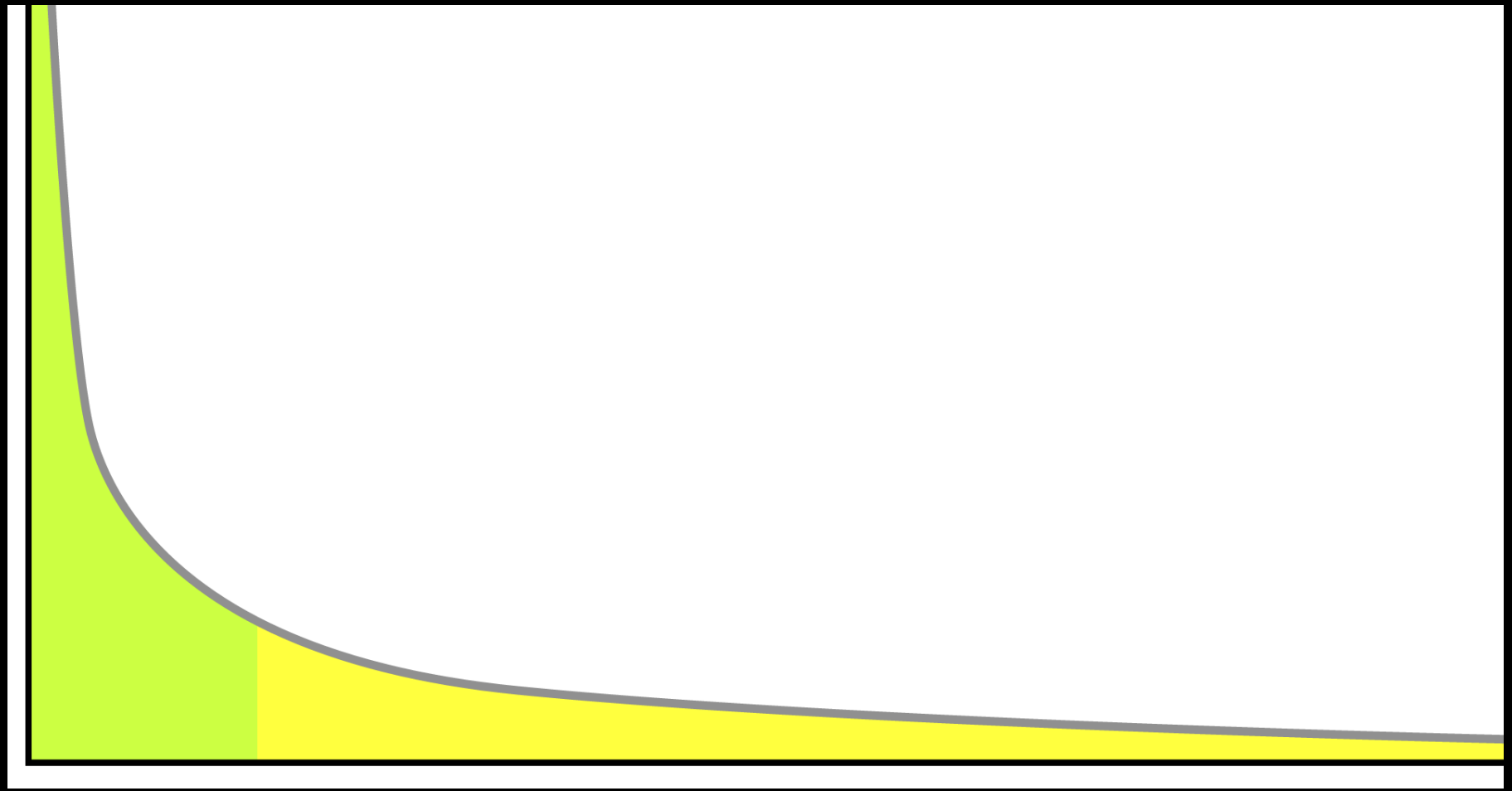

ABA
GPA I
AGB I
target games



(a) Random network


(a) Random network
(b) Scale-free network


p(k)
k

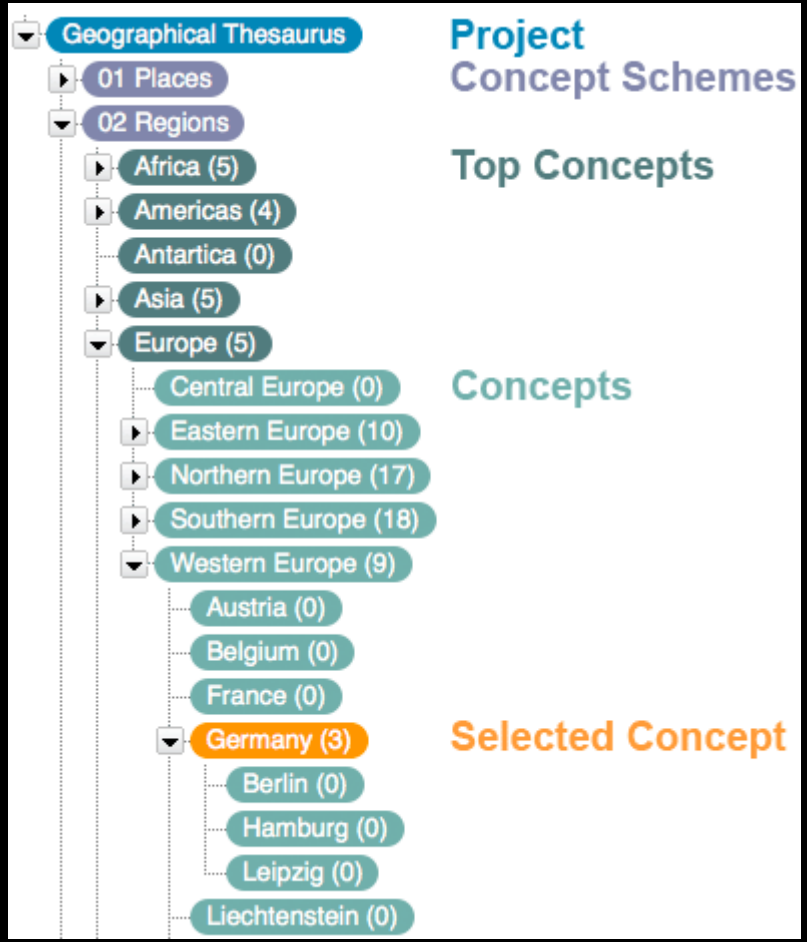


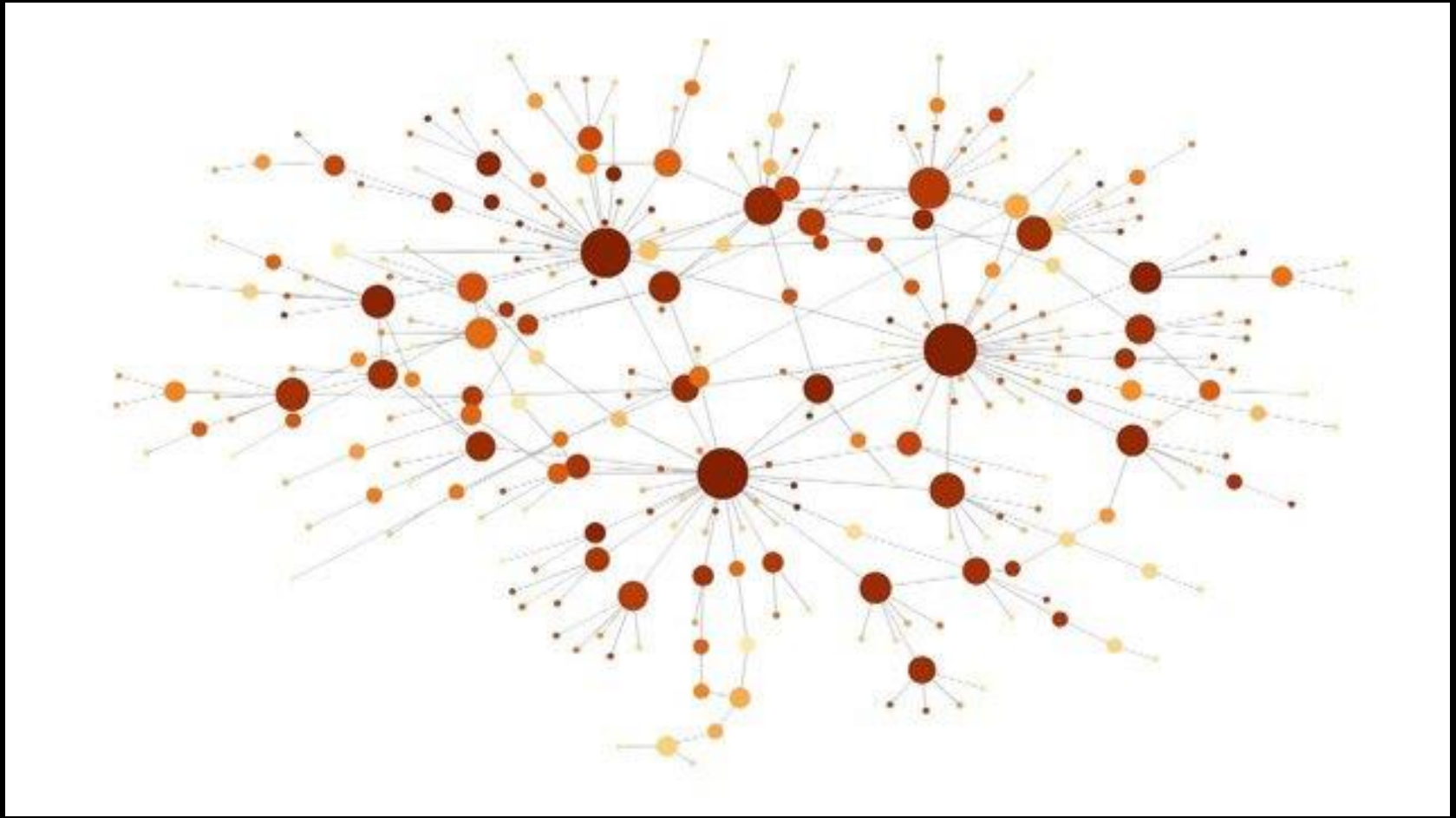
Emergence of Scaling in Random Networks

Albert-László Barabási* and Réka Albert

Department of Physics, University of Notre-Dame, Notre-Dame, IN 46556

Systems as diverse as genetic networks or the world wide web are best described as networks with complex topology. A common property of many large networks is that the vertex connectivities follow a scale-free power-law distribution. This feature is found to be a consequence of the two generic mechanisms that networks expand continuously by the addition of new vertices, and new vertices attach preferentially to already well connected


Geographical Thesaurus
01 Places
02 Regions
Africa (5)
Americas (4)
Antartica (0)
Asia (5)
Europe (5)
Central Europe (0)
Eastern Europe (10)
Northern Europe (17)
Southern Europe (18)
Western Europe (9)
Austria (0)
Belgium (0)
France (0)
Germany (3)
Berlin (0)
Hamburg (0)
Leipzig (0)
Liechtenstein (0)
Project
Concept Schemes
Top Concepts
Concepts
Selected Concept



This applies to code, too…

Basic Class Hierarchy ("*" indicates an abstract class)

SWT:

```
Object
    *Dialog
         ColorDialog
         DirectoryDialog
         FileDialog
         FontDialog
         MessageDialog
         PrintDialog
    *Widget
        Menu
        *Item
            CoolItem
            CTabItem
            MenuItem
            TabItem
            TableColumn
            TableItem
            TableTreeItem
            ToolItem
            TrayItem
            TreeColumn
            TreeItem
        *Control
            Button
            Label
            ProgressBar
            Sash
            Scale
            Scrollable
                Composite
                    Browser
                    Canvas
                        AnimatedProg
                        CLabel
                        Decorations
                            Shell
                        FormText
                        StyledText
                        TableCursor
                    CBanner
                    CCombo
                    Combo
                    CoolBar
                    CTabFolder
                    Group
                    ProgressIndicato
                    SashForm
                    ScrolledComposit
                    TabFolder
                    Table
                    TableTree
                    ToolBar
                    Tray
                    Tree
                    ViewForm
                List
                Text
            Slider
```

Swing:

```
Object
    *Component
        Container
            *JComponent
                *AbstractButton
                    JButton
                    JMenuItem
                        JCheckBonMenuItem
                        JMenu
                        JRadioButonMenuItem
                    *JToggleButton
                        JCheckBox
                        JRadioButton
                Box
                Filler
                JColorChooser
                JComboBox
                JDesktopIcon
                JFileChooser
                JInternalFrame
                JLabel
                JLayeredPane
                    JDesktopPane
                JList
                JMenuBar
                JOptionPane
                JPanel
                JPopupMenu
                JProgressBar
                JRootPane
                JScrollBar
                JScrollPane
                JSeparator
                JSlider
                JSplitPane
                JTabbedPane
                JTable
                JTableHeader
                *JTextComponent
                    JEditorPane
                        FrameEditorPane
                        JTextPane
                    JTextArea
                    Jtextfield
                        JPasswordField
                JToolBar
                JToolTip
                JTree
                JViewport
                    ScrollableTabViewport
            Panel
                Applet
                    JApplet
            Window
                Dialog
                    JDialog
                Frame
                    JFrame
                JWindow
```

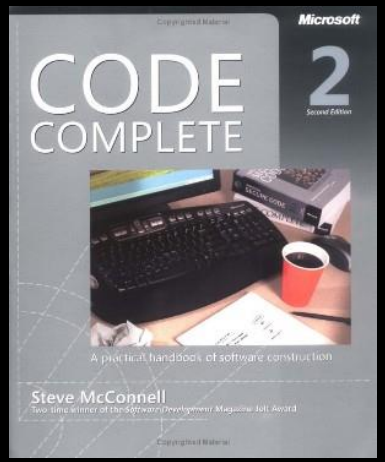
Code as hierarchy


BUG

The more code, the more bugs.

