Examining enterprise unified process

Introduction

A key success factor for any enterprise software implementation is the active existence of proper enterprise process governed by best practice recommendation from enterprise architectural reference. However, a very common symptom across industries is the absence of a process aligned with enterprise recommendations as well as architecture centric, and technology driven project delivery model. This quick ROI policy in most of the industry is infected with inadequate theoretical or conceptual foundations and shows a different tropical infection symptom which further requires refractory of IT strategies, architectures and policies. This article collates a few recommendations and best practices in the context of the generalized reference architectural approaches. In this paper the author briefly describes **the life cycle phases in enterprise architectural context, life history: The timeline aspect in methodologies, Enterprise modelling with integrated model representation and enterprise entity types**. Let an Enterprise Reference Architecture strategically guide the process aspects and process guides the enterprise project delivery roadmap.

In these next few sections, author describes how EUP is carrying most of the enterprise recommended aspects. With the proper foundation, EUP provides us a preventive process aligned with enterprise reference architectural foundations [also satisfies most of the recommendation of ISO 15704] which does not require any reactive approach to refractor any incorrect lifecycle aspects. Concisely in this paper, I show how EUP empowers your enterprise with its true enterprise nature.

What is Enterprise Process?

Enterprise process is a derivative of Enterprise reference architecture specification to capture both the functional and behavioural aspects of any enterprise systems. Enterprise architectural principles covers enterprise aspects ranging from the organizational structure, business processes, information systems, and infrastructure. It points out all essential steps or phases of the programs, identifies all tasks involved and shows their interrelationship. Enterprise process deals with all these enterprise activities including human oriented concepts, process oriented concepts and technology oriented concepts.

An enterprise is influenced by enterprise domains and enterprise environments. Each enterprise is an aggregation of multiple entities. On the other hand, enterprises have different set of stakeholders and associated enterprise concerns. Enterprise process ensures predictable behaviour of enterprise activities to satisfy all concerns of the identified stakeholders. Due to unclear understanding about the enterprise concepts, most of the organisations falsely lead towards an unrealistic enterprise process which carries the most common symptoms of interoperability problem. This is because these processes are mostly developed on the basis of several stovepipe applications which are successful in tactical sense but becomes the actual nightmare from the strategic point of view.

An Enterprise process objective is to unify all these enterprise activities within a guided environment. GERA defines following process oriented concepts:

a) Enterprise entity life cycle
b) Enterprise life history
c) Enterprise entity types, and
d) Enterprise modelling with integrated model representation and model views

Moreover GERA has introduced the human oriented, process oriented and technology oriented concepts as part of the recommendations.

Why GERAM?

Well-planned enterprise architecture ensures all possible business benefits through technology innovation. Moreover the use of established reference architecture as a framework for the process descriptions provides several demonstrable benefits. According to TOGAF "*The primary reason for developing enterprise architecture is to support the business by providing the fundamental technology and process structure for an IT strategy. This in turn makes IT a responsive asset for a successful modern business strategy*" [TOGAF]. Organizations embracing changes on their strategic plan usually prefer to use one of the available reference architectures to speed up implementation and take advantage of collated best practices. However, "*GERAM is not yet-another-proposal for enterprise reference architecture, but is meant to organise existing enterprise integration knowledge. The framework has the potential for application to all types of enterprise. Previously*

published reference architectures can keep their own identity, while identifying through GERAM their overlaps and complementing benefits compared to others" [GERAM].

GERAM / ISO IS 15704: 2000 provides a generalized set of recommendations which may be considered as common baseline requirements that constitute enterprise architecture and engineering. Such a baseline allows enterprises and architects to assess various candidate reference architectures and choose the most appropriate one based on their specific business needs. This makes the rationale for choosing a specific architecture reasoned and rational and not impulsive and swayed by hype.

"*The main aim of Generalised Enterprise Reference Architecture and Methodology is to generalise the contributions of various existing and emerging Enterprise Architecture Frameworks and Enterprise Reference Architecture in order to define a complete collection of tools, methods, and models to be employed by any enterprise engineering and integration effort. As such, GERAM assists in the choice of tools and methodologies by providing criteria that needs to be satisfied rather than trying to enforce particular options. Used as a generalisation of frameworks, GERAM may also assist in establishing the completeness and suitability of frameworks proposed to form the basis to a particular change process*." [Ovidu Noran].

