cOOherentBPR – A pattern language to build agile organizations

Michael A. Beedle
Framework Technologies, Inc.
Shaumburg, IL 60656
(847) 490-7110
beedlem@fti-consulting.com

Abstract

BPR and object oriented technology are - and have been for the last few years, some of the two most important business and technical currents. However, it is often seen that applications developed independently for each business process result in "system architecture silos". Therefore, the implementation of OO enterprise architectures brings ample business benefits in BPR environments such as: increased "enterprise conceptual integrity", reusability, generativity, and increased business effectiveness (cost, quality, service or speed). However, the simultaneous implementation of OO architectures and BPR also demands many changes in the business organization, its software development organization and its enterprise architecture. In particular, it requires a workflow manager to be present in the system architecture. This paper describes the structural and temporal evolution of the resulting business architecture when BPR is implemented by an OO enterprise architecture in terms of a pattern language.

1. Reengineering Review

BPR (Business Process Reengineering), is the leading functional strategy followed by most Global 2000 organizations. Its premises are revolutionary. It calls for a new order of organization that relies on the infiltration of software applications and other technologies as an enablers of the core business processes of the organization, that has proven to enhance productivity by factors of up to 1000%, while enhancing the work experiences of their employees. Simply said, the goal of reengineering is to create hyper-productive but enjoyable work environments that promote the growth and human comfort of their employees. Reengineering as defined by Hammer is:

> *The fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical contemporary measures of performance, such as cost, quality, service and speed.*

A key concept to understand how the business processes of an organization are redesigned is the *Business System Diamond.* This concept states that the definition of *Business Processes* lead to *Jobs and Structures*, which in turn require *Management and Measurement Systems*, that reinforce a set of *Values and Beliefs.* Every organization has a Business System Diamond, with the one in reengineered environments shinning brighter. Presenting a pattern language for BPR allows us to show important "organizational constructs" that are typically hidden beneath the artificial process steps of a given *BPR method.* However, enabling applications must be provided for true reengineering to emerge and these applications are best provided by means of OO enterprise architectures.

2. COOherentBPR and the Alexandarian paradigm

The paradigm that Alexander proposes to build things is based on three concepts: *The Quality*, *The Gate* and *The Timeless Way* [Alexander79]. This is how the COOherentBPR pattern language relates to these concepts:

The Quality

The COOherentBPR pattern language intends to create organizational architectures and implementations with *The Quality*. That is organizational designs that are "live" (flexible, extensible, adaptable, reusable), but most importantly, that create a working environment that increases the quality of life of its employees in the social and physical senses. It has been noted by Coplien, that the "social architecture" of a business organization is the prevalent factor in determining the "quality of life" at work [Coplien95]. This quality is experienced in the social sense, in the business organization by the reporting and interaction relationships among the employees. For example patterns such as Leader, Business Architect, Process Owner, Coach, Case Team, and Case Worker reflect these values.

The Gate

The Gate is the CPL (Common Pattern Language), which is the universal network of patterns and pattern relationships dedicated to a domain. A pattern language for a specific design is chosen by the designer from the Common Pattern Language; however, this chosen language needs to be a morphologically and functional complete. The COOherentBPR pattern language attempts to contribute in the first order of scale to the architecture of a business organization. And it attempts to be the "holder pattern language" for other patterns and pattern languages to that are needed to complete the business and systems architecture of the organization, referencing for example, other org-patterns, workflow patterns and system architecture patterns.

The Way

The COOherentBPR pattern language is applied using The Way. That is we apply one pattern at a time, "differentiating space" to successively evolve an initial architecture and unfold it into a "live design", or said it in Alexander terms, a design with The Quality. As such, the patterns in the COOherentBPR pattern language are not meant to be applied linearly as thy are written down in this document, but rather, each of the patterns is linked to other patterns through its resulting context.

Advanced Concepts

The evolution of the pattern language can be also be seen as the continuous application of Structured Preserving Transformations, which evolve a design with "wholeness" - the vision of the leader, into a reality of "social centers" that have the "universal properties" of live designs [OOPSLA96].

COOherentBPR, Agility, CAS, Learning Organizations and Generativity

The COOherentBPR pattern language is the company's "Knowledge Repository", and allows the company to behave as a CAS (Complex Adaptable System).

When the pattern language is taught to the whole organization, it allows its "adaptable agents" - its people, to share a common vision and a common language for the architecture of the organization. As they implement these patterns (recognizing contexts through their "sensors"), the resulting contexts of the patterns will encourage other patterns in the pattern language to be applied (through their "effectors"). This property is what is termed as "generativity" - applying one pattern encourages the rest of the patterns to be applied, eventually generating a new unique architecture.

This kind of organization is also described as having "agility", because it is composed of adaptable "agents" willing to learn and try new things. But it also means that as the people in the organization apply new concepts, that new "emergent behaviors" should be expected while they do that, and therefore allowing for the "generativity" of new business architectures.

From this perspective the COOherentBPR pattern language acts like DNA in a living organism, that tells the business organization when and how to "differentiate" its cells. Therefore, it is extremely important that the whole business organization assimilates the pattern language, and that is willing to accept new information and add it to its repository knowledge - the pattern language.

This kind of organization has also been described by Peter Senge as a "Learning Organization" [Senge90], that has a Shared Vision, understands the same Mental Models, is capable of Team Learning and whose individuals are ever seeking their own Personal Mastery and practice Systems Thinking.

3. The Architecture of the COOherentBPR pattern language

The COOherentBPR pattern language describes the evolution (social, spatial and temporal) of a business and it has been written in the same sprit in which Alexander wrote his pattern language [Alexander78]. Navigating the pattern language “tells the reengineering story”. That is, it provides the reader with recurring business constructs that occur in the reengineering environment. (In this paper only a fraction of these patterns are presented in full due to space limitations. Where deemed necessary, "pattlets" were used instead of patterns.) This seemed to be the best compromise between size and "completeness" of the pattern language.

From a process point of view, one can also identify milestones that mark stages through the reengineering process such as: Strategic Assessment, Business Analysis, Business Design, Business Evolution and Business Maintenance. Figure 1, marks the stages where the patterns are most likely to occur when contrasted with an iterative incremental BPR life-cycle, for example Pattern Based Reengineering [Beedle97].

However, one must realize that in the business evolution phase, iterations to deliver complete or partial applications and business processes are delivered through an iterative incremental cycle.