“The industry average is about 15 - 50 errors per 1000 lines of delivered code.“

- Steve McConnell




Jazero
IDYLKY Z PEKLA
NEBOJTE SE LÁSKY
ION ARIEŠANU
ZLOŽITÝ POCIT ŠŤASTIA
DĚJINY EVROPSKÉ CIVILIZACE
Finanční příručka
Rebecca Gablé
Poslední allegretto

Code is not the solution, code is the problem.


BAD
BAD

the

next

the

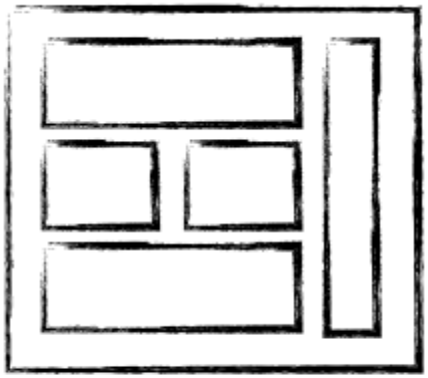

the
next
big


thing
next
the
big


thing
next
is
the
big

small

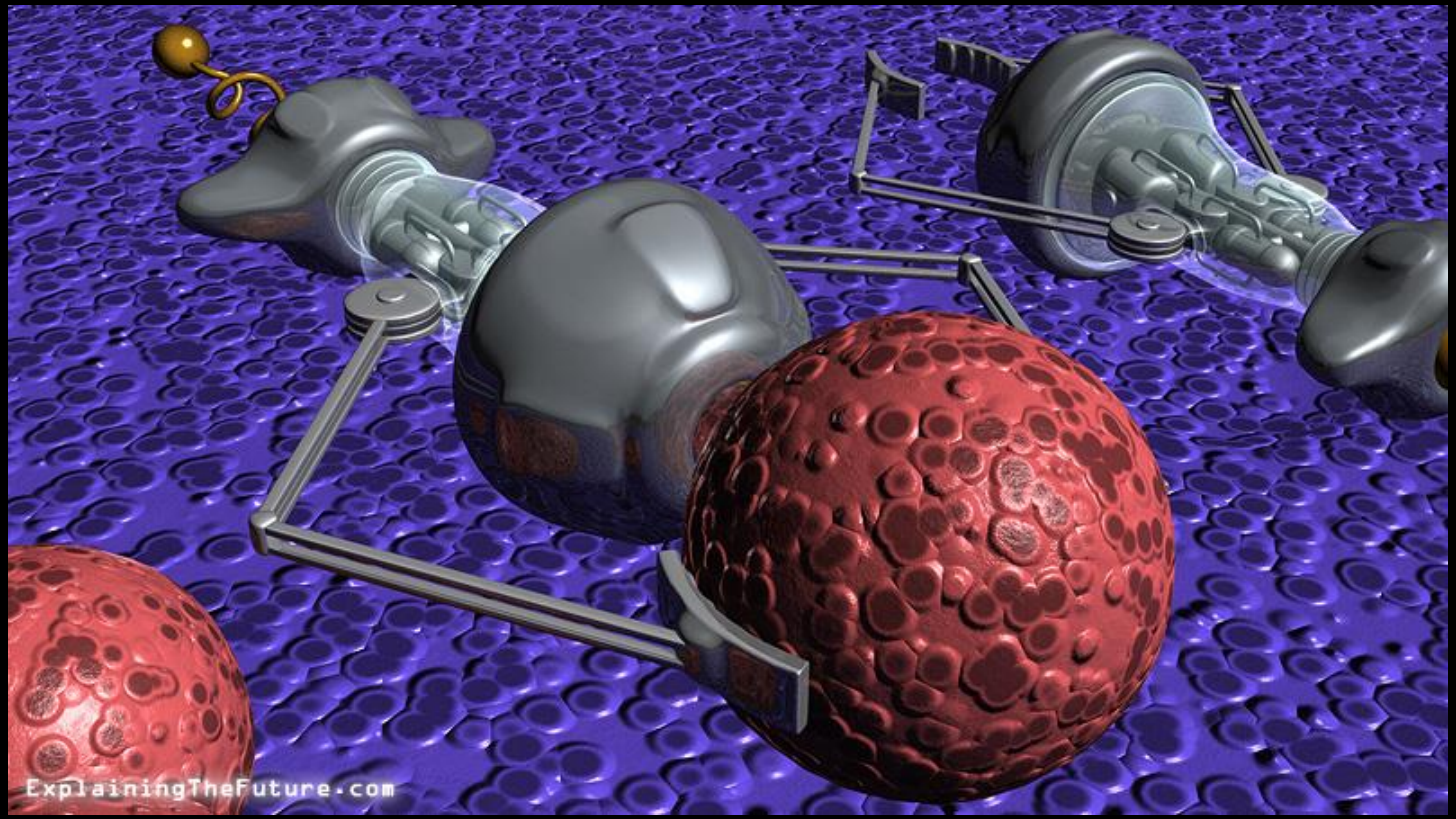

thing
next
is
the
big
small



MONOLITHIC/LAYERED

MICRO SERVICES


ESC
TAB
CTRL
SHIFT
P1
P2
ENTER


ExplainingTheFuture.com

NO

NO CODE

Artisinal Luddites

THE LEADER or the LUDDITES

Ned Ludd, 1811

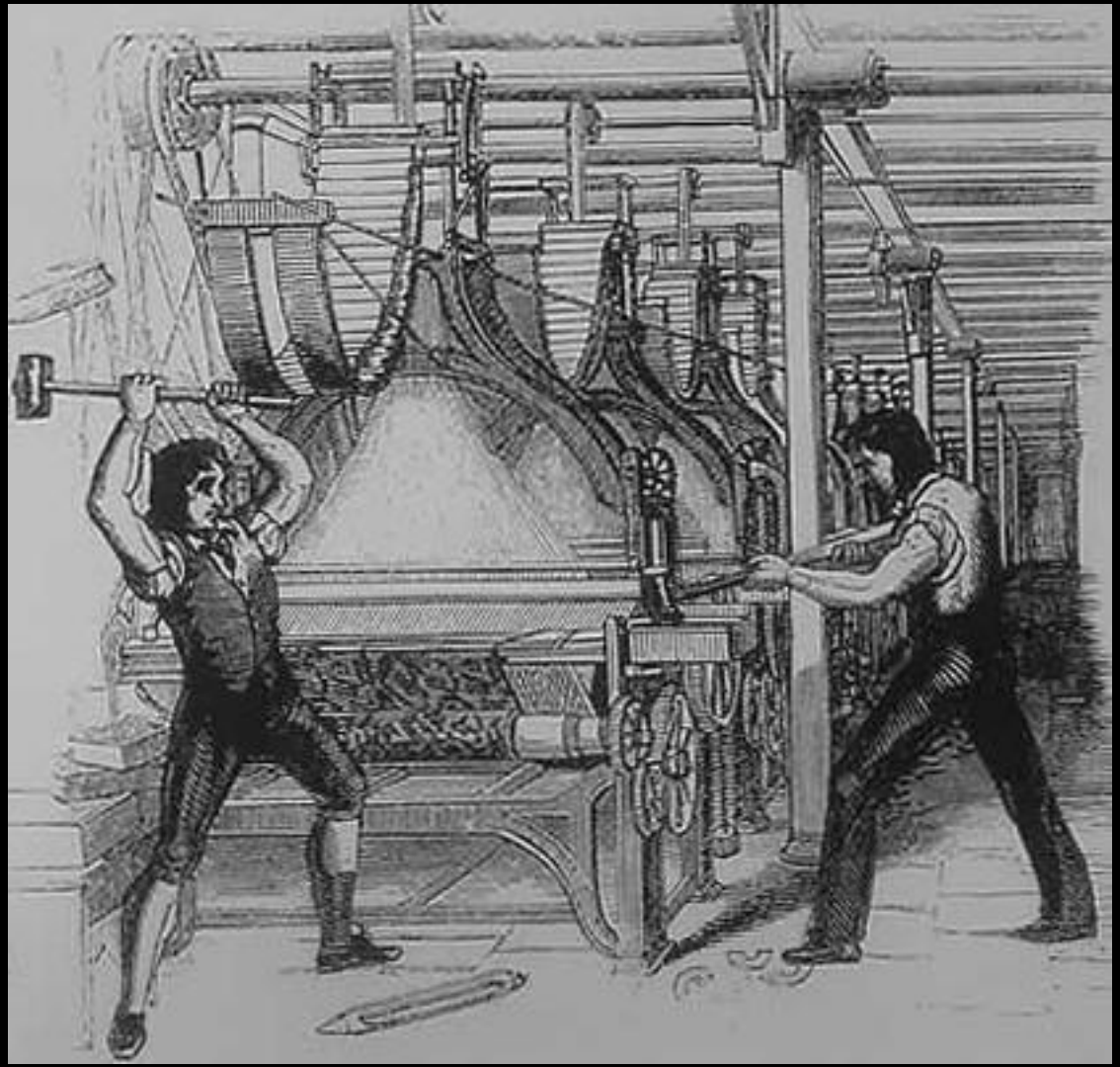
*The **Luddites** were 19th-century English textile workers who protested against newly developed labor-economizing technologies*

- Wikipedia

Ned Ludd, 1811

*The **Luddites** were 19th-century English textile workers who protested against newly developed labor-economizing technologies*

- Wikipedia



ar·ti·san·al

/ärˈtēzən(ə)l/

adjective

relating to or characteristic of an artisan.
"artisanal skills"

ar·ti·san·al

/ärˈtēzən(ə)l/

adjective

relating to or characteristic of an artisan.
"artisanal skills"

- (of a product, especially food or drink) made in a traditional or non-mechanized way.
  "artisanal cheeses"

Slow Programming

*The **slow programming** movement is a software development philosophy that emphasizes careful design, quality code, software testing and thinking.*

- Wikipedia

Slow Programming

*The **slow programming** movement is a software development philosophy that emphasizes careful design, quality code, software testing and thinking.*

- Wikipedia


SAS
SAS

“What works good is better than what looks good,
because what looks good can change,
but works good will still work.”

– Charles Eames

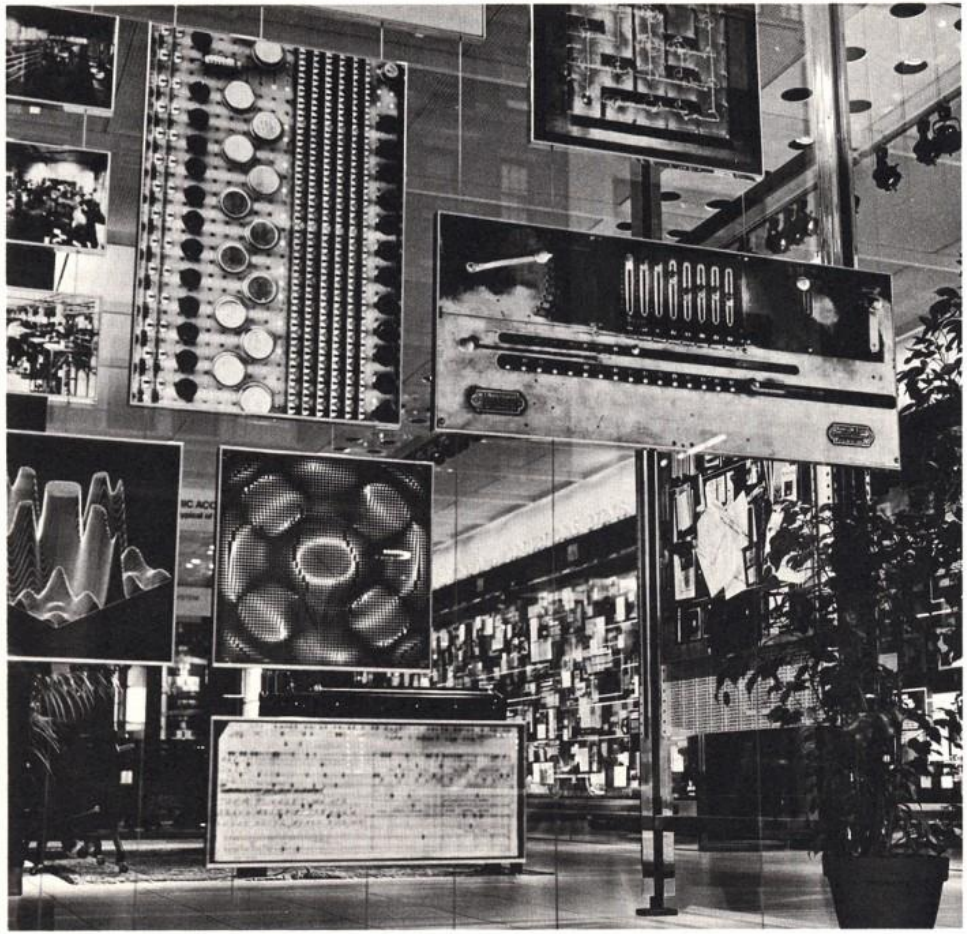
“What works good is better than what looks good,
because what looks good can change,
but works good will still work.”