GERAM enterprise life cycle aspect

GERA defines seven life-cycle activities (refer figure 1) for any enterprise or any of its entities that are pertinent during the life of the entity. These activities may be subdivided further into several lower level types of activities (based on the customary subdivision in many industries of design into preliminary- and detailed design activities). The life-cycle diagram used in the description of the life-cycle of an entity is itself a model of the enterprise engineering methodology.
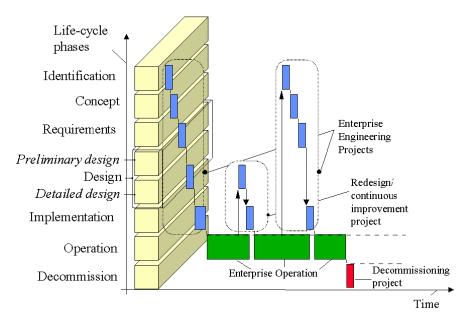

Figure 1: GERAM Life-Cycle and Life history of an enterprise entity [GERAM]. Used with kind permission of Prof P Bernus.

Following table summarizes the seven life-cycle steps of recommended by GERAM:

Life-Cycle activities	Sub-activities	Description
Entity Identification		This step identifies the particular enterprise entity in terms of its domain boundaries and its environments
Entity Concept	❑ Mission ❑ Vision ❑ Values ❑ Strategies ❑ Objectives	These are set of foundational activities which are needed to consider within the underlying enterprise entity during its life history.

	❑ Operational Concepts ❑ Policies ❑ Business Plans	
Entity Requirement		These are set of human oriented aspects, process and technology oriented activities needed to describe the operational requirements of the enterprise. These requirements are expressed in terms of stakeholders' concerns, Views and View Points. Eventually these requirements could be expressed in terms of UML use-cases.
Entity Design	❑ Functional Design ❑ Detailed Design	This step models the enterprise entity. In this life-cycle step, enterprise requirements are sub-divided into functional design and detailed design. These designs are suitable fabrication of the actual system which helps us to understand the intended system functionalities.
Entity Implementation		Designed system is actually developed into real components in this implementation phase. Once the development is finished, system is thoroughly tested and validated. After finishing the successful system integration and acceptance test, system is released into operation.
Entity Operation		This phase is actual production system where system is actually tested by the end users or customers. Based on user feedback, this step may generate a change request or feature update request which in turn may initiate the same cycle starting with requirement phases.
Entity Decommissioning		After a long successful run of the operation, enterprise may require to dispose of all or part of the entity which may start with re-visioning, redesigning and recycling of the system.

In reality all these steps may be evolved in an iterative way. All these life-cycle steps are really very crucial for any entity. Therefore, any enterprise process should be able to support us with all the GERAM recommended steps. In our later part of this article, we will explain how EUP is helping us in terms of life-cycle aspects.

GERAM Life History aspects: The timeline aspect of methodologies

Life history is the actual sequence of steps (also known as activity types) an entity evolves over its lifetime. Any enterprise entity has its own history that captures the main events, or milestones in the life of the entity. Change processes that may happen simultaneously and may need to be captured as parts of the operational processes are inevitable within the history of an entity. Figure 1 shows the life history aspect of an entity within the enterprise, life history representing a simple case with a total of seven processes: three engineering processes (in blue colour), three operational processes (in green colour), and one decommissioning process (in red colour).

One entity may go through several events based behavioral states to perform its functionality during its entire life history.

Life history of one entity may affect the life-history of other entities. This feature of entity life history introduce the concept of life-history traceability and may encourage concern oriented development concept during development phase.

Life Cycle

Entity 1
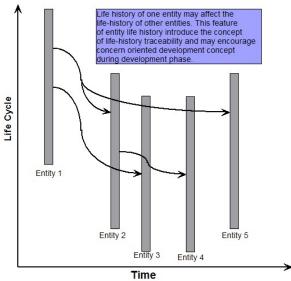
Entity 2

Entity 3   Entity 4

Entity 5

Time

Figure 2: Life history of one entity may affect life history of other entity.

Enterprise deals the cross-system issues more occasionally which are not easy to capture with enterprise life cycle only. The critical difference between life cycle phases and timeline aspect is that the evolution of the enterprise entity through time is not covered in the life cycle phases. Life history of any entity describes its evolution and adaptability towards any change in business environment. This demonstrates that any event in the life history of the entity may affect the life history state of other entities which may be described as follows in figure 2. This life history diagram points to the need for managing systems and projects at an enterprise level.