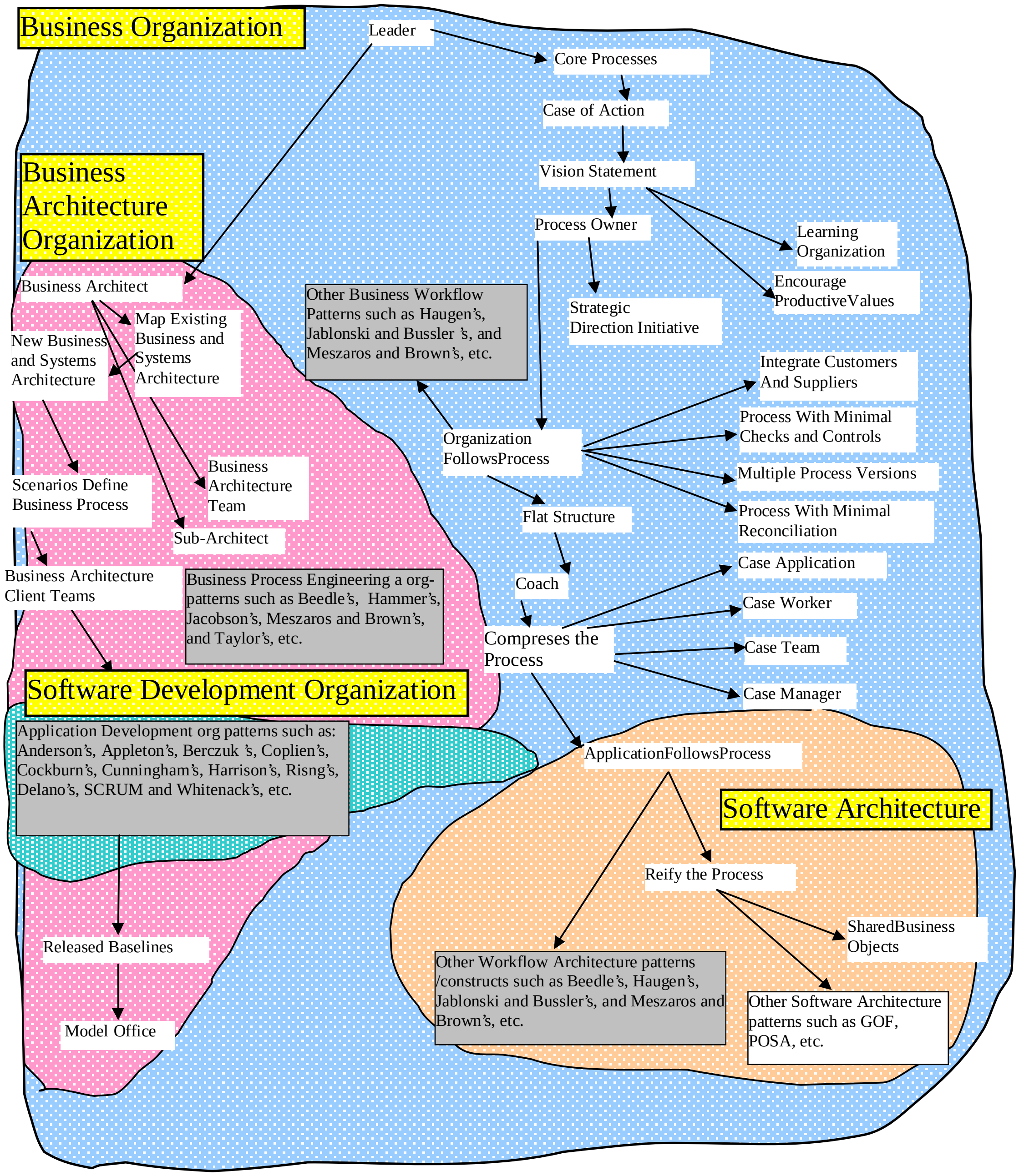
Business Organization
Leader
Core Processes
Case of Action
Vision Statement
Business Architecture Organization
Process Owner
Learning Organization
Business Architect
Encourage ProductiveValues
Other Business Workflow Patterns such as Haugen’s, Jablonski and Bussler ’s, and Meszaros and Brown’s, etc.
Strategic Direction Initiative
Map Existing Business and Systems Architecture
New Business and Systems Architecture
Integrate Customers And Suppliers
Organization FollowsProcess
Process With Minimal Checks and Controls
Business Architecture Team
Multiple Process Versions
Scenarios Define Business Process
Process With Minimal Reconciliation
Flat Structure
Sub-Architect
Case Application
Coach
Business Architecture Client Teams
Business Process Engineering a org-patterns such as Beedle’s, Hammer’s, Jacobson’s, Meszaros and Brown’s, and Taylor’s, etc.
Case Worker
Compreses the Process
Case Team
Software Development Organization
Case Manager
Application Development org patterns such as: Anderson’s, Appleton’s, Berczuk ’s, Coplien’s, Cockburn’s, Cunningham’s, Harrison’s, Risng’s, Delano’s, SCRUM and Whitenack’s, etc.
ApplicationFollowsProcess
Software Architecture
Reify the Process
SharedBusiness Objects
Released Baselines
Other Workflow Architecture patterns /constructs such as Beedle’s, Haugen’s, Jablonski and Bussler’s, and Meszaros and Brown’s, etc.
Other Software Architecture patterns such as GOF, POSA, etc.
Model Office

Figure 1. PattBPR - A pattern language for business process reengineering.

4. An Example

The following is a diagram of the superimposed architectures of three fictitious reengineered companies. The "short" reengineering story reads like so: The leader appoints the Business Architect. Together they discover the the CoreProcesses, find the business problems of the company through the CaseOfAction and they craft a process centric vision through the VisionStatement. This vision allows the architect to propose a NEW business architecture that contains an object oriented enterprise system architecture (Shared Business Objects). Then selected business processes are chosen to be "reengineered", and candidate applications are designed (ApplicationFollowsProcess). These applications are capable of managing workflow through an OOWMS (object oriented workflow management system), and their goal is to enable the Case Workers, Case Teams and Case Managers to be empowered by the Shared Business Objects with corporate information. This allows these process centric units to eliminate most delays, batches, queues, iterations, feedback loops, rework, unnecessary checks and controls and other business process ailments. Also by managing the business processes from within the software (ReifyTheProcess), the system architecture is capable of dramatically enhance the adaptability and reusability of business process steps. These time efficiencies allow the workers to grow and to enhance their overall work experience by having more time to "learn", "interact with others" and "invent" new business solutions.

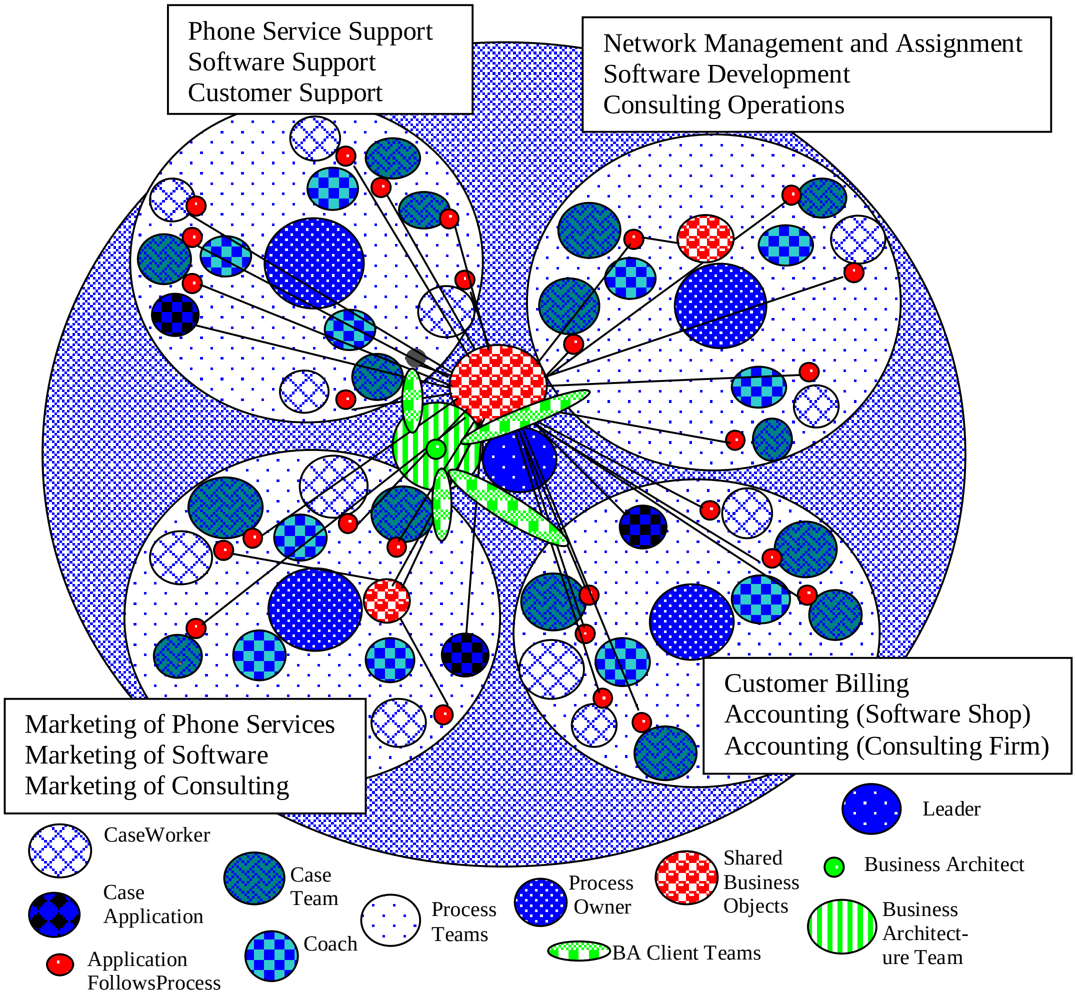

Figure 2. Three overlapped examples (phone company, software firm and consulting) of a reengineered Organizational Architecture.

5. The patterns in COOherentBPR

Leader(***)

Context
(From *Chaos***-** *the pattern of disorder)* An organization needs to make decisions at the business strategy level to adapt and survive within a constantly changing business environment.

Problem
Who should be the major sponsor and supporter of the reengineering effort? Who should make financial decisions that involve the major business functions of the organization? Who should communicate to the whole company what reengineering means to them?

Forces
A central point for decision making, global communications, resolution of issues, and vision at the highest level can be a bottleneck, but an incoherent direction can be disastrous.

Solution
A *Leader* needs to be carefully appointed (or emerges) in the organization to make these decisions. Good candidates for the *Leader* role are for example the CEO, the President, the CIO, the COO, the Director or VP of Strategy, Change Management or Operations.

The *Leader* provides many different things: on-going status of the organization, identification of problems (through a *Case of Action*), definition of a vision (through the Vision Statement), motivation, resolution of issues, communications, financial and moral support, a common purpose, and a common set of values and beliefs. He uses signals to communicate to the organization, symbols to show proof for his communications and systems to reinforce his messages. The *Leader* appoints the *Business Architect* and the *Process Owners* of the organization.