– Charles Eames

“What works good is better than what looks good,
because what looks good can change,
but works good will still work.”

– Charles Eames


Get Lucky



Exhibition designed for IBM by the Office of Charles and Ray Eames

A COMPUTER PERSPECTIVE

A sequence of 20th century ideas, events, and artifacts from the history of the information machine

Charles & Ray Eames

“I think everybody in this country should learn how to program a computer because it teaches you how to think.”

- Steve Jobs

Life skills are not the same as livelihood.

"People may outlaw driving cars because it's too dangerous. You can't have a person driving a two-ton death machine.“

- Elon Musk


ONE NIGHT ONLY
SATURDAY AT THE ARENA:
THRILL to the CONTEST of
MAN VS. MACHINE!
CERTIFICATE OF SHARPENING
DAVID REES
★ ARTISANAL PENCILSMITH ★
"OL' STEADY HAND"
THE ONLY STEEL... IS IN HIS NERVE
★★★★
PANASONIC KP-310
★ ELECTRONIC AUTO-STOP ★
"IT NEVER SLEEPS"
NO PENCIL IS SAFE FROM ITS JAWS
WHO WILL BE APPOINTED KING?
SAFETY GOGGLES A MUST!!

The Future of Code

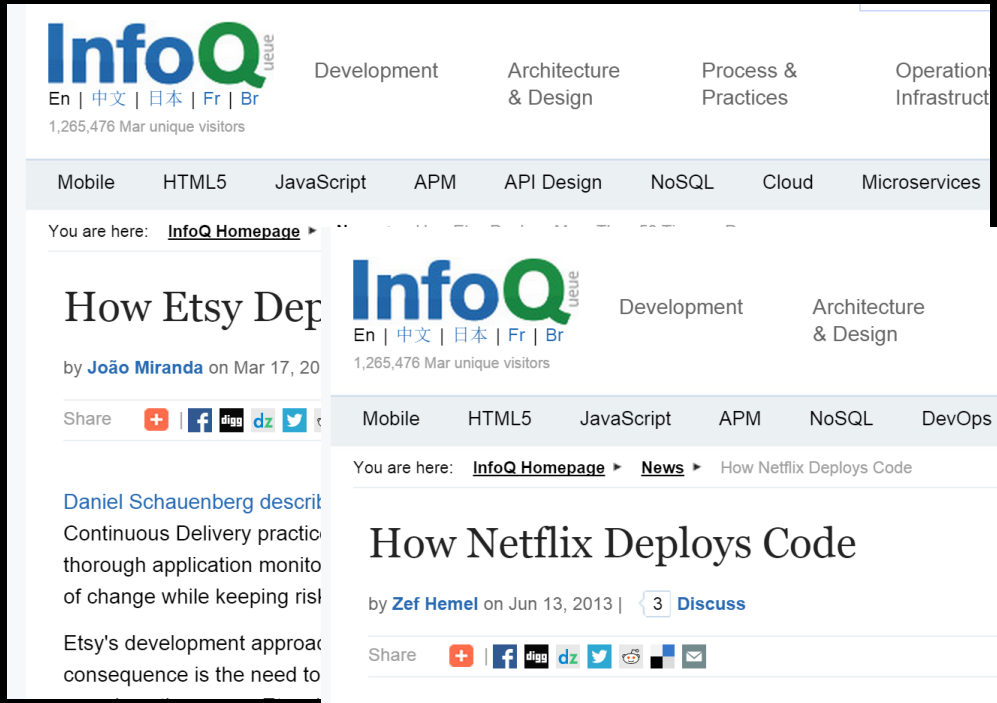
The opposite of Artisinal Programming is…

How Etsy Deploys More Than 50 Times a Day

by **João Miranda** on Mar 17, 2014 | 3 **Discuss**

Daniel Schauenberg described at the last QCon London how Etsy, renowned for its DevOps and Continuous Delivery practices, does 50 deploys/day. A fully automated deployment pipeline, thorough application monitoring and IRC-based collaboration are all important to achieve this rate of change while keeping risk to a minimum.

Etsy's development approach revolves around making many small, continuous changes. A direct consequence is the need to do many deployments a day. In the words of Daniel Schauenberg, at


InfoQ
En | 中文 | 日本 | Fr | Br
1,265,476 Mar unique visitors
Development
Architecture & Design
Process & Practices
Mobile
HTML5
JavaScript
APM
API Design
NoSQL
Cloud
Microservices
You are here: InfoQ Homepage
How Etsy Dep
by João Miranda on Mar 17, 20
Share
Daniel Schauenberg describ
Continuous Delivery practic
thorough application monito
of change while keeping risk
Etsy's development approac
consequence is the need to
InfoQ
En | 中文 | 日本 | Fr | Br
1,265,476 Mar unique visitors
Development
Architecture & Design
Process & Practices
Mobile
HTML5
JavaScript
APM
NoSQL
DevOps
You are here: InfoQ Homepage News How Netflix Deploys Code
How Netflix Deploys Code
by Zef Hemel on Jun 13, 2013 | 3 Discuss
Share
Netflix, the popular movie streaming site, deploys a hundred times per day, witho
Chef or Puppet, without a quality assurance department and without release engi
Netflix built an advanced in-house PaaS (Platform as a Service) that allows each
their own part of the infrastructure whenever they want, however many times they
require. During QCon New York 2013, Jeremy Edberg gave a talk about the infra
built to support this rapid pace of iteration on top of Amazon's AWS.

InfoQ
En | 中文 | 日本 | Fr | Br
1,265,476 Mar unique visitors

Development Archite & Desi

Mobile HTML5 JavaScript APM API Des

You are here: InfoQ Homepage

How Etsy Dep

by **João Miranda** on Mar 17, 20

Share

Daniel Schauenberg describ
Continuous Delivery practic
thorough application monito
of change while keeping risk

Etsy's development approac
consequence is the need to

InfoQ
En | 中文 | 日本 | Fr | Br
1,265,476 Mar unique visitors

Mobile HTML5

You are here: InfoQ Homep

How Netfli

by **Zef Hemel** on Jun 13,

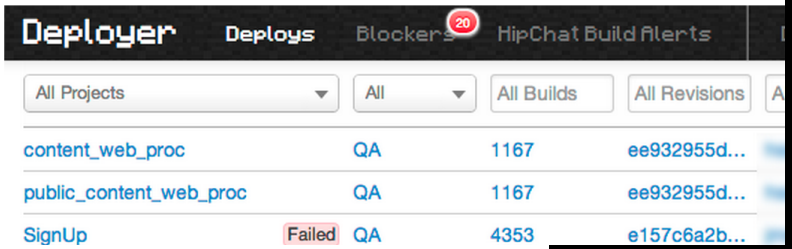
Share

How We Deploy 300 Times a Day

Zack Bloom November 18, 2013

Tweet 1 Like 0 Share

Part of my job at HubSpot is to meet and welcome new po
the most surprising things I get to tell them is that we deplo



Netflix, the popular movie streaming site, deploys a hundred times per day, witho
Chef or Puppet, without a quality assurance department and without release engi
Netflix built an advanced in-house PaaS (Platform as a Service) that allows each
their own part of the infrastructure whenever they want, however many times they
require. During QCon New York 2013, Jeremy Edberg gave a talk about the infra
built to support this rapid pace of iteration on top of Amazon's AWS.

What’s going on here?

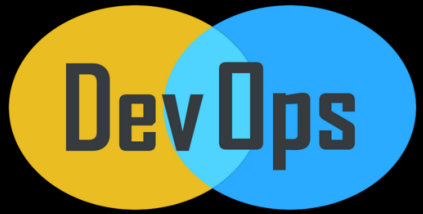
Software Development at the Speed of DevOps

DevOps seeks to improve the speed and quality of software development and support. It involves integrating and automating the work of software developers and IT operations professionals.

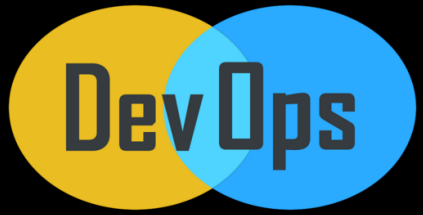
In recent years, some IT organizations have moved to a new software development delivery model called “DevOps.” DevOps refers to a set of practices designed to facilitate interactions between the application development and operations functions, and ultimately aims to increase the speed, improve the quality, and reduce the cost of software development, ongoing enhancement, and support.

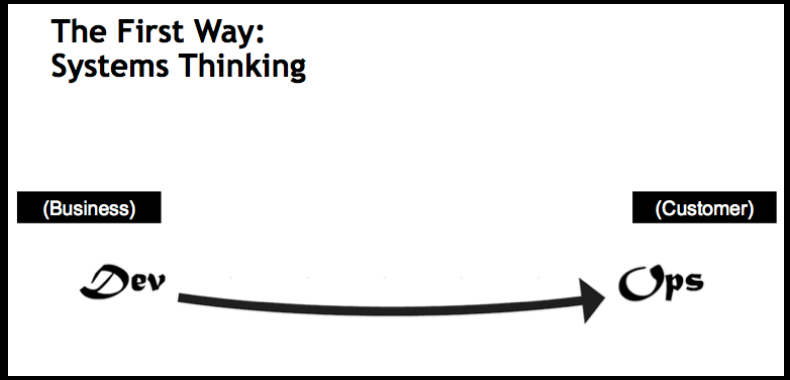
Traditionally, IT organizations have run their development and operations functions separately, largely because the two groups have different missions that yield opposing behaviors. “App Dev”

Yep. DevOps.



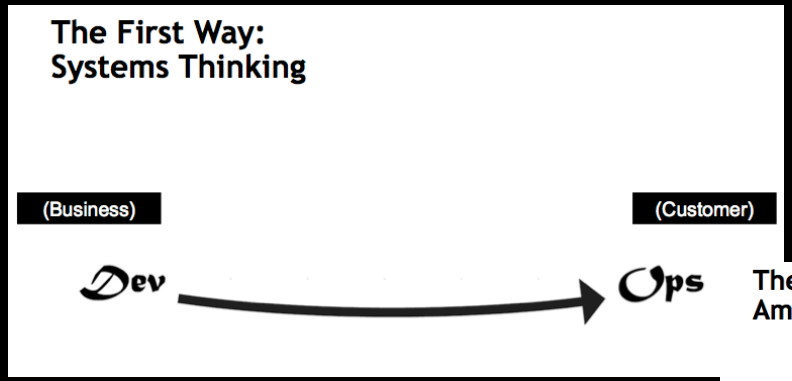
Yep. DevOps.
But for *code.*

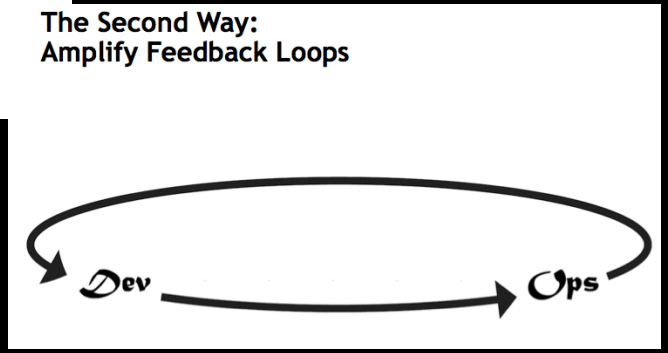




DevOps

The Three Ways: The Principles Underpinning DevOps
By Gene Kim





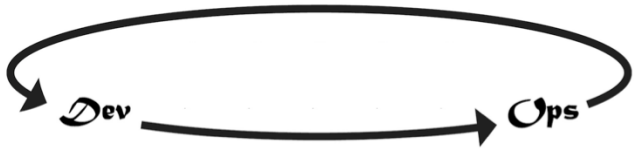
DevOps

The Three Ways: The Principles Underpinning DevOps
By Gene Kim

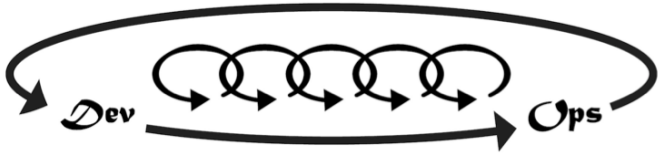
The First Way:
Systems Thinking

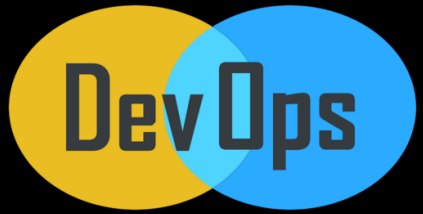
(Business) (Customer)