Enterprise modelling with integrated model representation and model views

GERA divides enterprise into multiple views to reduce model complexity and cover the viewpoints of different stakeholders. GERA identifies the following model views:
1. **Model Content Views**: Function, Information, Resource, Organization
2. **Purpose Views**: Customer Service, Management & Control
3. **Implementation Views**: Human (Manual) and Machine (Automated Tasks)
4. **Physical Manifestation Views**: Software, Hardware

Views

Generic
Partial
Particular
} Subdivision according to genericity

Instantiation

Identification
Concept
Requirements
Preliminary design
Design
Detailed design
Implementation
Operation
Decommission

Life-cycle phases

Customer service
Management- and control
} Subdivision according to purpose of activity

Software
Hardware
} Subdivision according to physical manifestation

Resource
Organisation
Information
Function
} Subdivision according to modelling views

Machine
Human
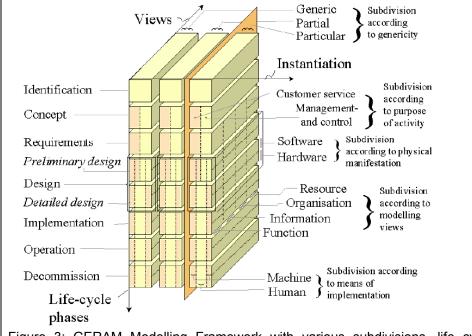} Subdivision according to means of implementation

Figure 3: GERAM Modelling Framework with various subdivisions, life cycle phases and model categories [GERAM]. Used with kind permission of Prof P Bernus.

Primary focus of Model Content Views is user oriented process representation of enterprise entities whereas Physical Manifestation Views addresses software and hardware. On the other hand Entity Implementation Views distinguishes between automated task and task needed manual intervention. Purpose Views with focus on mission of the enterprise entity and products & services needed to support enterprise objectives roughly map to TOGAF Business Architecture Views.

Enterprise entity types

As already mentioned GERAM prefers identifying entity as an essential activity in enterprise life-cycle and relationship between them. An entity is a logical/virtual organization, which could be a department, division or an entire organization, with a life history (Refer to Y-axis in figure 2). This entity classification helps any organization to build up more structured enterprise foundations. GERA introduces the concept of five most important entity types as follows:

1. Strategic enterprise management entity
2. Enterprise engineering entity
3. Enterprise entity
4. Product entity
5. Methodology entity

GERA also identifies the concept of Recursivity of entity types, where Recursivity is defined as the direct and active influence of one entity in the development of another entity.

So far we have described GERA and its process oriented concepts applicable to all enterprises. As mentioned earlier, the main objective of this paper is to provide enterprise decision makers, architects and engineers analysing EUP from enterprise point of view, with an understanding of GERA specified recommendations. This foundation knowledge would further help us assessing EUP from GERA point of view and to justify whether it is adequately equipped with all necessary enterprise recommendations. Now we will briefly discuss Enterprise Unified Process.

According to EUP "*The Enterprise Unified Process (EUP) is an extension to the RUP that is best for organizations with multiple systems and multiple development teams. The EUP includes new disciplines and phases which should be tailored into the standard RUP, making it more effective for these types of organizations*" [EUP 2005]. RUP describes nine disciplines which are extended by EUP. EUP adds seven new enterprise disciplines to tackle enterprise issues.

RUP as a software development process

RUP is one of the first iterative software development processes in the market. The RUP is based on six best practices: develop iteratively, manage requirements, use component architecture, model visually (UML), continuously verify quality, and manage change. RUP is initially created on the basis of diagnosing the characteristics of different software projects that failed. In that way, RUP is a reverse-engineering effort to ensure the success of IT projects and resolve those project failure symptoms.