Related Patterns Patron from [Coplien95].

Resulting Context
(To *Core Processes*, *Business Architect***)** To make decisions at the business strategy level the fundamental business architecture of the organization must be determined.

Example
The Taco Bell subsidiary of PepsiCo was sick and getting sicker when John E. Martin was named CEO in 1983. Martin's problem was not convincing PepsiCo that the company had to reinvent itself, but that it had to reinvent itself in radical ways in a short period of time.

Charles Lee from GTE is probably the best example of a reengineering leader.

Figure 3. Charles Lee from GTE is the leader behind one of the best reengineering efforts ever to take place.

Known Uses
Robert L. Stark from Hallmark.
Norm Phelps from Capital Holding Corporation (DRG).

BusinessArchitect(***)

Alias
Reengineering Czar [Hammer93]

Context
(From *Leader***)** To make decisions at the business strategy level the fundamental business architecture of the organization must be determined.

Problem
The *Leader* is not someone that is an expert in process design, organizational structures, measurement and management systems or that has experience in change management issues. Who should be responsible for creating and documenting the existing and new business architectures? Who should be responsible to integrate all the sources of change into these models?

Forces
Having a single individual controlling the *architecture* of the organization can help in managing the conceptual integrity of the model.

Having a single individual controlling the architecture or the organization can bias its structure of the organization according to the expertise of the architect and it may be a limiting factor in the speed of change. Having multiple inputs from many individuals can compromise the conceptual integrity of the model but it would allow for greater speed and diversity.

Solution
The *Leader* appoints the *Business Architect* and gives him ownership of the *Enterprise Architecture* to ensure the its conceptual integrity. The *Business Architect* is an expert in business architectures that are process-centric. His primary responsibilities are:

1) Helping the *Process Owners* to run each of their processes,
2) Overseeing the activities of the *Business Architecture Team.*
3) Selecting, prioritizing, managing and releasing changes of the *Enterprise Architecture* to ensure that this is a coherent and congruent model of the organization,
4) Communicating to the *Leader* and the business *Sub-Architects*, and
5) Recommending resources to be assigned to the *Process Teams* and the *Business Architecture Team.*

The *Business Architect* must have the support of the *Leader* so that his business designs and decisions are implemented. The headquarters of the *Business Architect* is the *Business Engineering Process Team.*

Related Patterns - Architect Controls Product [Coplien95].

Resulting Context
(To *Sub-Architect, Business Architecture Team***)** The *Enterprise Architecture* has an owner **-** the *Business Architect,* but the amount of work is too much to handle by a single individual.

Example
Michael Beedle working at Nike Securities. Many other management consultants in BPR assignments worldwide.

Known Uses
GTE, Aethna, Ford, IBM Credit and most other reengineering efforts.

*Sub-Architect (*Pattlet)*

Context
(From *Business Architect***)** The *Enterprise Architecture* has an owner **-** the *Business Architect*, but the amount of work is too much to handle by a single individual.

Problem
What is the best way to delegate the responsibilities of the *Business Architect?*

Forces
The conceptual integrity of the *Enterprise Architecture* is managed by the *Business Architect,* but there is too much work for a single individual. Additional resources may help speed things up; however, this may be done at the risk of exposing the conceptual integrity of the system.

Solution
The *Business Architect* appoints *Sub-Architects* to delegate his responsibilities. There are at least two major categories of sub-architect: business sub-architects and system sub-architects. At least one business sub-architects is assigned to work with each of the *BusinessArchitectureClientTeams,* and he/she works very closely with the *ProcessOwner* from the business process being reengineered. Ideally the Business Architect and the System Architect are the same person, but in any event, the system sub-architects are typically arranged according to Conways' Law, and Organization Follows Market [Coplien95]. If the system is large enough each sub-architect is typically associated with a framework team (attributed to Ralph Johnson).

Related Patterns - Abstractionists [Booch94], Framework Team attributed to Ralph Johnson, Architect Controls Product [Coplien95], Dedicated Improvements Processors [Appleton97]

Resulting Context
(To) Strategies to delegate the work of the *Business Architect* have been defined, but the models of the *Enterprise Architecture* need to be specified as well as methods to design, document and implement these models.

*Business Architecture Team (*Pattlet)*

Context
(From *Business Architect***)** The *Enterprise Architecture* has an owner **-** the *Business Architect*, but the amount of work is too much to handle by a single individual.

Problem
What is the best way to delegate the responsibilities of the *Business Architect?*

Forces
Implementations of business designs made by the architect and its sub-architects may be delayed or not implemented at all because there is too much work to be done.

Additional resources may help speed things up; however, this may be done at the risk of exposing the conceptual integrity of the system.

Solution
Create a team of resources that implement and release new business solutions collectively. This team of resources is central to the organization and is formed by all of the sub-architects, the *BusinessArcyhitectureClientsTeams* resources, and the common resources among the client teams such as the integration and *ModelOffice* teams [Beedle97-2].

Related Patterns - Center PEG [Appleton97]

Resulting Context
(To) Strategies to delegate the work of the *Business Architect* have been defined, but the models of the *Enterprise Architecture* need to be specified as well as methods to design, document and implement these models.

EnterpriseArchitecture (*Pattlet)

Context
(From *Business Architect)* The business organization needs a "map" of the organization to understand its problems and to propose future solutions.

Problem
What is the best way to document the organization, its processes, its system architecture, and its production systems?

Forces
Having many models of the organization may lead to incongruent, incoherent or duplicate business models, and may impose a higher learning curve on the members of the organization.

Having a single model may require training for all the employees of the organization.

Solution
Therefore the *Business Architect* creates a single *Enterprise Architecture* that encompasses consistent models of the organization at different levels of abstraction that includes business processes and the enterprise system architecture.

Related Patterns - Process Follows Practice [Appleton97]

Resulting Context
(To *ScenariosDefineBusinessProcesses)* The models of the *Enterprise Architecture* are well specified and techniques to design and implement them are required.

ScenariosDefineBusinessProcesses (*Pattlet)

Context
(From *Business Architect)* Techniques to determine the existing and new business' and systems' organizations need to be determined.

Problem
What is the best way to document the requirements of a business process?

Forces
Not documenting the requirements makes it difficult to manage them.

Overly documentation the process may be time consuming and can result in paralysis by anaylisis.

Solution
Document the business process as a collection of business scenarios.

Related Patterns Business Use Cases [Jacobson95], story-boarding, scenario planning, Scenarios Define Problem [Coplien95]

Resulting Context
(To *ScenariosDefineBusinessProcesses)* The models of the *Enterprise Architecture* are well specified and techniques to design and implement them are required.

CoreProcesses(***)

Alias
Value Streams, Business Process Focus

Context
(From *Leader*) To make decisions at the business strategy level the fundamental business architecture of the organization must be determined.

Problem
What is the best way to model the organization and think about its problems and goals?

Forces
Requiring to list the processes of the organization is important because it makes the executives understand their operations in terms of value provided to its customers; however, processes are not the only aspect that is important to the organization. Other things like organizational structure, measurement and management systems, values and beliefs and/or financial indicators are also important.

Solution
The *Leader* creates short descriptions of the *Core Processes* or value streams of the organization which are key to understand what is important to the organization and its customers.

The *Core Processes* are bundles of related business activities that typically have external interactions or that are critical for the delivery of goods or services offered by the organization. Once created the *Core Processes* of the organization are distributed to all the members of the organization.

Resulting Context
(To CaseOfAction, ProcessOwners) Once the *Core Processes* of the organization are determined it is necessary to evaluate them.

Example
The *Core Processes* of Texas Instruments are:

1) The *Strategy Development* process converts market requirements into a business strategy, which identifies markets to be served and products and services to be offered.
2) The *Product Development* process uses this output as input in order to produce new products designs.
3) The *Customer Design and Support* process creates “qualified” designs as its outputs, using standard product designs and customer requirements as inputs.
4) The *Manufacturing Capability Development* process takes a strategy as its input and produces a factory as its output.
5) The *Customer Communications* process inputs are customer questions and inquiries; its outputs are heightened interest in TI products and consolidated responses to customers.
6) The *Order Fulfillment* converts an order request, a product design and factory into a product that is delivered into a customer’s hands.

Known Uses
IBM, GTE, Aethna Insurance and most other Fortune 1000 companies.

CaseOfAction(***)

Context
(From *Core Processes***)** The problems of the organization need to be stated.

Problem
What is the best way to evaluate and articulate the problems of the current processes of the organization?

Forces
Stating the problems of the organization can be done in many ways financially, socially, according to values, according to customer satisfactions; and they all skew the perceptions of what is wrong in particular way.

Stating the problems of the organization in "operations" terms through Core Processes, skews the view of problem analysis towards functional strategies.