The Second Way:
Amplify Feedback Loops



The Third Way:
Culture Of Continual Experimentation And Learning



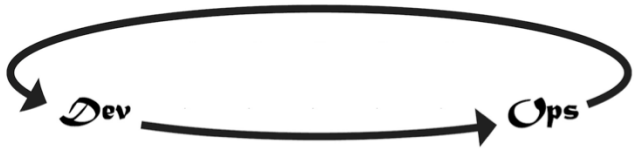


The Three Ways: The Principles Underpinning DevOps
By Gene Kim

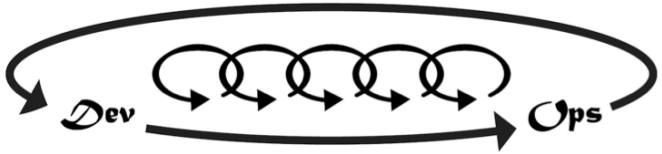
The First Way:
Systems Thinking

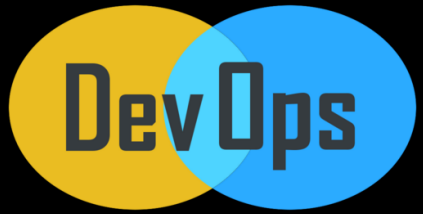
(Business) (Customer)

The Second Way:
Amplify Feedback Loops



The Third Way:
Culture Of Continual Experimentation And Learning





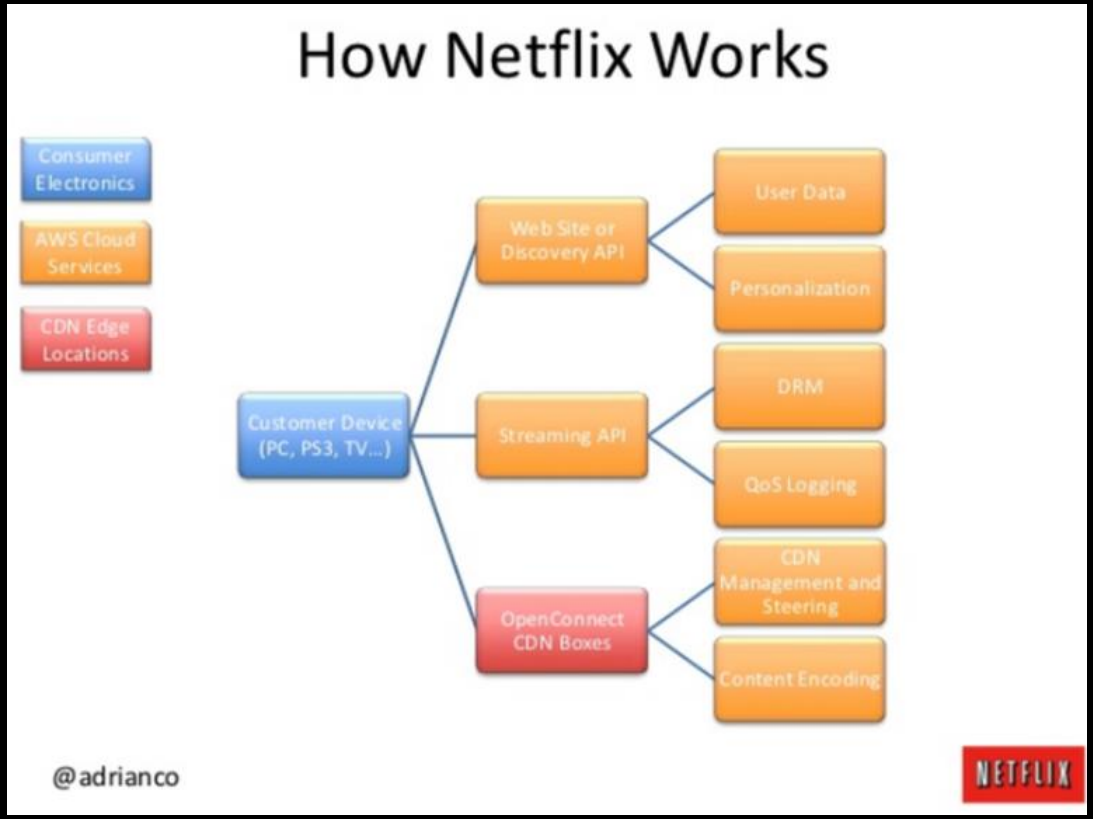
The Three Ways: The Principles Underpinning DevOps
By Gene Kim

Four Pillars of OPS

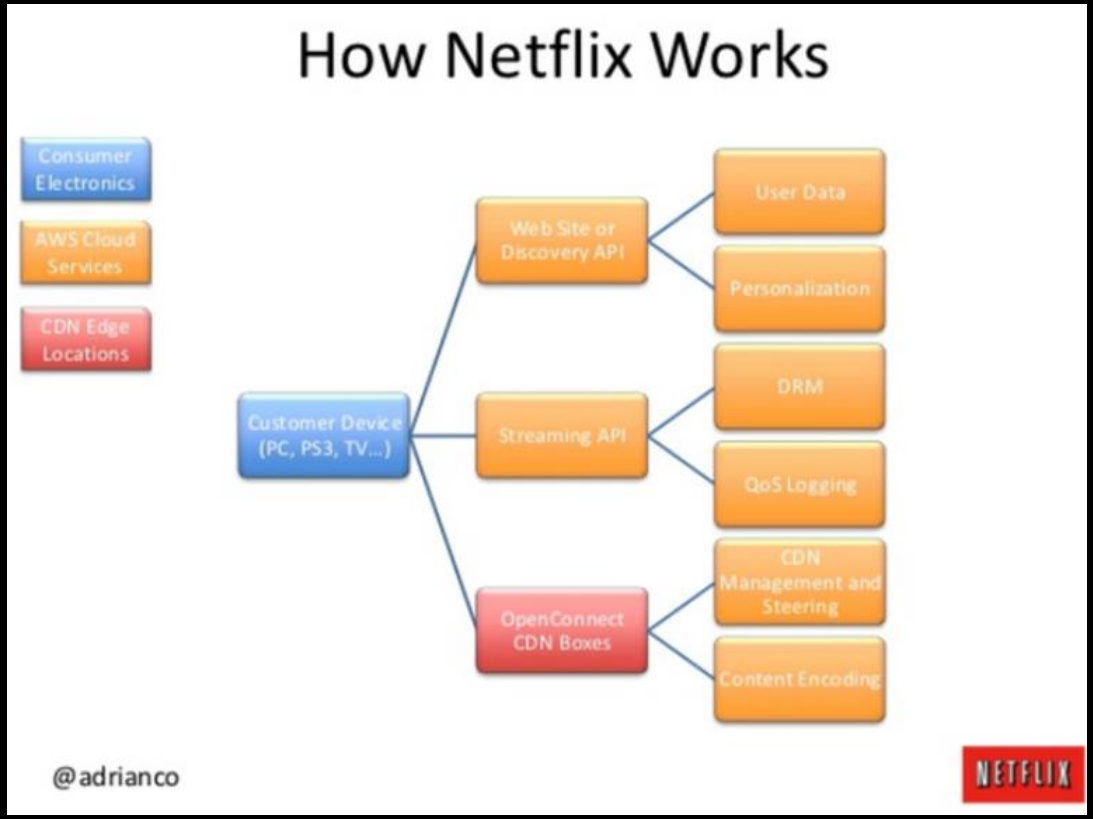
Four Pillars of OPS

- Reduce Cost
- Increase Speed
- Improve Safety/Resiliency
- Provide Visibility/Feedback

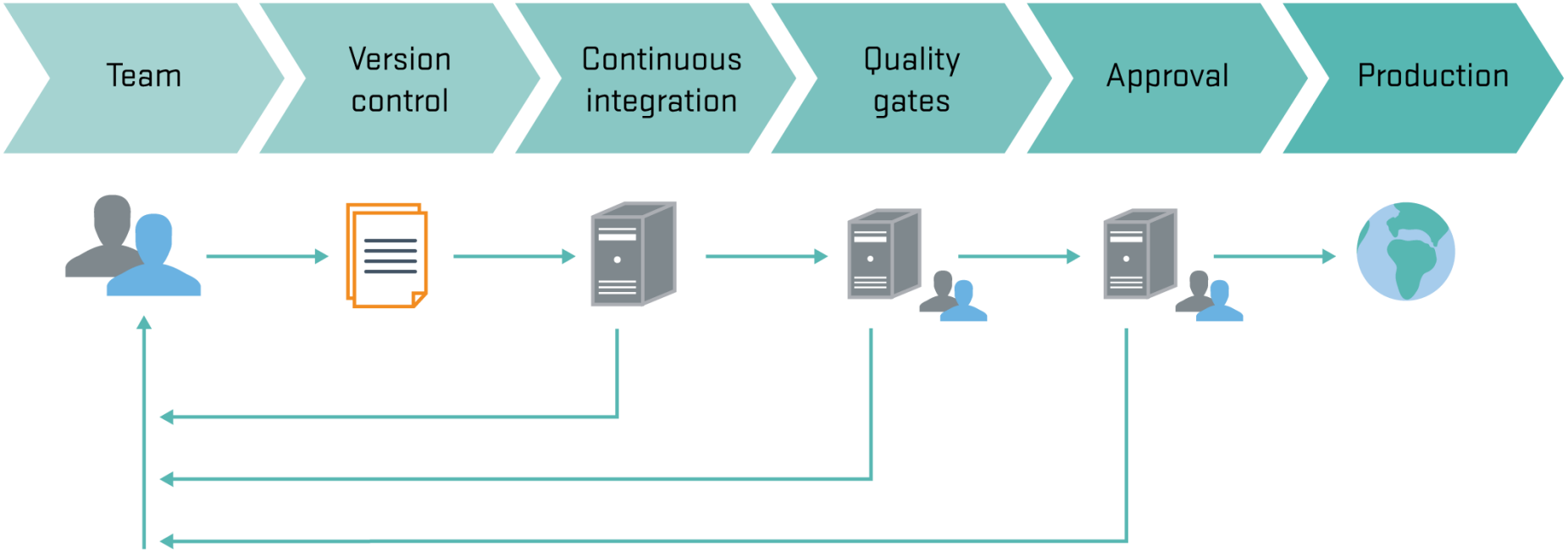
Reduce Cost


How Netflix Works
Consumer Electronics
AWS Cloud Services
CDN Edge Locations
Customer Device (PC, PS3, TV...)
Web Site or Discovery API
Streaming API
OpenConnect CDN Boxes
User Data
Personalization
DRM
QoS Logging
CDN Management and Steering
Content Encoding
@adrianco
NETFLIX

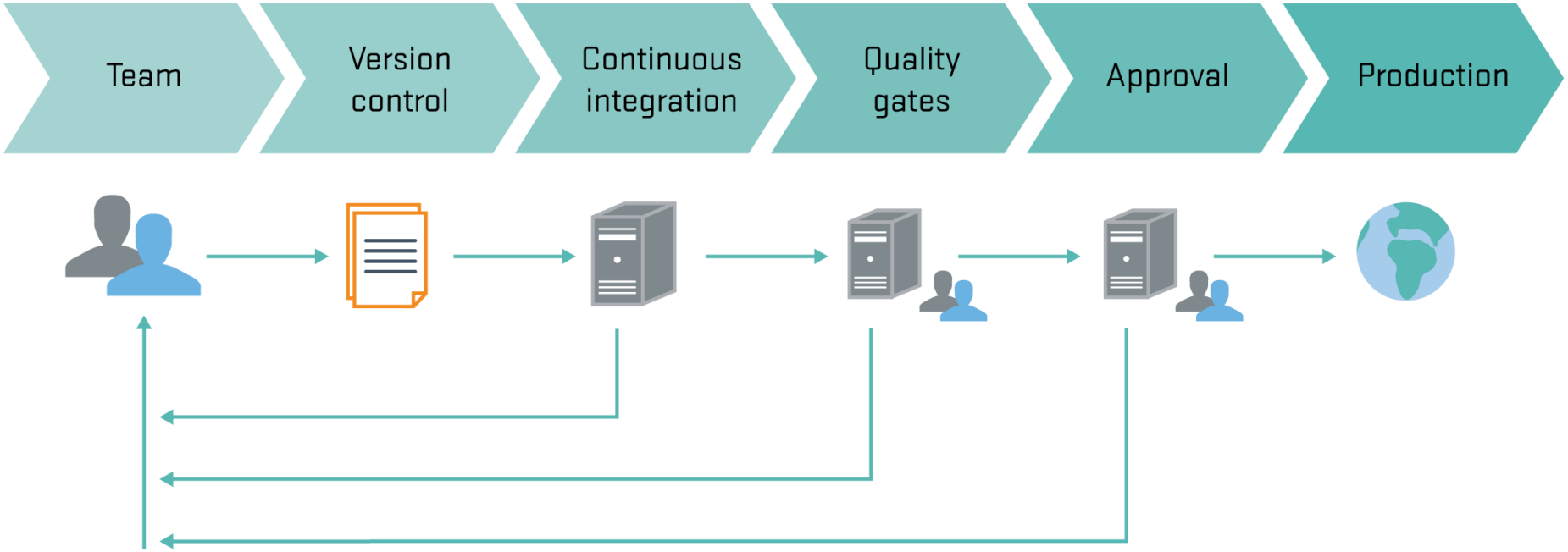
Reduce Cost - Virtualize Hardware



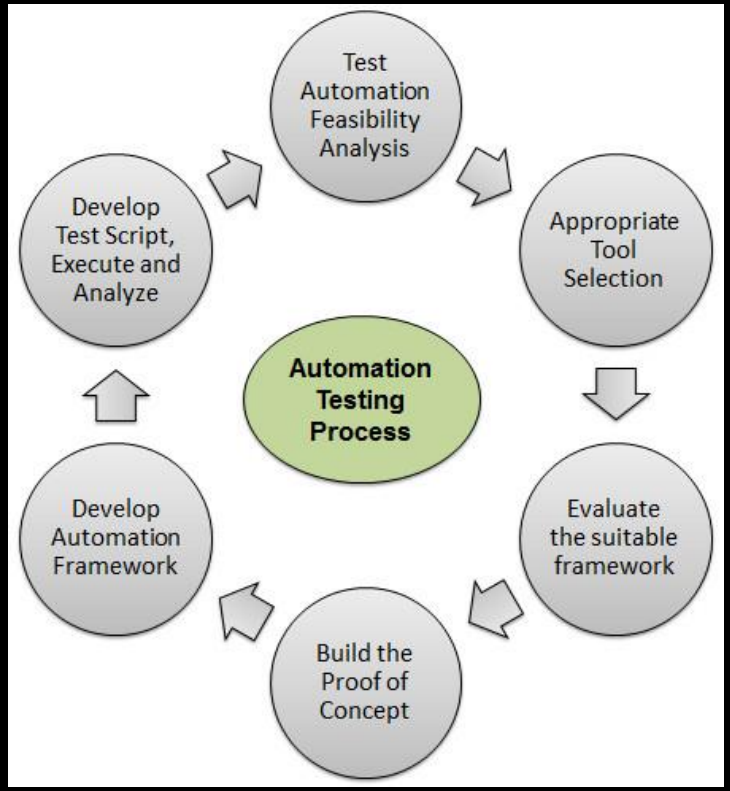
Increase Speed



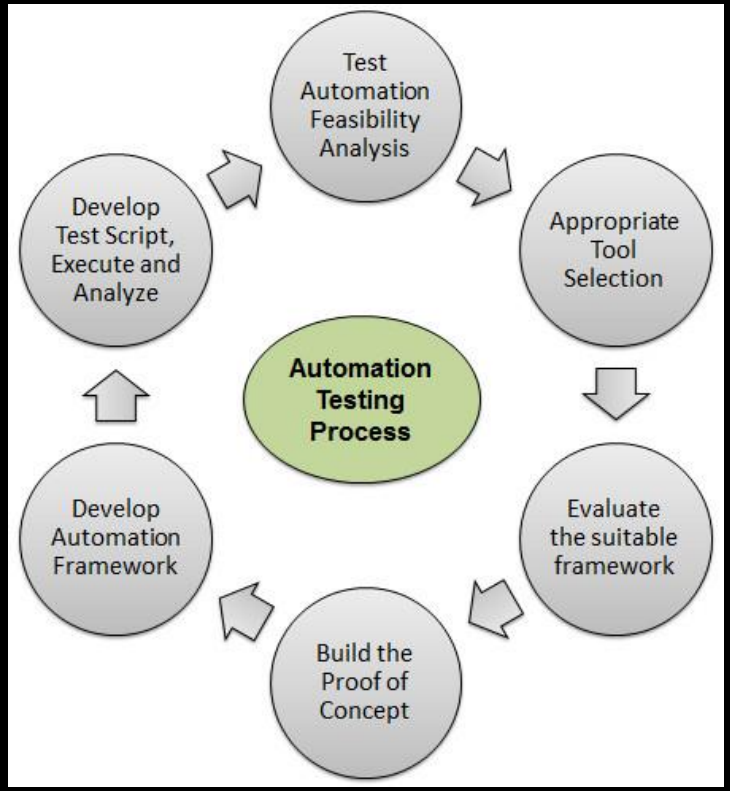
Increase Speed – Automate Deployment



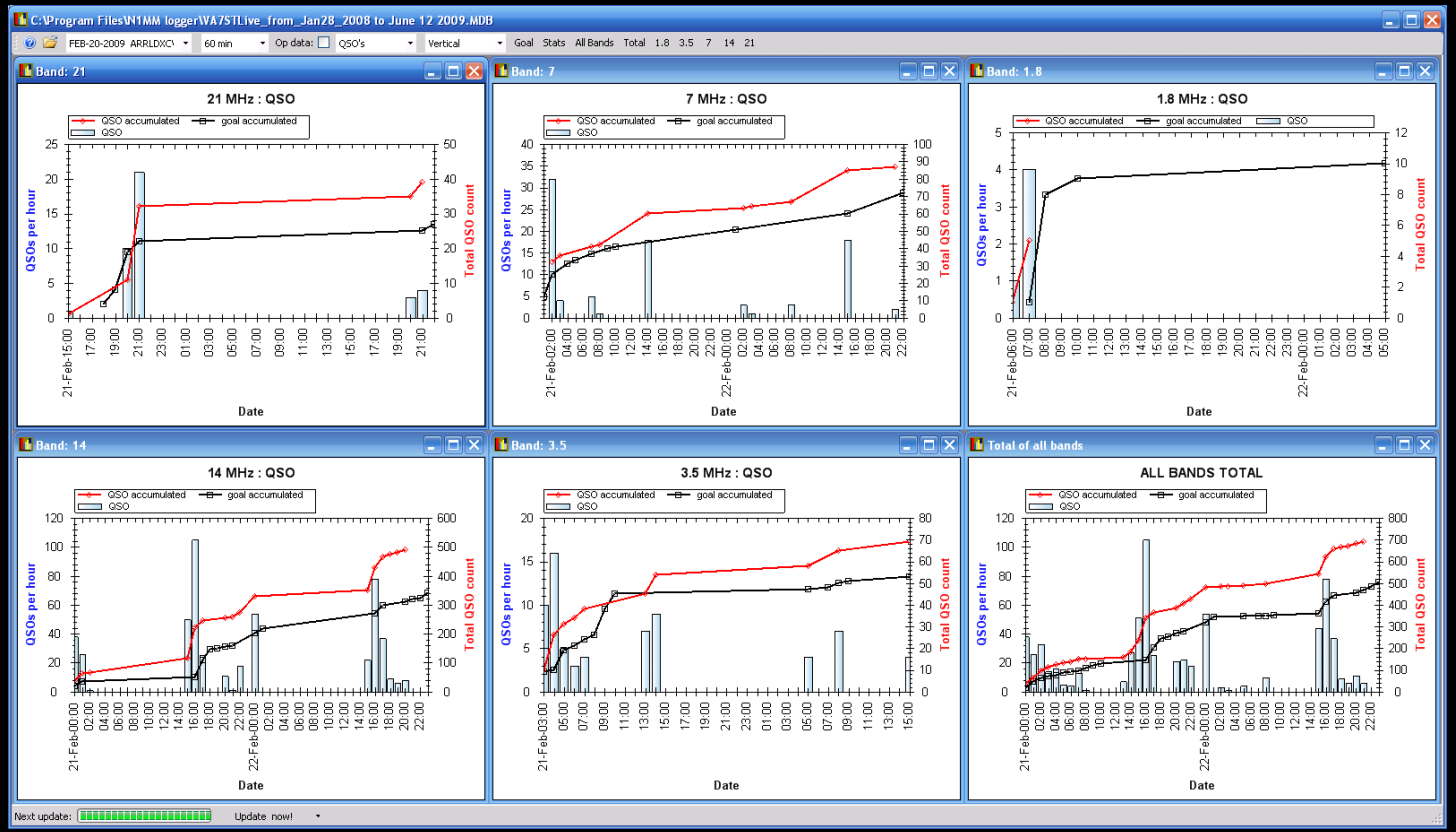
Improve Resiliency



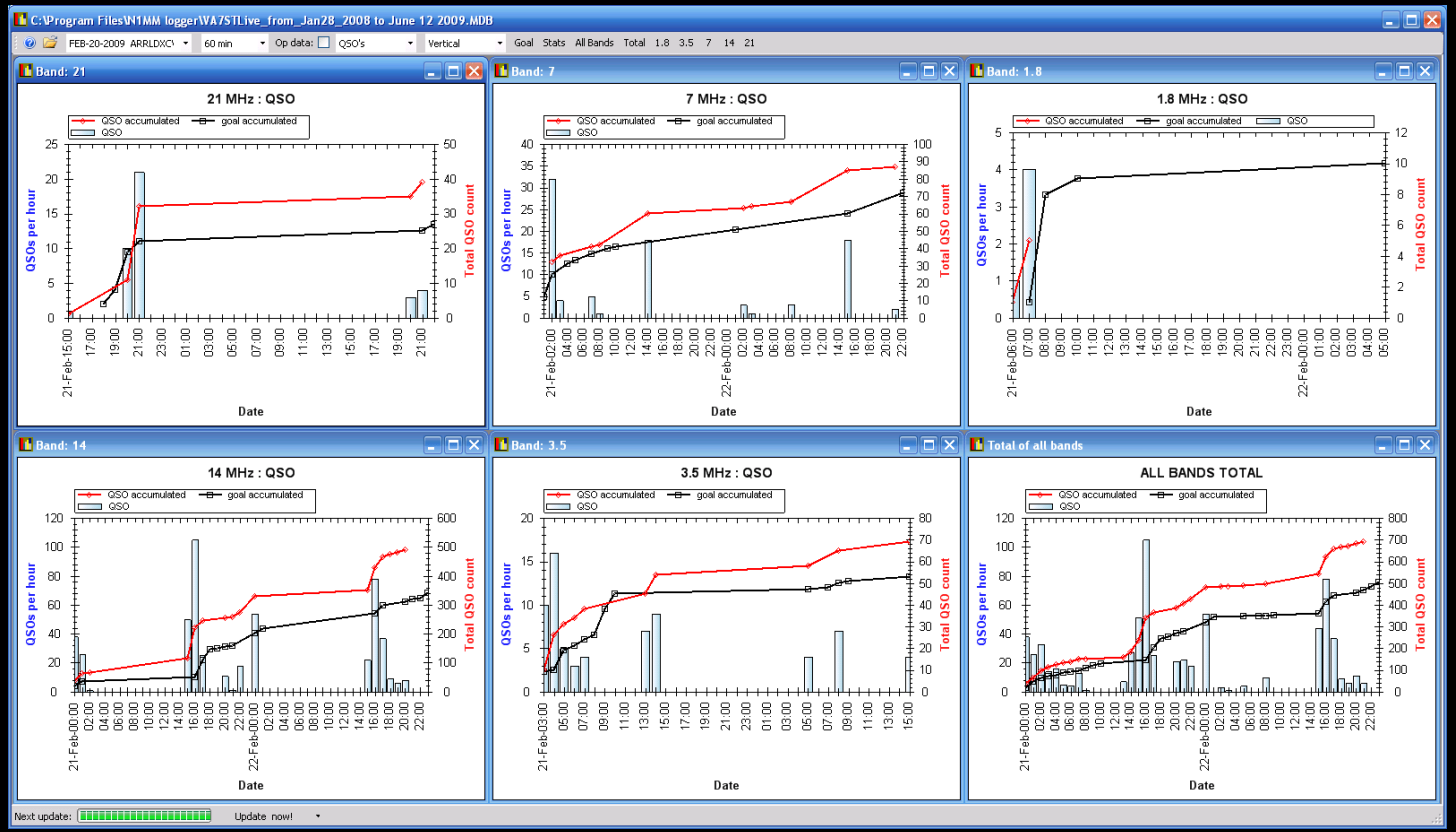
Improve Resiliency – Automated Testing



Provide Visibility



Provide Visibility – Ubiquitous Logging



Same Pillars for DEV

Four Pillars of DEV

- Reduce Cost
- Increase Speed
- Improve Safety/Resiliency
- Provide Visibility/Feedback