Solution
The *Leader* creates the *Case of Action* using data from the operations of the organization. It summarizes the problems of the organization in a brief five to ten page document and it contains five important elements:

1) The Business Context. In this section what is happening, what is changing and what is important in the business environment is documented.
2) Marketplace Demands. Here the contextual conditions that impose requirements over the *Core Processes* are documented.
3) The Business Problem. In this section the problems found in the *Core Processes* of the organization are documented.
4) Diagnostics. This section explains why the *Core Processes* do not meet the expectation of the Marketplace Demands.
5) Costs of Inaction. Lastly, this section is a warning of what are the possible scenarios if these problems are not solved.

Once created the *Case Of Action* is distributed to all the members of the organization.

Resulting Context
(To *Vision Statement***)** The problems of the *Core Processes* are understood and it is necessary to define how these processes should look like in the future.

Example
Case of Action
Pharmaceutical Company

- We are disappointed by the length of time we require to develop and register new drugs in the United States and in major international markets.
- Our leading competitors achieve significantly shorter development cycles because3 they've established larger-scale, highly flexible, globally integrated R&D organization that operate with a uniform set of work practices and information systems.
- The competitive trend goes against our family of smaller, independent R&D organizations, which are housed in several decentralized operating companies around the world.
- We have strong competitive and economic incentives to move as quickly as possible toward a globally integrated model of operation. Each week we save in the development and registration process extend the commercial life of our patent protection and represents, at minimum and additional $1 million in annual pretax profit - for each drug in our portfolio.

Known Uses
IBM, GTE, Aethna Insurance and most other Fortune 1000 companies.

VisionStatement(***)

Context
(From *Case of Action***)** The problems of the *Core Processes* are understood and it is necessary to define how these processes should look like in the future.

Problem
What is the best way to document the goals of the organization?

Forces
Stating the future of the organization using the *Core Processes* results in visions of the organization that have a customer and operations focus; however, there are other important aspects of the organization that are de-emphasized such as revenue, profits, dividends for stockholders and/or profitability ratios.

Solution
A *Vision Statement* is created by the *Leader* to define the future of the organization. A *Vision Statement* is a brief - five to ten pages description of what the organization is to become. It outlines how the company is going to operate, and the kind of results that are expected. A *Vision Statement* must include the following three items:

- It focuses on Operations.
- It has Measurable Objectives and Metrics
- In some cases it changes the basis of competition for the industry.

Once created the *Vision Statement* is distributed to all the members of the organization.

Resulting Context
(To *Enterprise Architecture)* The vision of how the *Core Processes* are to look is understood and the business architecture that satisfies these goals needs to be created.

Example
Vision Statement
Federal Express

We will deliver the package by 10:30 the next morning.

Vision Statement
Pharmaceutical Company

- We are a worldwide leader in drug development.
- We have shortened drug development and registration by an average of six months.
- We are an acknowledged leader in the quality of registration submissions.
- We have maximized the profit potential of our development portfolio.
- We have created, across our operating companies, a worldwide R&D organization with management structures that let us mobilize our collective development resources responsively and flexibly.
- We have established uniform and more disciplined drug development planing, decision-making, and operational processes across all sites.
- We employ innovative technology-based tools to support our work and management practices at all levels and between all R&D sites.
- We have developed and implemented a common information technology architecture worldwide.

Known Uses
Federal Express, Hallmark, Ford, IBM, GTE, and most other Fortune 1000 companies.

LearningOrganization (***)

Context
(From *Leader)* Business organizations undergoing change.

Problem
Competition requires adaptation, and adaptation often requires learning new things. The organization educational needs are much greater than any "training department" can provide. How should the organization be educated as a whole?

Forces
Training organizations can provide basic knowledge and encourage people to learn on their own but often lack the sufficient depth to make everyone excel on their job functions.

Having everyone involved in training activities all the time can mean less productive time to work.

The ability for the organization and individuals can be taught but it takes time to develop.

Avoiding training needs in an organization for an extensive period of time could result in a “passive corporate culture” that stands at the mercy of market competitors.

Solution
Provide very many ways to educate the organization in order to accomplish its VisionStatement.

For example, if BPR is the vision of an organization, BPR and its new vocabulary (this pattern language), must be taught to everyone in the organization. This will help the organization create a new culture based on the new concepts introduced (such as Process Owners, Case Teams and so forth). Also provide seminars and focus groups where the CoreProcesses, the CaseOfAction, and the VisionStatement of the organization are explained and discussed. This effort involves all the members of the company. For reengineering to work, reengineering must be understood by everyone.

However, to excel at their particular processes, everyone must be provided with specialized education particular to their job functions in order for them to excel. For example:
- the marketing process needs to learn about its competitors,
- the operations department needs to provide better information systems,
- the manufacturing process needs to be aware of new technologies available to build products,
- and the executives need to know of ingenuous ways to change the culture of their companies, etc.

Many different ways to help the organization learn can be provided through:

- Corporate seminars
- Internet access and specialized INTRANET sites
- Magazine and periodicals subscriptions
- CBT (computer based training)
- Formal education programs (such as tuition re-inbursement, grants and fellowships)
- Email broadcasting or mailing lists
- Corporate mail
- Study or focus groups that introspect about the organization
- Brown bag lunches
- Formal mentoring programs
- Corporate libraries, etc.

A key element of the Learning Organization is that job preparation changes from training to education. Publish information about the organization to its employees so that they can understand what the organization, its process and its needs are.

Related Patterns - Virtual Forum [Appleton97]

Resulting Context
(To *LearningOrganization***)** Business organizations in a competitive market.

Example

Some divisions of Motorola allow their employees to use up to 20% of their time in education.

Known Uses

Motorola; AT&T; IBM

EncourageProductiveValues (***)

Context

(From *Productive Values***)** The context is in large organizations where protective values are the norm.

Problem

For reengineering to work, the correct new culture needs to be breed. Employees do not see their significance to the customer because they feel they are "too small" to make a difference. What is the best way to institutionalize customer focus, value adding, and mutual caring for other employees?

Forces

Employees are apprehensive about sharing their work and their knowledge because the fear being replaced or found obsolete if they reveal their secrets.
Competitive advantage at the individual level correlates with the information that one knows and holds.

Sharing information among employees leads to better understanding of the common goals.

Information about someone else's work may be used against others to point out defects in their work, replace workers or reduce their amount.

Solution

Change the values of your organization from protective to productive. Encourage, compensate and facilitate cross training of employees and ensure employees of their value to the company. Avoid massive replacements, or lay-offs in areas of the company where information sharing is encouraged. Provide constant nurturing on the employees' mutual trust, customer focus and provide facilities to enhance the employee's well being.

Resulting Context

(To *Productive Values (the pattern is recursive and never ending)***)** The resulting organization has their priorities aligned on their customers and their personal growth.

Example

Encourage values like:

- Our customers always pay our salaries.
- All of us have equally important jobs.
- We get paid for adding value to the organization and its processes.
- We belong to a team. We fail or succeed together.
- We must constantly learn and grow as individuals.

Provide facilities where employees may learn and interact with each other like study groups, sport activities, and discussion groups.

Provide constant information about the status of the organization CaseOfAction, organization, vision and business architecture of the organization to increase the organization's self-awareness.

Known Uses

Xerox, AT&T, IBM, GTE; Ford.

ProcessOwners(***)

Context
(From *Case of Action)* The problems of the *Core Processes* are understood and it is necessary to define how these processes should look like in the future.

Problem
What is the best way to get accountability out of a process? Functional departments create assembly lines and the ownership of the process and its results are always in question.

Forces
Hiring, managing and building functional departments is easy; however, they create assembly line processes without encouraging ownership of the results for the customers. Allocating ownership to processes has more customer focus, because the products for a client have more accountability.

Solution
The *Leader* assigns *Process Owners* corresponding to the *Core Processes* of the organization. Good candidates to become *Process Owners* are old functional managers that understand the importance of the new focus on lateral management.

These *Process Owners* are sometimes already realizing most of the work in their functional management positions but haven't been officially assigned the ownership of the process. Hiring external *Process Owners* will send a strong signal to the organization in case there is opposition by the current functional managers.

In some cases, when the process is very broken *Process Owners* play the role of "lateral" managers and coordinate activities across large functional units, at least temporarily while *Process Teams* are created.

One could say that almost anything could be a CoreProcess; however the key is to concentrate in complete business activities that produce outcomes for customers.

Resulting Context
(To *Strategic Direction Initiative)* Structures, tactics and priorities to redesign the *Core Process* and the system architecture need to be chosen.

Example
IBM Credit's Wayne Hooever, Regis Filtz, Bell Atantic's CAS Leader.