Reduce Cost

CA Service Virtualization

Rapidly Create High-Quality Software

CA Service Virtualization

SOLUTION

- Quickly create, edit and deploy virtual services
- Patented "Learning Mode" which automatically learns differences between your test and live environments
- Uses artificial intelligence to translate any protocol when creating virtual services

VALUE

- Dramatically reduce the time it takes to build critical business applications
- Avoid delays caused by constrained or unavailable resources
- Enable more comprehensive testing that begins earlier in the cycle so more defects are found sooner to improve quality and reduce costs

Reduce Cost - Virtualize System

CA Service Virtualization

Rapidly Create High-Quality Software

CA Service Virtualization

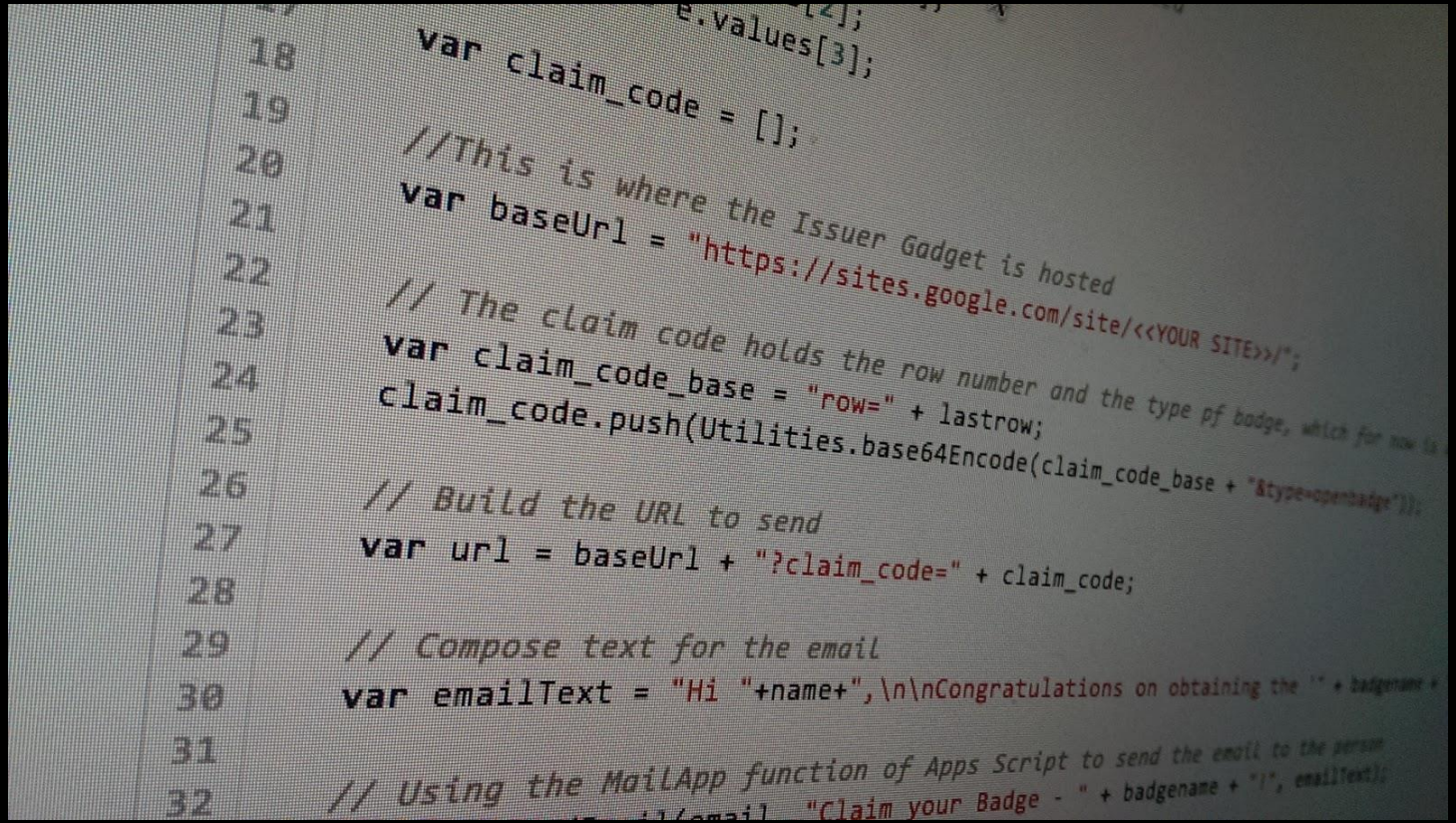
SOLUTION

- Quickly create, edit and deploy virtual services
- Patented "Learning Mode" which automatically learns differences between your test and live environments
- Uses artificial intelligence to translate any protocol when creating virtual services

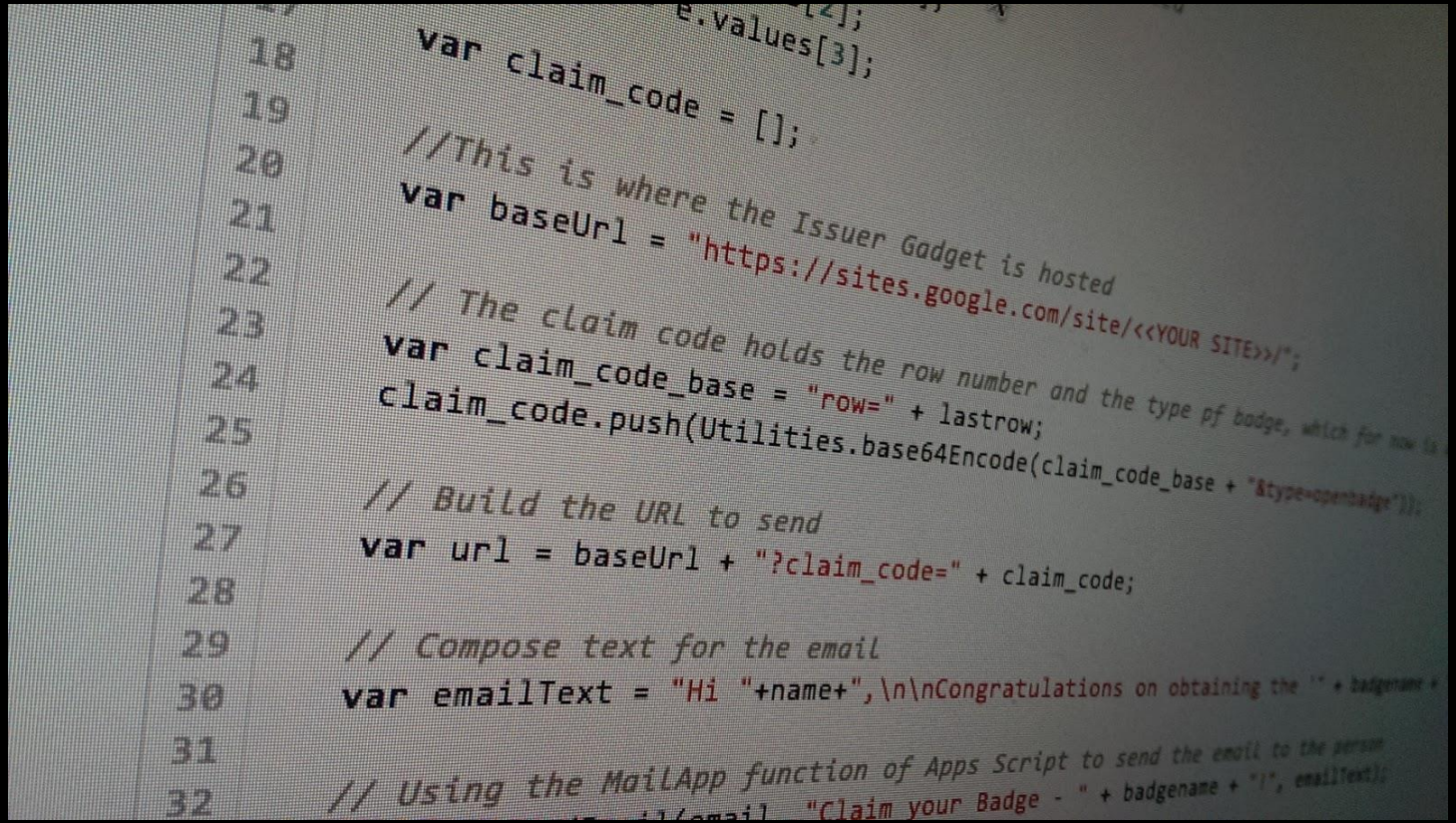
VALUE

- Dramatically reduce the time it takes to build critical business applications
- Avoid delays caused by constrained or unavailable resources
- Enable more comprehensive testing that begins earlier in the cycle so more defects are found sooner to improve quality and reduce costs

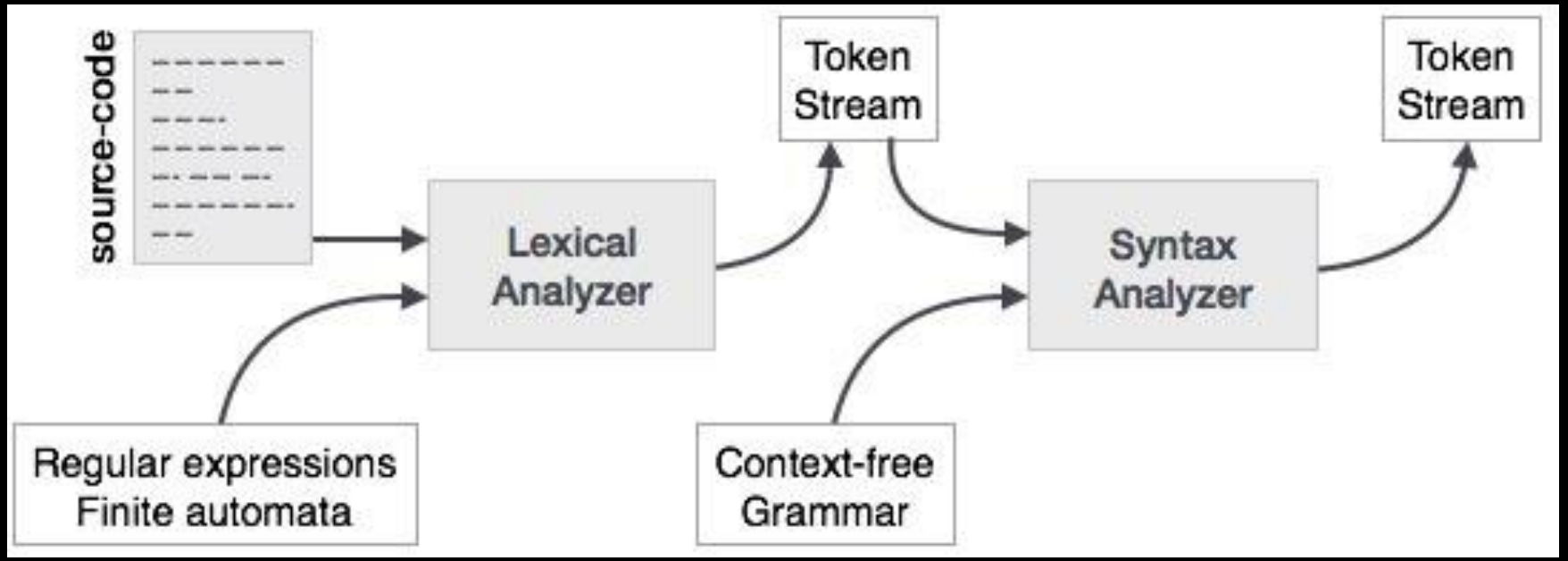
Increase Speed



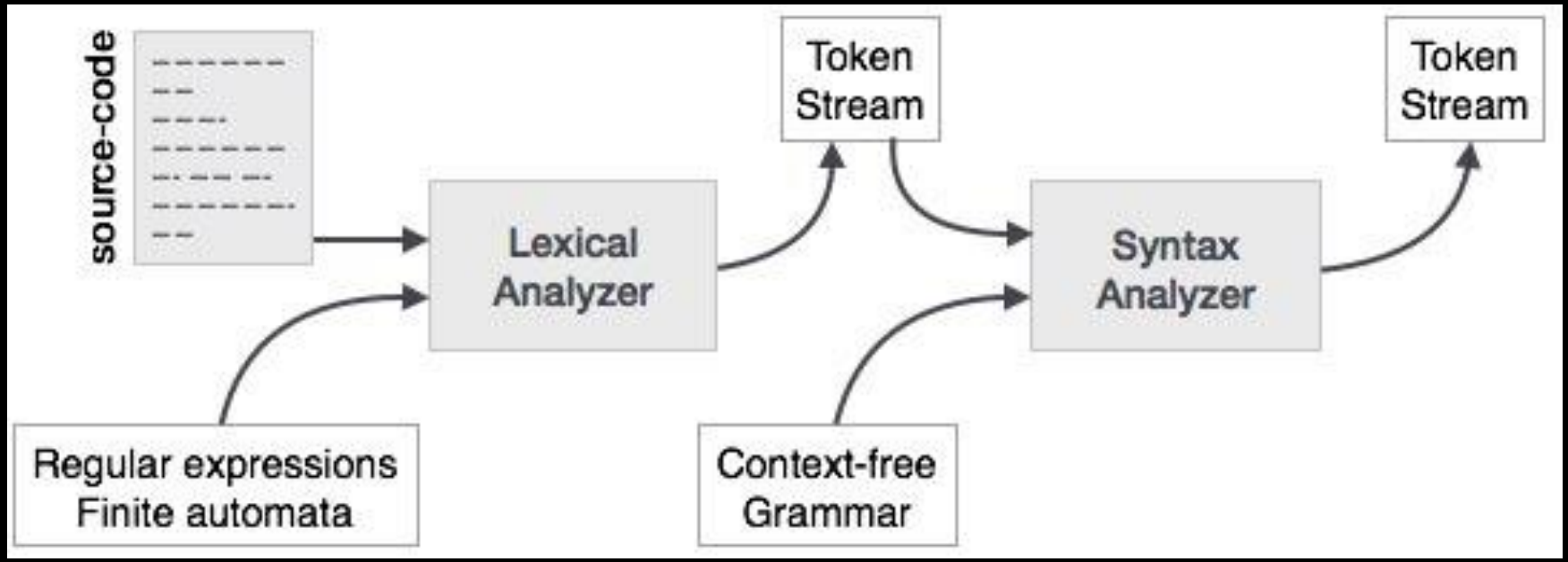
Increase Speed – Automate Code



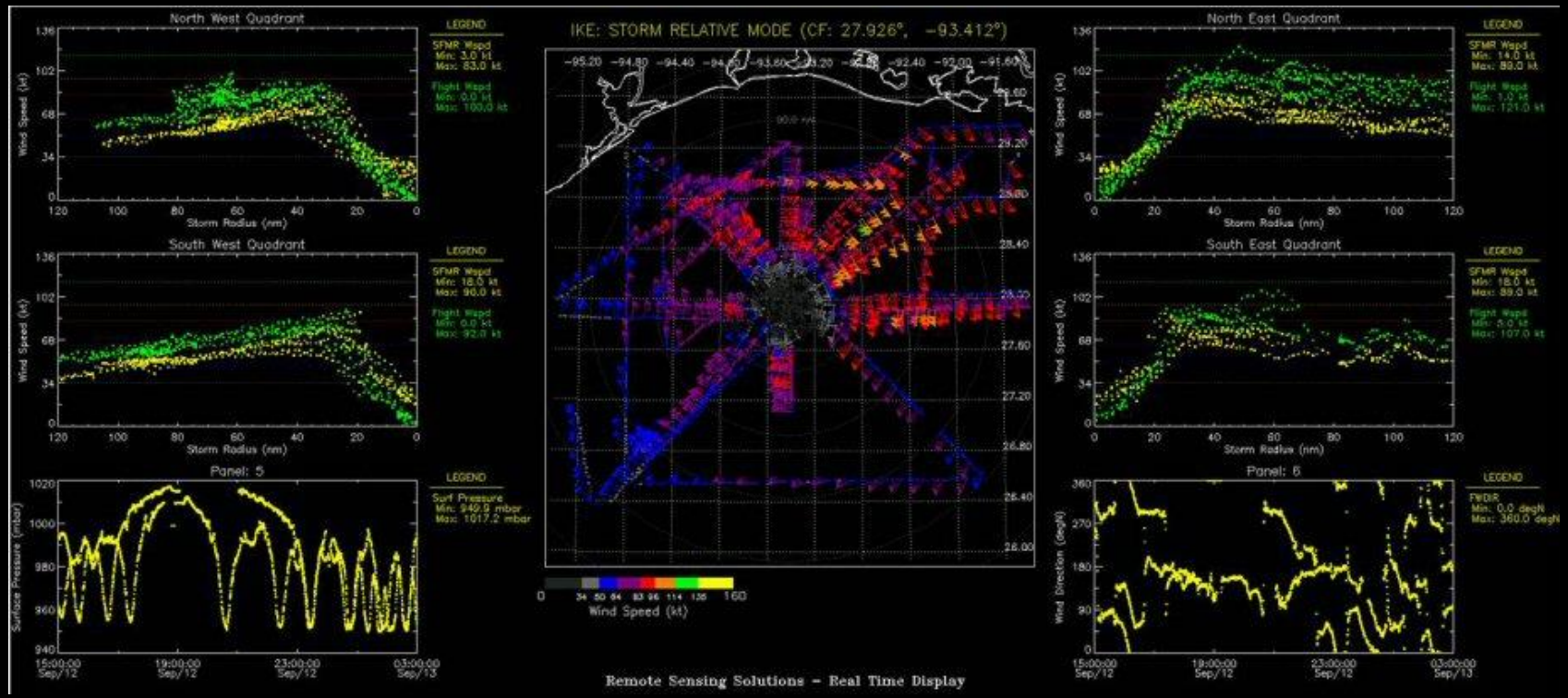
Improve Resiliency



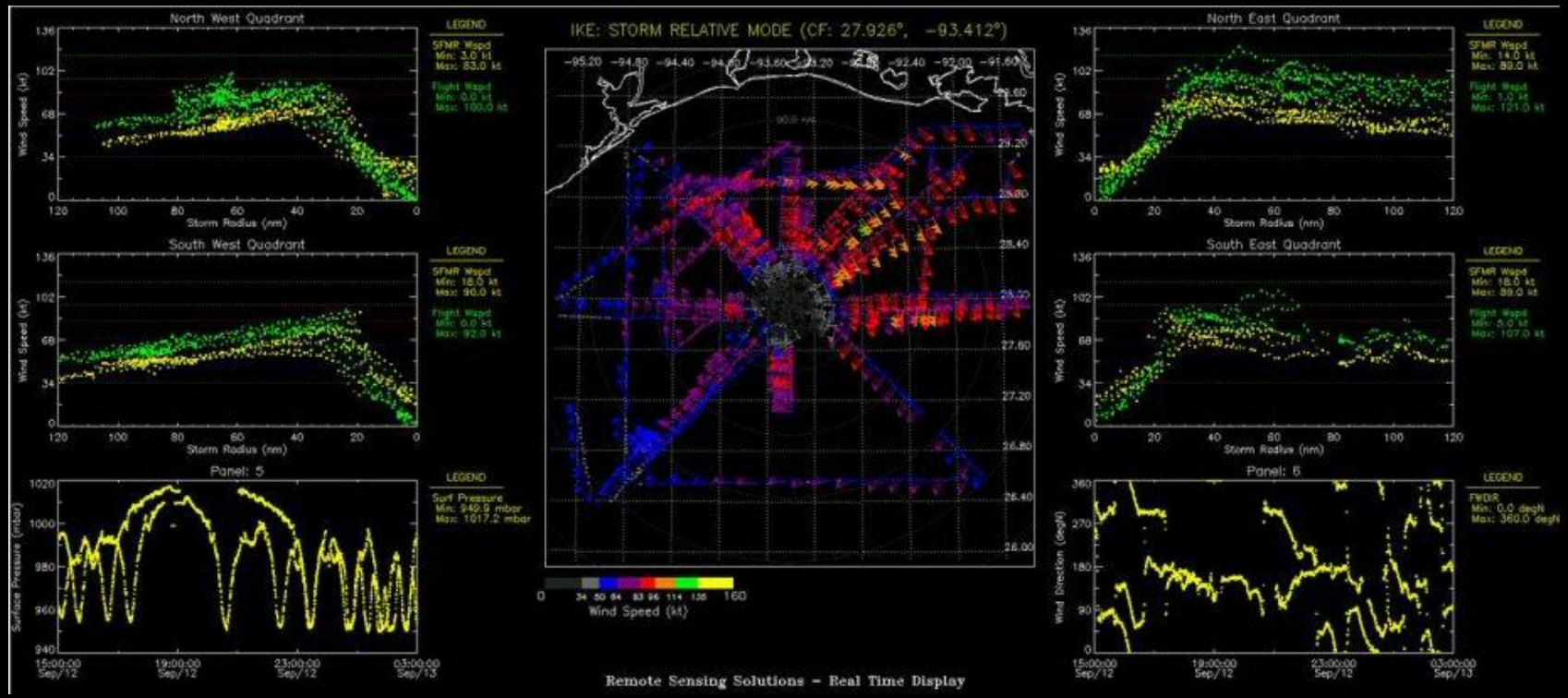
Improve Resiliency – Code Analytics



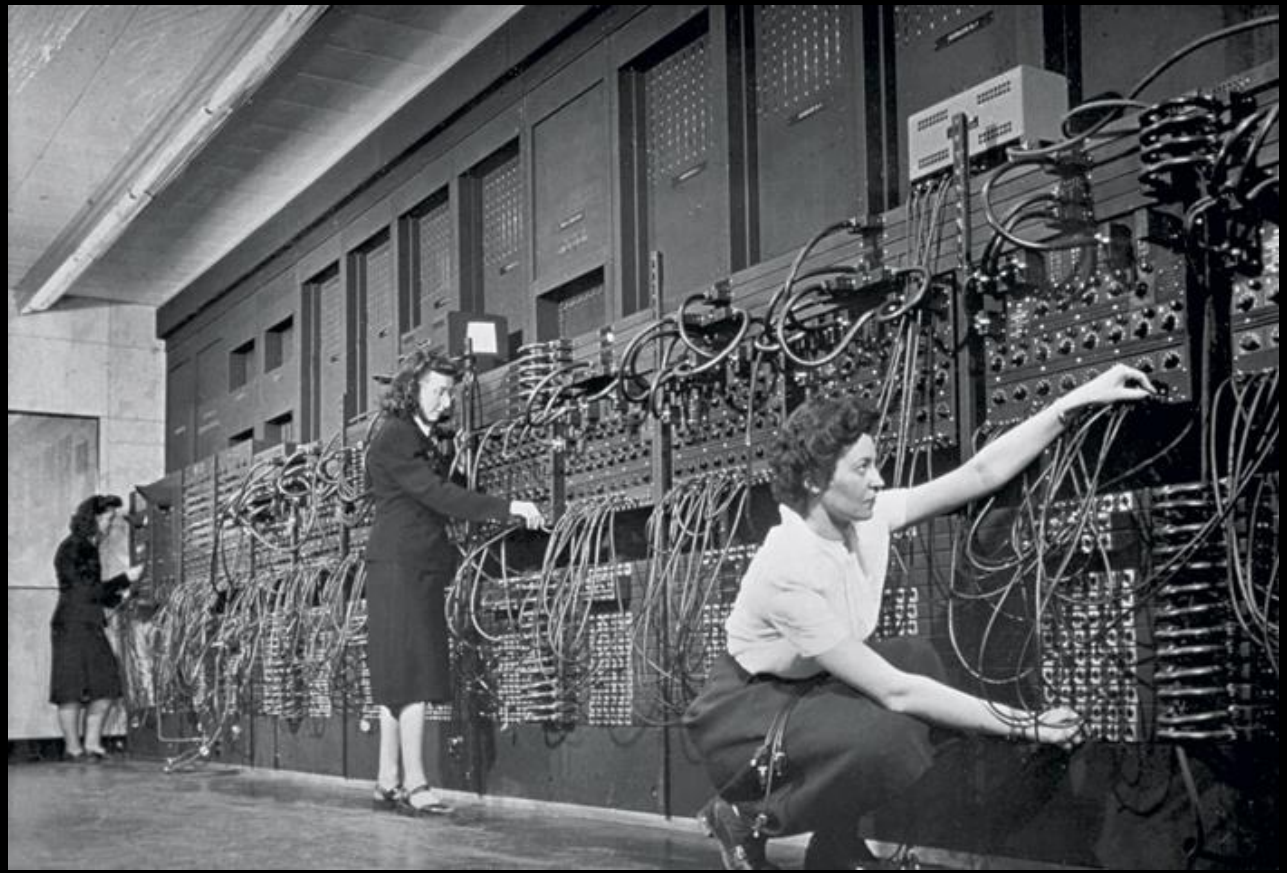
Provide Visibility



Provide Visibility – Runtime Visualization

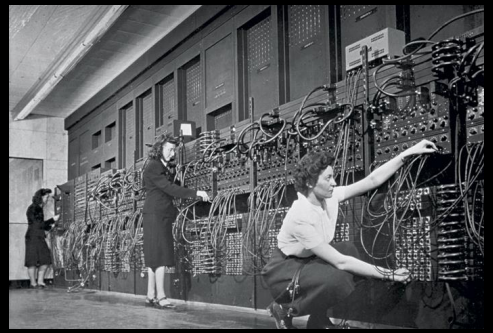


Summary

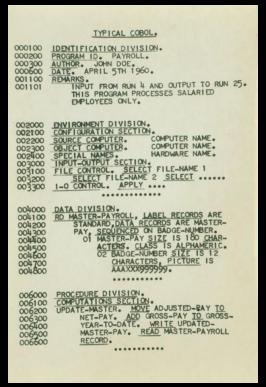


Jennings, Wescoff, & Lichterman, 1946

The program was the machine.