Known Uses
IBM Credit, GTE, Aethna Life, TI Semiconductor Group

*StrategicDirectionInitiative (*Pattlet)*

Context
(From *Process Owners***)** Structures, tactics and priorities to redesign the *Core Process* and the system architecture need to be chosen.

Problem
Knowing that there are problems in the *Core Processes* through the *Case of Action* and goals as stated in the *Vision Statement* is not enough to determine which processes can be or should be redesigned. What is the best way to determine reengineering opportunities? What processes should be redesigned first? How do we determine the order of processes that need to be redesigned?

Forces
Not acting quickly on a process that is broken may incur high costs, loosing market share, or profits.

Redesigning an important process is risky and may be betting the company in the effort.

Solution
Evaluate each of the *Core Processes* of the organization in a *Strategic Direction Initiative* to identify their dysfunction, importance, and feasibility to be redesigned. The *Process Owners*, the *Business Architect* and the *Leader* are the main active participants of this effort.

Resulting Context
(To *Business Architecture Team)* The order in which the *Core Processes* can be redesigned has been determined and ways to implement this prioritized list are required.

BusinessArchitectureClientTeam (***)

Alias
Reengineering Team

Context
(From *Strategic Direction Initiative)* The order in which the *Core Processes* can be redesigned has been determined and ways to implement this prioritized list are required.

Problem
Who does reengineering? Who releases enabled applications to production and trains the new CaseWorker s, CaseTeam s or CaseManager s into their new roles? Who advises human resources of the new management and measurements systems? Who resolves issues with released Process Teams?

Forces
Having the whole organization involved in process redesign is ideal **-** after all they know what the business is all about;

The members of the organization may not fully understand the principles of business design and it may also be distracting and time demanding for them.

Designing the business processes and the software to enable the business processes may be hard to integrate if done separtely.

Combining business design and software development resources to work together is not and easy task because of the wide gap in their backgrounds.

Having separate efforts to do business design and application design almost always results in disasters, either building the "wrong" thing, or being delayed by mutual dependencies that are often irreconcilable.

Solution
To balance these forces, create a *BusinessArchitectureClientTeam* every time that a process is being redesigned. This team is responsible for ALL stages of the process from the analysis and design of the business to the implementation and delivery of the software that enables the business process. This team is composed with most of the traditional software development groups but it is expanded to include Business Analysis and Business Design Stage stages before the traditional software development stages, and consequently it also includes a *ModelOffice* stage where the business processes are tested.

The *BusinessArchitectureClientTeam* seeks to find the way people are working in a given process, to understand or abstract the business requirements of that process, and to eventually redesign the process into a more effective way of conducting. The *BusinessArchitectureClientTeam* for a process is often composed of insiders (members of the process being redesigned), and outsiders (some of which may be members of other processes, consultants, information systems specialists or HR specialists.).

The recruits of this team need to be carefully selected and balanced. It is a requirements that at least two members are insiders with extensive business knowledge, and that at least one member is part of the system requirements group. Another reason to have a *BusinessArchitectureClientTeam,* is that the original organizers of the may have been dismantled and there may not be anyone responsible for updating or clarifying the model. This is the typical result of assignments of purely management consulting companies such as McKinsey and Co, that are too expensive to keep around until the "process delivery" date.

RelatedPatterns Improvement Actions Teams, PIT also practices [Appleton97].

Resulting Context
(To *Process Teams, Queue of Work***)** The structure and ownership of the team that does reengineering has been assigned and tactics to redesign and evolve the organization's architecture are required.

Example
An example of such a *BusinessArchitectureClientTeam* is the one used by Bell Atlantic to reengineer their CAS service.

Known Uses
David Taylor uses similar teams in his Convergent Engineering practice. James Martin uses teams of Enterprise Engineers in his management consulting practice, etc.

OrganizationFollowsProcesses (***)

Context
(From *BusinessArchitectureClientTeam)* The structure and ownership of the team that does reengineering has been assigned and tactics to redesign the organization, its processes, its enabling applications and its system architecture are required.

Problem
What is the best way to implement an organization that is aligned according to its *Core Processes*?

Forces
People are educated with a specialty.

Functional structures create queues, handoffs, batches and feedback loops across different organizations.

Creating single units that own the processes from beginning to end have greater customer focus and accountability.

Owning a whole business process implies greater responsibilities.

Solution
Assign a *ProcessTeam* s to the *ProcessOwners* that are responsible for the *CoreProcesses*. This transitions the organization from functional departments to a process centric business architecture. Typically this transition is accomplished one process at a time but based on the design of an *EnterpriseArchitecture*.

ProcessTeam are composed of a mixture of *CoachCoach* s, *CaseApplication* s, *CaseWorker* s, *CaseTeam* s and/or *CaseManager* s. These constructs provide for the implementation of single instances of the process. For example, in a manufacturing environment the process may be conducted with a single *CaseTeam*, with a combination of multiple instances of connected *CaseTeam* s, with a single *CaseWorker*, or with Sub-Process owners.

Resulting Context
(To *FlatReportingStructure, NaturalOrderPlan, MultipleProcessVersions, WorkAllocation, MinimalChecksAndControls, MinimalReconciliation***)** Structures to and tactics to implement the *Process Teams* need to be chosen.

Example
TI assigns process teams to their order fulfillment process.

Known Uses
IBM Credit, GTE, Aethna Life, TI Semiconductor Group.

FlatReportingStructure (*Pattlet)

Context
(From *ProcessTeam*) Structures to and tactics to implement the *Process Teams* need to be chosen.

Problem
What is the best way to structure reporting hierarchies?

Forces
Organizations with long chains of command have fewer responsibilities for the participants in the management chain. In theory this allows managers to focus more on a given responsibility.

Long reporting chains of command are slow to react and adapt to new business conditions and act as "Broken telephones" that convey incomplete or inaccurate information to upper level management layers, and they de-emphasize the ownership and involvement of the people that perform the work of the process;

Short chain of command management structures impose a higher responsibility and standards for the managers of the organization.

Solution
Create supporting organizations for *ProcessTeam s* that have flat structures.

Resulting Context
(To *CoachCoach*) Structures to implement flat organizations need to be chosen.

CoachCoach (*Pattlet)

Context
(**From** *FlatReportingStructure*) Structures to implement flat organizations need to be chosen.

Problem
What is the best way to supervise, monitor, control, assign and check work, workers and work schedules?

Forces
Self-directed teams and self-directed individuals are rarely a stable steady state in a business organization.

High support and high direction requires more time from managers, but people often need help to accomplish their work.

Supervision is sometimes required to check the quality of work and provide facilitation, conflict resolution or career development; however, many interruptions involving the workers may slow down business activities and increase costs.

Solution
Change the role of managers from supervisors to coaches and from scorekeepers to leaders. The CoachCoach chooses the correct balance of direction given to his constituents. He may in fact check the quality of work, provide facilitation, conflict resolution or career development; but he encourages and teaches his constituents to be self-directed in their work activities.

Resulting Context

(**To** CompressTheProcessHorizontally)Characteristics of the structures that perform the actual work need to be determined.

CompressTheProcess(***)

Context

(**From** *BusinessArchitectureClient Team*) Characteristics of the structures that perform the actual work need to be determined.

Problem

How should the structures that provide work be architected?

Forces

Creating organizations of “functional specialists" that pass work among each other was the philosophy of the Division of Work that came out of Adam Smith’s The Wealth of Nations. However, these organizations discourage the view of the customer and create systemic problems such as queues, batches, feedback loops and delays.

People in general tend to be specialists, so hiring and training a functional organization is easy.   Instead training someone to accomplish the work of an entire business process could result in unrealistic expectations.

Solution

Combine several jobs into one whenever possible using the CaseWorker, CaseTeam, CaseManager and CaseApplication organizational patterns. These are organizational structures that assign a single organizational construct to be responsible for a whole process instance. A consequence of this is that jobs change from single task to multi-dimensional work, but in doing so many "systemic" problems such as queues, batches, feedback loops and delays are eliminated.

CaseWorker s, CaseTeam s, CaseManager s and CaseApplication s, always CompressTheProcessHorizontally.

Related Patterns - Developer Controls Process [Coplien95]

Resulting Context

(To *Case Application, Case Worker, Case Team, Case Manager*) The structures that perform the work compressing the business processes need to be determined.

Known Uses

Developer Controls Process [Coplien95], Tall-fat men [Booch94], [Hammer93], [Senge90], [Hammer96]

CaseApplication(***)

Context

(**From** *CompressTheProcessHorizontally)* A process where no workers are allowed because of financial or time constraints or because it is more convenient for the customer needs to be implemented.

Problem

What is the best way to implement a process with no people involved?

Forces

Involving people in a business activity has the advantage of creating a more “personable” business environment; however, people are inherently inefficient to communicate with each other or to make computations.

Machines are now cheap when compared to the services provided by people.