The machine was the code.



```
                    TYPICAL COBOL.

000100  IDENTIFICATION DIVISION.
000200  PROGRAM ID.  PAYROLL.
000300  AUTHOR.  JOHN DOE.
000600  DATE.  APRIL 5TH 1960.
001100  REMARKS.
001101      INPUT FROM RUN 4 AND OUTPUT TO RUN 25.
            THIS PROGRAM PROCESSES SALARIED
            EMPLOYEES ONLY.

002000  ENVIRONMENT DIVISION.
002100  CONFIGURATION SECTION.
002200  SOURCE-COMPUTER.      COMPUTER NAME.
002300  OBJECT-COMPUTER.      COMPUTER NAME.
002400  SPECIAL NAMES.        HARDWARE NAME.
003000  INPUT-OUTPUT SECTION.
003100  FILE CONTROL.  SELECT FILE-NAME 1
003200      SELECT FILE-NAME 2  SELECT ******
003300  I-O CONTROL.  APPLY ****
                  ***********

004000  DATA DIVISION.
004100  FD MASTER-PAYROLL, LABEL RECORDS ARE
004200      STANDARD, DATA RECORDS ARE MASTER-
004300      PAY, SEQUENCED ON BADGE-NUMBER.
004400      01 MASTER-PAY. SIZE IS 100 CHAR-
004500          ACTERS. CLASS IS ALPHAMERIC.
004600          05 BADGE-NUMBER SIZE IS 12
004700              CHARACTERS. PICTURE IS
004800              AAAXXX999999.
                  ***********

006000  PROCEDURE DIVISION.
006100  COMPUTATIONS SECTION.
006200  UPDATE-MASTER.  MOVE ADJUSTED-PAY TO
006300      NET-PAY.  ADD GROSS-PAY TO GROSS-
006400      YEAR-TO-DATE.  WRITE UPDATED-
006500      MASTER-PAY.  READ MASTER-PAYROLL
006600      RECORD.
                  ***********
```

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

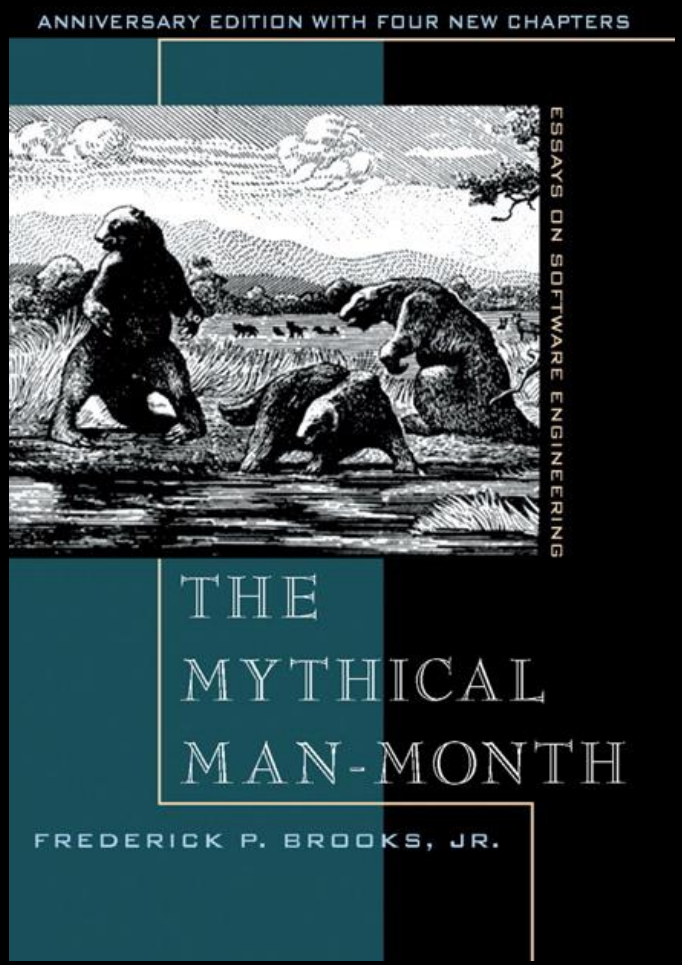
Dr. Conway is manager, peripheral systems research, at Sperry Rand's Univac Div., where he is working on recognition of continuous speech. He has previously been a research associate at Case Western Reserve Univ., and a software consultant. He has an MS in physics from CalTech and a PhD in math from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

Communication dictates design.

-- Mel Conway, 1967



Brooks' Law

"*Adding manpower to a late software project makes it later*."

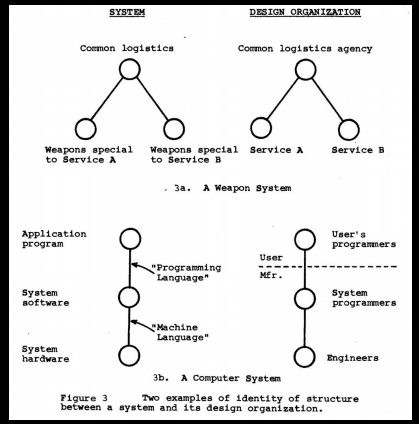
-- Fred Brooks, 1975

Dunbar Groups

Intimate friends: 5
Trusted friends: 15
Close friends: 35
Casual friends: 150

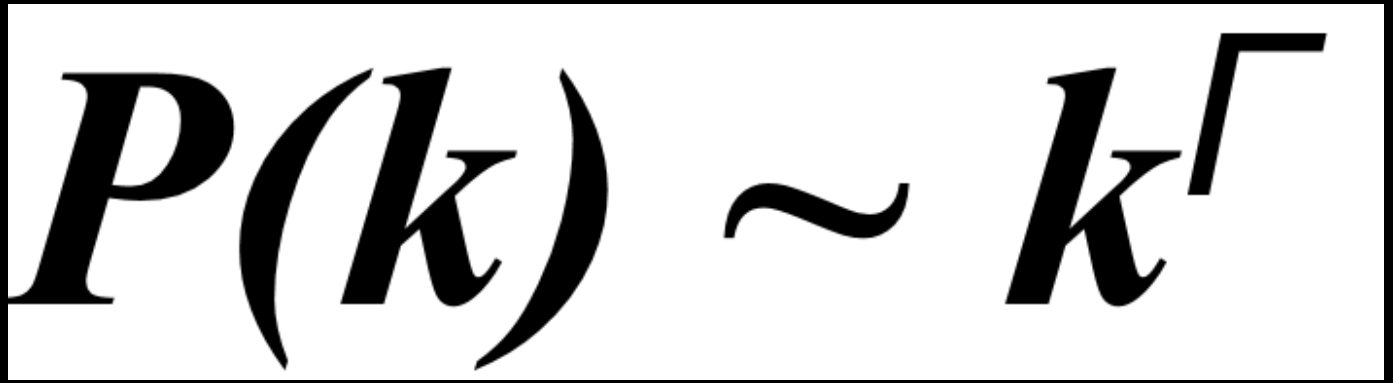
-- Robin Dunbar, 1992

The machine was the organization.

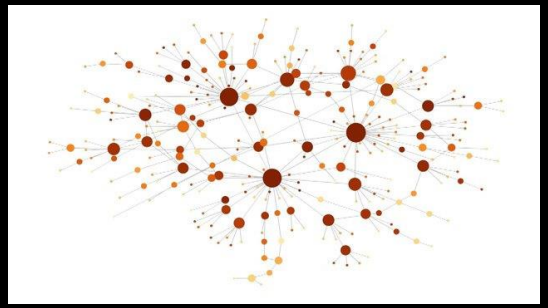


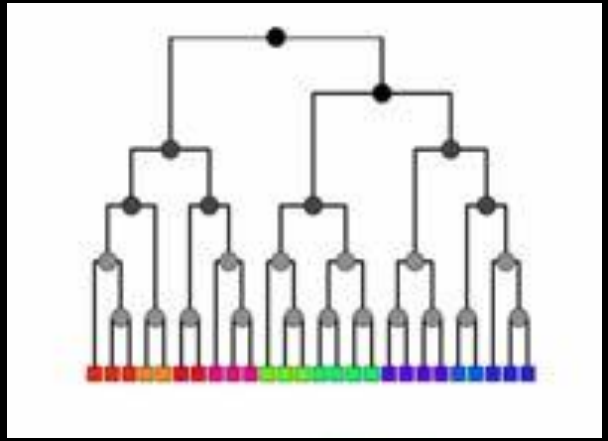

LAMONT
LIBRARY

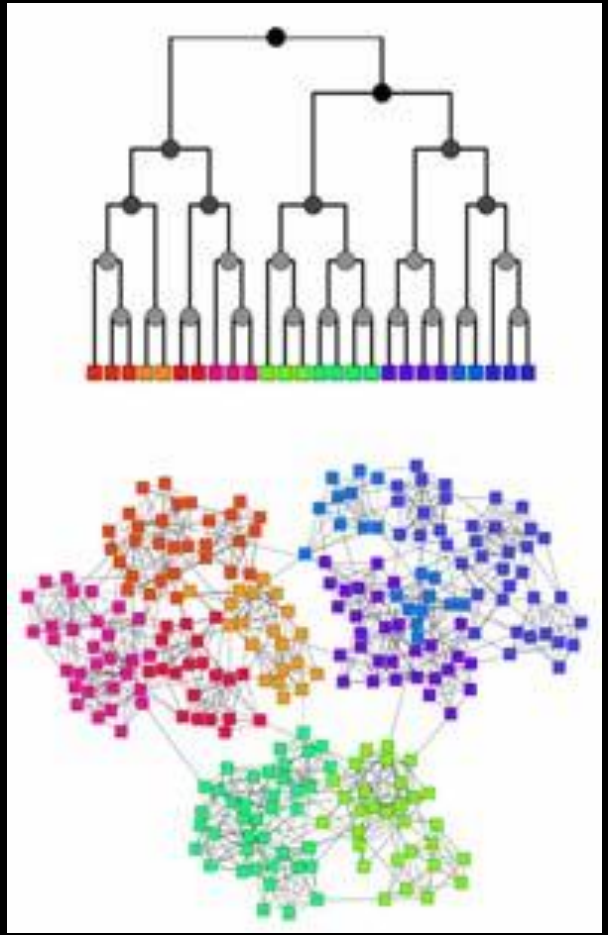

ABA
GPA I
AGB I
target games


P(k) ~ k^Γ

*The machine is the **network**.*







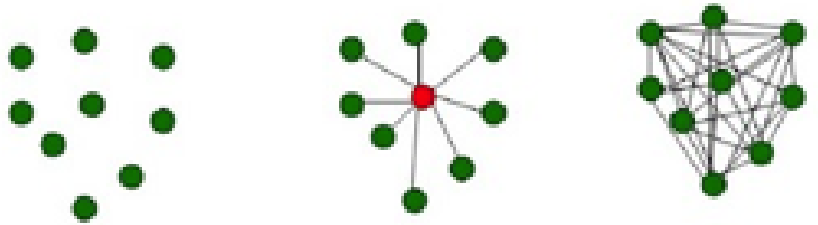
Four Pillars of DEV

- Reduce Cost
  (Virtualize the System)
- Increase Speed
  (Automate/Eliminate Code)
- Improve Safety/Resiliency
  (Code Analysis)
- Provide Visibility/Feedback
  (Runtime Visualization)

Bridging the Gap

working smarter in networks

Artisans - Hierarchies - Networks



~19th C +/- 20th C 21st C

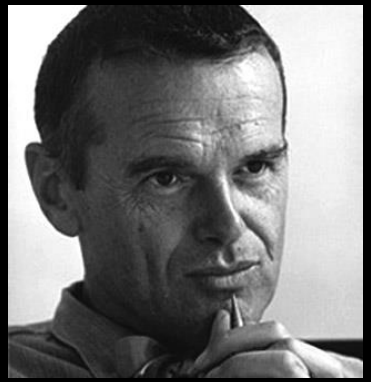
Designing the Enterprise, Ken Barnett 2011 (blog)

“We must break away from the sequential. We must state definitions and provide for priorities and descriptions of data. ***We must state relationships, not procedures.”***

- Grace Hopper, 1962

“Recognizing the need is the primary condition for design.”

– Charles Eames



Those who ignore the mistakes of the future are bound to make them.”

– Dr. Joseph Miller

Beyond Source Code

Mike Amundsen,
API Academy / CA
@mamund