Machines can automate many tedious tasks; however, machines not always make the right choices, especially when dealing with new contexts.

Solution

Create an application that takes care of the whole business process.

Resulting Context

(**To** *ApplicationFollowsProcess,* **Alternate** *CaseApplication, CaseWorker, CaseTeam, CaseManager*) Specific tactics to enable the business processes need to be determined.

Example

Citibank ATMs' network spread throughout the world.

Known Uses

IVR (interactive voice response) units that carry the work for many banking, insurance and customer service business processes.

CaseWorker(***)

Context

(**From** *Compress the Process Horizontally*) The structures that perform the work compressing the business processes need to be determined.

Problem

The realization of a business process may take several forms. Are there organizational structures that enhance productivity, cooperation, and customer focus but at the same time induce the growth of the company and its employees?

Assembly lines create "systemic" problems such as queues, batches, feedback loops, delays, high inventories and high ratios of checking and control activities vs. value added activities. High "inventory" in creative or intensive human labor environments translates in large amount of idle human resources. What is the best way to organize resources that are capable of fulfilling all the needs of a given process?

Forces

Involving people in a business activity has the advantage of creating a more "personable" business environment; however, people are inherently inefficient to communicate with each other or to make computations.

Machines are now cheap when compared to the services provided by people.

Machines can automate many tedious tasks; however, machines not always make the right choices, especially when dealing with new contexts.

Solution

Assign a CaseWorker aided with the proper enabling application (ApplicationFollowsProcess) to fulfill the needs of an entire business activity (process). This solution to the problem of compressing the process has a more "personlized touch" than a CaseApplication. So the choice may be made on the need to be "personable" and the feasibility of the implementation.

The CaseWorker should be trained to become a generalist in the business domain in order to respond to all the business requests assigned to him. There may be many types of CaseWorker s ranging from the very directed, inflexible and guided jobs of telemarketing and customer service; to the very creative engineer, researcher and software designer/programmer positions.

The CaseWorker will have a great deal to learn to become better at what they do. Also they may find very fulfilling to train or mentor other CaseWorker s into their corresponding positions. Also the CaseWorker get a large sense of accomplishment since they are responsible for a whole business activity. Since they will have a direct impact to the bottom line, it is often found that they are remunerated accordingly; not only in good salaries, but in performance bonuses as well.

They also form an integral part in designing new or alternate ways to perform the business process. And as they grow within the company, they can leverage extensively their hands-on business experience and translate their business knowledge into their most valued company's asset.

A definite characteristic of a CaseWorker is that his job will change from single task to multi-dimensional work.

Resulting Context
(**To** *Enabling Application* **Alternate** *Case Application, Case Worker, Case Team, Case Manager*) Specific tactics to enable the business processes need to be determined.

Example
Ameritech Customer Service Representative in the customer service department

Known Uses
US Spring, MCI, AT&T, American Airlines and most other service oriented organizations.

CaseTeam(***)

Context
(**From** *Compress the Process Horizontally*) A process instance team has to be instantiated and many functional resources exist with some that are generalists.

Problem
What is the best way to organize cross-functional resources that are capable of fulfilling all the needs of a given process?

Forces
Hiring and managing "specialists" is relatively easy and it is well understood; however, this leads to assembly lines with "systemic" problems, and to unhappy employees whose potential and growth is constrained by their own specialization.

Hiring, mentoring and building teams of cross-functional resources is challenging; however, this leads to efficient processes and for desirable interactions among the members of the team that allow them to grow as individuals.

Solution
Create a *Case Team* composed of cross-functional resources that can fulfill the needs of the whole process. The "mix" of a *Case Team* needs to be carefully evaluated in order to make the team whole.

The members in a *Case Team* have "fuzzy" ownerships and responsibilities regarding the execution of tasks and artifacts within the process. As a consequence of their constant interaction, the members of a *Case Team* constantly learn from each other and have a better chance to enjoy a dynamic working environment. This gives them a "growth" path by learning from more experienced or more knowledgeable members in the *Case Team.*

In some cases, the members of a CaseTeam are also capable to reorganize themselves and adapt to multiple business situations and environments and therefore are a good tool to harness "uncontrollable" processes. Their diversity gives them a chance to use their individual strengths in different situations, without being limited or confined to a specific role forever.

Finally, it also allows them to exercise their leadership abilities periodically without being exposed to larger responsibilities and the pressure of an assigned position of responsibility. This gives them a chance to grow as leaders and assume greater increased responsibilities as time evolves. *Case Teams* always have *Enabling Applications* to help them accomplish their duties.

The *Case Team* is bounded by the task that it is assigned to, it emphasizes "fuzzy" internal ownership, but "strong" external ownership, it uses a SCRUM like approach to develop deliverables [Sutherland96] and it works on the basis of *Encourage Productive Values* among its members.

Resulting Context
(**To** *Enabling Application* **Alternate** Case Application, Case Worker, Case Team, Case Manager) Specific tactics to enable the business processes need to be determined.

Example
Business Environment
IBM Credit's cross-functional teams.

Application Development
Replacing the Developer role in *Developer Controls Process* with a cross-functional team of developers in a three-tiered client-server architecture development effort using *Form Follows Function*.

Known Uses
Ford, GTE, Aethna, Hallmark, PepsiCo and most other Global 2000 companies.

*CaseManager(*Pattlet)*

Context
(**From** *CompresTheProcessHorizontally*) There are many available "functional workers" but there are pressing business need to implement a "process solution".

Problem
How can one implement a process solution with functional workers?

Forces
Creating a process as an assembly line with the functional workers available does not have customer focus and can lead to business problems.

Mentoring and building cross-functional teams and resources is challenging will take a longer time than the time that is allowed.

Solution
Create a CaseManager role that "front-ends" a larger more complex organization underneath. In this case the CaseManager is the "face" to the Customer and he is accountable for the process.

Resulting Context
(**To** *EnablingApplication* **Alternate** CaseApplication, CaseWorker, CaseTeam, CaseManager) The process has customer focus but there might be process problems underneath that may surface from time to time.

ApplicationFollowsProcess (***)

Alias
Enabling Applications, Process Wizards

Context
(**From** *CaseApplication, CaseWorker, CaseTeam, CaseManager*) Specific tactics to enable the business processes need to be determined.

Problem
What is the best way to enable reengineering construct such as CaseWorker s, CaseTeam s or CaseManager s?

Forces
Creating applications to enable business processes requires many skills, it has a high cost in terms of software and hardware, and it may require interactions between the people that run the business processes and the ones that write the software.

However, having enabling applications that speed up the processes and help accomplish the work faster may impact dramatically the bottom line of the business as a whole.

Solution
Create enabling applications that "follow" the business processes. The applications not only should display “data” that is important to the users but they need to "walk" them through the process instances.

Enabling applications are created through the BusinessEngineering life-cycle, which includes: Business Requirements, Business Design, Business Evolution and Business Evolution.

It is best if these applications are created from a common base - an Enterprise System Architecture. That way, the business logic is reused in creating multiple applications. It is best to choose a technology that directly supports reuse in order to facilitate the delivery of applications, such as object-oriented languages and architectures.

In creating the enabling applications, it important to identify the “hot spots” - places where the business process may change, in order to provide for appropriate hooks for later extensions. This pattern naturally leads to ReifyTheProcess.

Resulting Context
(**To** *ReifyTheProcess*) Specific tactics to implement the design of the EnablingApplication need to be determined.

Example
Bell Atlantic implemented enabling applications that allow them to reduce cycle time in their CAS process by several orders of magnitude.

Known Uses
IBM Credit, Hallmark, Bank of America

ReifyTheProcess(***)

Alias
OO Workflow Manager

Context
(**From** *ApplicationsFollowsProcess*) Specific tactics to implement the design of the EnablingApplication need to be determined.

Problem
What is the best way to architect enabling applications in a reengineered environment?

Forces
Software can be easily written to present “data” to the user when he/she needs it; however, that doesn’t make the programs compatible with the process they are enabling or for that matter more adaptable. Software that has “reified” versions of the process are harder to write because a more detail accounting of how that software interacts with the business processes is needed; however, they become more adaptable and accept variations on the process they enable.

Solution
Create an application that contains a “reified” version of the process within the software. That way the process can be managed from within the software, offering the possibility of running multiple versions of it, creating variations on the domain objects that run with it, or saving it’s state to be completed at a later time or by a different person or group.

The reification of processes means literally creating an object that represents the process, that way the process steps can also be reified to offer the possibility of applying polymorphic interfaces and design patterns to them. This adaptability is key in the rapid adaptation of the organization to the business environment, and therefore is of “critical competitive advantage” to use this technique whenever possible.

Resulting Context
(**To** (Other components of the OO workflow manager))The architecture to implement the enabling applications for a business process has been identified and it is necessary to identify presentations, process objects and domain objects that are required for the *Enabling Application*.

Example
William M. Mercer pension calculation system at the MAC (Mercer Administration Center).

Known Uses
Hewitt Associates, Action Workflow, American Airlines reservation systems

SharedBusinessObjects (***)

Context
(**From** *Reify the Process*) Specific tactics to implement the design of the enabling application need to be determined.

Problem
What is the best way to architect the enterprise system architecture in a reengineered environment?

Forces
Software can be written to access data in persistent stores but it is very inefficient and it is also has concurrency problems.

Software can be written to work from "caches" or "in-memory" databases, but it is very resource demanding.

Solution
Create a distributed object oriented architecture that caches business objects that are used in most business transactions. They may be for example, highly used static information and information that only changes according to a given business cycle (hourly, daily, weekly, etc.).

The advantages of caching these business objects typically reflect many orders of magnitude in the performance of the programs and in the reduction of code complexity. From the reengineering perspective, information means "empowerment", so the more information the worker is able to manipulate the better more adaptive and effective applications the process workers will have.

Resulting Context
(**To** Frameworks of Business Objects, other system architecture patterns like POSA[POSA95], GOF [GOF95], PLOP1+2+3+4 books, etc.) The architecture to implement the enabling applications for a business process has been identified and it is necessary to identify presentations, process objects and domain objects that are required for the *Enabling Application.*

Example
All applications created with CORBA architectures. All applications created with Persistence "live object cache" . William M. Mercer pension calculation system at the MAC (Mercer Administration Center).

Known Uses
Nike Securities, Hewitt Associates, Motorola Iridium, Boeing BPR efforts (and most other OO architectures).

*MultipleProcessVersions(*Pattlet)*

Context
(**From** *OrganizationFollowsProcess*) Two very similar business processes are required to be implemented with significant differences on the kind of customer, the amount of work to be done, or the number of approvals needed to accomplish it.

Problem
A process that is made to fit all possibilities imposes impractical financial and time constraints for whoever uses it. What is the best way to please multiple customers with slightly different requirements within the same process?

Forces
A well understood process provides with a single way to get things done and potential savings from the reuse of its supporting organization.

Different customers may find unrealistic expectations using the same process according to their needs.

Allowing different versions of the same process sometimes means overhead.

Solution
Create different versions of a process if there are exceptional business scenarios that are important for a customer.

Resulting Context
(**To** *MultipleProcessVersions, NaturalOrderPlan,* IntegrateCustomersAndSuppliers, *MinimalChecksAndControls, MinimalReconcilliation*) Different customers are satisfied using different processes that might re-use some portions of each other.

IntegrateCustomersAndSuppliers(***)

Context
(**From** *ProcessTeam*) Structures to and tactics to implement the *ProcessTeams* need to be chosen .

Problem
Work is not always assigned to right place, done at the right time and assigned to correct people. Where should one allocate work reduce cycle time and increase profitability?

Forces
Work can be assigned to the incorrect place and the process still may work; however, inefficiencies may accumulate or prevent the organization to grow.

Assigning work to the correct place, time or group is not always easy; however, this may have long term implications for the growth of the organization.

Solution

Go beyond the historical, organizational, financial or political barriers and allocate work accordingly. In some cases work that traditionally belongs to the organization is transferred to the customer, the supplier or another internal department. This situation may be the result of misfits between the logical and physical business architectures, but it can also be the result of balancing the strengths and weaknesses of the parties involved to their mutual benefit.

Resulting Context

(**To** *MultipleProcessVersions, NaturalOrderPlan,* IntegrateCustomersAndSuppliers, *MinimalChecksAndControls, MinimalReconciliation*) Structures to and tactics to implement the *ProcessTeams* need to be chosen.

Example

Navistar has shifted some of its work back to its suppliers. Instead of managing its own warehouse inventory of tires to be installed on the trucks it manufactures, it delegates Good Year, its supplier to do that, because they have a much better inventory management methods.

Walmart used to place orders from customers. Now customers check their inventories at Walmart and they stock their own inventories placing orders to themselves.

Known Uses

Navistar/Good Year, Wal-Mart/Suppliers, Ford/Suppliers.

ProcessWithMinimalChecksAndControls(***)

Context

(**From** *Process Team*) Structures to and tactics to implement the *Process Teams* need to be chosen.

Problem

A large amount of checks and controls slows down processes and increase overhead for an organization. How should one ensure quality in a process?

Forces

Checks are needed to ensure quality. Having no checks or controls at all leads to poor quality products and services and unmanageable organizations.

Too many checks and controls reduce productivity.

Solution

Minimize in as much as possible the checks and controls imposed into a process. First make an inventory of all the checks and controls within the existing process. We also recall, that the CoreProcesess and EnterpriseArchitecture gave us the understanding of the processes.

Then, review the need, the format, the grouping and the order, for each of the checks and controls in the process. Keep only the checks and controls that are absolutely necessary and change the order in which they are applied to match the natural breakdown of work. In many cases, check and controls can be deferred and automated, in some cases they may be "aggregated", in some other cases they may triggered by a "measured level", etc. This pattern reduces both cost and cycle-time, while it simplifying the process by minimizing work interruptions. The solution is understood from the point of view of system dynamics, because it eliminates delays, feedback or feed-forward loops, queues, batches, etc.

Resulting Context

(**To** *MultipleProcessVersions, NaturalOrderPlan,* IntegrateCustomersAndSuppliers, *MinimalChecksAndControls, MinimalReconcilliation*) Structures to and tactics to implement the *Process Teams* need to be chosen.

Example
Business
Purchasing departments always check for approval signatures. However, sometimes the “checking cost” of an approval is larger than the cost of purchased item. Instead, purchasing approvals may have “approval ceilings”, in order to avoid costly approvals.

Capital Holding used to check for every detail before paying an insurance claim. After revising the assumptions it was found that for small claims it was better to pay it first because the costs in revising every detail beforehand were too costly.

Software Development
Having unit tests, design reviews and code inspections for every component such as a method or a class imposes very high costs in the software development process. This is especially true given the nature of software development, where many things may keep “changing” for extended periods of time. Instead, it is better to educate the developers how to correct 90% of the problems, or to use quality-checking tools; and then “aggregate” and automate the tests in integration or system tests.

Known Uses
Insurance Companies (small claims), Banks (cash station and credit card small claims fraud), etc.

ProcessWithMinimalReconciliation(***)

Alias
Minimal Artifacts

Context
(**From** *Process Team*) Structures to and tactics to implement the *Process Teams* need to be chosen.

Problem
Reconciliation across many artifacts/documents of related information is difficult and requires large amounts of work. What is the best way to document work related artifacts, or business activities?

Forces
Documenting information more than once requires reconciliation but it may serve the needs of different audiences.

Documenting information once implies minimal reconciliation.

Solution
Minimize the number of by-products or artifacts that a process uses. The less documents of artifacts there are to reconcile the less amount of overhead work there will be. This solves the systemic problems of rework that imply delays, feedback loops, queues and batches of work to be reconciliated. To implement it, an analysis of the work products needs to be done, and areas of duplication need to be identified. Once this is done, careful redesign of process documentation can be done avoiding duplication.

Resulting Context
(**To** MultipleProcessVersions, NaturalOrderPlan, IntegrateCustomersAndSuppliers, MinimalChecksAndControls, MinimalReconcilliation) Structures to and tactics to implement the *Process Teams* need to be chosen.

Example
Business
Ford Motor Co. used to have three contact points where invoicing duplication was duplicated: 1) the purchasing department through the **purchase order**, 2) the receiving dock through the **receiving paperwork**, and 3) accounts payable through the **invoice**. Instead it eliminated the invoice creating opportunities to reduce costly overheads.

Software Development
In many software development shops there are as many as 150 inter-related documents related to a single project. This creates the need for unnecessary reconciliation. Instead, successful software shops keep a minimal set of documentation; for example, the requirements document, the architecture document and design documents, the code, the testing plan, and the project plan.

Known Uses
Ford

ModelOffice(***)

Alias
Business Process Test

Context
(From *ReleasedBaselines)* . ReleasedBaselines containing business functionality and the enabling application (ApplicationsFollowsProcess) have been released and methods to validate and test this functionality are required.

Problem
What is the best way to test business processes and their respective applications?

Forces
Releasing the business processes without testing may lead to business problems.

Testing the business processes is difficult because the business environment needs to be simulated.

Solution
Recreate the business environment in a smaller scale so that the business processes can be enacted and tested before release. This activity serves many purposes:

- Testing the designs of the business processes and their corresponding EnablingApplication s.
- Educating the employees of the released business processes in their respective new duties and responsibilities.
- Discovering new ways of doing things in practice. Etc.

Compare to Engage QA [Coplien95].

Resulting Context
(To Working Business Processes **-** not a pattern**)** Techniques for growing and building business organizations needed to be determined.

Example
Bell Atlantic had an ongoing ModelOffice that allowed business processes to be tested and released.

Known Uses
Bell Atlantic, IBM Credit, GTE

6. Acknowledgments

I would like to thank Brad Appleton, Steven Berczuk, Alistair Cockburn, James Coplien, Martine Devos, Robert Hanmer, Robert Haugen, Ralph Johnson, Wolfgang Keller, Urvashi Kaul, Tim Ottinger, Scott Preece, Shalom Reich, Linda Rising, Ward Cunningham and many other participants from the org-patterns list that contributed with many comments and suggestions.

7. References

[Alexander78] C. Alexander, *A Pattern Language*, Oxford University Press, New York, 1978.

[Alexander79] C. Alexander. *The Timeless Way of Building,* Oxford University Press, New York, 1979.

[Alexander97] C. Alexander. *The Nature of Order,* Oxford University Press, New York, 1997.

[Appleton97] B. Appleton, Patterns for Conducting Process Improvement, accepted to the PLOP97 conference, Allerton Park, Illinois, 1997.

[Beedle95] M. Beedle. *Object Based Reengineering*. Object Magazine 4(2), 1995.

[Beedle97] M. Beedle. *Pattern Based Reengineering*. Object Magazine, January 1997.

[Beedle97] M. Beedle. *A "light" distributed OO Workflow Management System for the creation of OO Enterprise System Architectures in BPR environments*, sumitted to OOPSLA's Business Objects III workshop, October 1997.

[Beedle97] M. Beedle. *Reengineering the Application Development Process*, SIGS, New York, 1997 - forthcoming.

[Blanchard90] K. Blanchard et. al., *The One Minute Manager Builds High Performance Teams,* William Morrow, New York, 1990.

[Boehm81], B. Boehm, *Software Engineering Economics*, Prentice Hall, Upper Saddle River (NJ), 1981.

[Bohm80] D. Bohm, *Wholeness and the Implicate Order*, Ark Paperbacks, London 1908.

[Booch94], G. Booch, *Object Oriented Analysis and Design,* Benjamin Cummings, New York, 1994.

[Booch95] G. Booch, *Object Solutions*. Addison and Wesley, Reading, 1995.

[Brooks95] F. Brooks, *The mythical man-month,* Addison and Wesley, Reading, 1995.

[Carr95] D. Carr, H. Johansson, *Best Practices in Reengineering*, McGraw Hill, New York, 1995.

[Chomsky79] N. Chomsky, *Rules and Representations*, Columbia University Press, New York, 1979.

[Cook96], M. Cook, *Building Enterprise Information Architectures*, Prentice Hall, New Jersey, 1996.

[Coplien95] J. Coplien and D. Schmidt, *Pattern Languages of Program Design* (*A Generative Development-Process Pattern Language),* Addison and Wesley, Reading, 1995.

[Coplien96] J. Vlissides, J. Coplien and N. Kerth, *Pattern Languages of Program Design - 2*, Addison and Wesley, Reading, 1996.

[Coplien96] J. Coplien, *Software Patterns*, SIGS, New York, 1996.

[Coplien97] J. Coplien, *"Re: pidgin, creolized"*, mail message from the patterns-discussion held at UIUC.

[Covey91] S. Covey, *The 7 Habits of Highly Effective People*,

[Cunningham95] W. Cunningham, *Episodes - A pattern language of competitive Development*, Review Draft of August 6, 1995.

[Davenport93], T. Davenport, *Process Innovation - Reengineering Work Through Information Technology*, Harvard Business School Press, 1993.

[DeMarco] T. De Marco, *Peopleware*,

[Drucker97] F. Hesselbein, M. Goldsmith, R. Beckhard, *The Organization of the Future*, Jossey Bass, San Francisco, 1997.

[GOF95] E. Gamma et al. *Design Patterns - Elements of Reusable Object Oriented Software*. Reading, MA. Addison-Wesley, 1995.

[Gabriel96] R. P. Gabriel, *Patterns of Software*, Oxford University Press, New York, 1996.

[Graham94] I. Graham, *Migrating to Object Technology,* Addison and Wesley, Workingham, England, 1994.

[Fowler96] M. Fowler, *Analysis Patterns*, Reading, MA. Addison-Wesley, 1996.

[Hammer93] M. Hammer and J. Champy, *Reengineering the Corporation: A Manifesto for Business Revolution*. Harper Collins, New York, 1993.

[Hammer95] M. Hammer and S. Stanton, *The Reengineering Revolution*, Harper Collins, New York, 1995.

[Hammer96] M. Hammer, *Beyond Reengineering*, Harper Collins, New York, 1996.

[Jacobson92] I. Jacobson et al. *Object-Oriented System Engineering - A Use Case Driven Approach.* Reading, MA. Addison-Wesley, New York, 1992.

[Humphrey95] W. Humphrey,*A Discipline for Software Engineering,* Addison-Wesley, Reading, MA.,1992.

[Jablonski96] S. Jablonski, C. Bussler. *Workflow Management: Modelling Concepts, Architecture and Implementation*, International Thomson Computer Press, London, 1996.

[Jacobson92] I. Jacobson et al. *Object-Oriented System Engineering - A Use Case Driven Approach.* Reading, MA. Addison-Wesley, New York, 1992.

[Jacobson94] I. Jacobson. *Business Process Reengineering with Object Technology*. Object Magazine 4(2), 1994.

[Jacobson95] I. Jacobson et al. *The Object Advantage*. Reading, MA. Addison-Wesley. New York, 1995.

[McMaster96] M. D. McMaster, *The Intelligence Advantage - Organizing for Complexity*, Butterworth-Heinemann, Boston, 1996.

[MartinJ95] J. Martin, *The Great Transition*, AmaCom, New York, 1995.

[MartinJ96] J. Martin, *CyberCorp,* AmaCom, New York, 1996.

[MartinR95] R. Martin, *Designing Object Oriented C++ Applications using the Booch Method*, Prentice Hall, Englewood Cliffs, 1995.
[OMG-BAA95] OMG, *OMG Business Application Architecture*, White Paper, Framingham, MA, 1995.
[OOPSLA96] C. Alexander, *OOPSLA Speech*, transcript, (provided by Dan Palanza), 1996.
[Porter80] M. Porter, *Competitive Strategy*, Free Press, New York, 1980.
[Porter85] M. Porter, *Competitive Advantage*, Free Press, New York, 1985.
[POSA96] F. Buschman, et. al., *A System of Patterns*, John Wiley & Sons, Chichester (UK), 1996.
[Rummler90] G. Rummler and A. Brache. *Improving Performance - How to manage the white space on the organization chart*, Jossey-Bass, San Francisco, 1990.
[SEI95] Carnegie Mellon University - SEI, *The Capability Maturity Model - Guidelines for Improving the Software Process*, Addison and Wesley, Reading, 1995.
[Senge90] P. Senge, *The Fifth Discipline - The art and Practice of the Learning Organization*, Doubleday/Currency, New York, 1990**.** [Shelton94] R. E. Shelton. *Software Process Reengineering*. Supplement to Object Magazine, 1994.
[Senge94] P. Senge, et. al. *The Fifth Discipline - Fieldbook*, Doubleday/Currency, New York, 1994.
[Shelton94] R. Shelton. *The distributed enterprise.* Supplement to Object Magazine, 1994.
[Shelton-F] R. Shelton, *OOBE: The Modeler's Handbook*, Addison and Wesley, Forthcoming title.
[Sowa] J. F. Sowa and J. Zachman, *Extending and formalizing the framework for information systems architecture*, IBM Systems Journal, 31(3), 590-616.
[Skunk97] Alan Webb, Project Management Today, http://www.projectnet.co.uk/pm/pmt/kelly97.html
[Stacey96] R. D. Stacey, Complexity and Creativity in Organizations, Berret-Koehler Publishers, San Francisco, 1996.
[Sutherland96] J. Sutherland, *Scrum Web Home Page*, http://www.tiac.net/users/jsuth/scrum/, 1996.
[Taylor95] D. Taylor. *Object Oriented Business Engineering*. Englewood Cliffs, New Jersey. Prentice Hall, 1995.
[UML97] G. Booch, J. Rumbaugh, I. Jacobson, *Unified Method*, version 1.0, Rational, 1997.
[Yourdon95] E. Yourdon, P. Harmon, *BPR and Software Development*, Cutter Information Corp., Arlington (Ma), 1995.
[Wheatley94] M. Wheatley, *Leadership and The New Science*, Berret Koehler, San Francisco, CA, 1994.
[Zachman] J. Zachman, *A framework for information system architecture*, IBM Systems Journal, 26(3), 276-92.