RUP disciplines, roles, deliverables

Assuming the fact that readers are familiar with RUP concepts, RUP concept could be summarised as a software development process with four life-cycle phases and nine disciplines. The static aspect of this process describes who is doing what, when, and how they are doing that particular piece work (represented as what):

A. Workers are who
B. Artifacts are what
C. Workflow is when
D. Activity is how

Following table simplifies the relationship between Who, What and When:

Disciplines/When	Participant Roles/Workers	Deliverable Artifacts
Business modelling	Business-Process Analyst, Business Designer	Target-Organization Assessment, Business Vision, Business Glossary, Business Rules, Supplementary Business Specification, Business Use-Case Model, Business Analysis Model, Business Architecture Document
Requirements	System Analyst, Requirements Specifier, User-Interface Designer	Requirements Management Plan, Software Requirement, Stakeholder Requests, Glossary, Vision, Requirements Attributes, Supplementary Specification, Software Requirements Specification, Use-Case Model, Storyboard
Analysis and Design	Software Architect, Designer, Test	Reference Architecture, Architectural Proof-of-Concept, Software Architecture Document, Use-Case Realization,

	Designer, User-Interface Designer	Analysis Model, Design Model, Deployment Model, Data Model, User-Interface Prototype, Navigation Map
Implementation	Integrator, Software Architect, Implementer	Build, Implementation Model, Integration Build Plan, Developer Test
Test	Test Manager, Test Designer, Test Analyst, Tester	Test Plan, Test Evaluation Summary, Test Strategy, Test Idea List, Test Case, Test Data, Test Results, Workload Analysis Model, Test Interface Specification, Test Script, Test Suite, Test Environment Configuration, Test Automation Architecture, Test Log
Deployment	Deployment Manager, Configuration Manager, Graphic Artist, Implementer, Technical Writer, Course Developer	Deployment Plan, Product, End-User Support Material
Configuration and Change Management	Configuration Manager, Configuration Manager, Any Role, Configuration Manager, Change Control Manager	Configuration Management Plan, Project Repository, Workspace, Configuration Audit Findings, Change Request
Project Management	Deployment Manager, Project Manager, Plan Project Manager, Reviewer	Deployment Plan, Business Case, Software Development, Risk List, Issues List, Iteration Plan, Iteration Assessment, Status Assessment, Work Order, Project Measurements, Review Record
Environment	Process Engineer, System Administrator, Tool Specialist	Development Organization Assessment, Development Process, Development Infrastructure, Tools

The Rational Unified Process® or RUP® provides necessary roles and artifacts for each of the disciplines to ensure the production of high-quality software. However it does not explicitly specify any tools for each of these disciplines. It highly suggests them through the use of tool mentors though.

RUP Limitations

RUP is one of most successful software development process which is iterative in nature and one of the first processes which encouraged UML as the common process-oriented language protocols. RUP is very much development focused process and that is why it can not see anything beyond the development boundary like enterprise level issues and different management aspects of the organizations.

EUP extensions with enterprise promises

The Enterprise Unified Process (EUP) is an extension to RUP concepts and best practices to fit it with all enterprise essences so that is best suit for enterprise organizations. The EUP introduces five new best practices as Proven architecture, Modelling, Collaborative development, Look beyond development, Deliver working software on a regular basis and Manage risk. The EUP includes two new phases and seven new enterprise management disciplines to handle multi-geographic presence and cross-system issues that organizations should address to be successful at IT. Enterprise Unified Process is described as below:
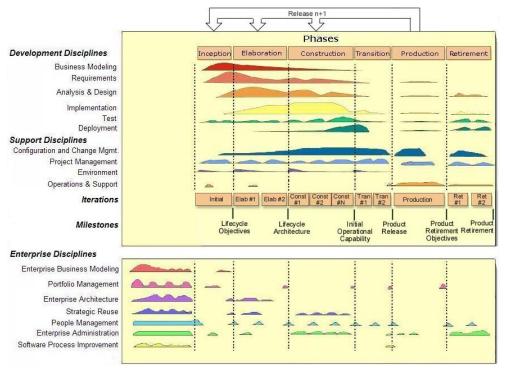
Figure 4: Enterprise Unified Process lifecycle. Used with kind permission of Scott W Ambler.

Two new phases

1. Production Phase

The primary focus of the Production phase is offering required assistance to the end users by clarifying their queries, resolving the production issues through the combination of root-cause analysis and applying appropriate fixes which includes dealing with change request from users as well.

2. Retirement Phase

This phase guides us how to effectively decommission the product from the production phase to the retired phase.

Seven new disciplines

A. Operations and Support Discipline

A well-planned operation and support discipline is absolutely critical for any enterprise to maintain and improve ROI of the enterprise economy. The Operations and Support discipline facilitates uninterrupted operations of software systems. It is aimed to ensure a fault tolerant environment and recoverable system through appropriate backup & disaster recovery plan.

B. Enterprise Business Modelling Discipline

One incredibly important feature of any enterprise is to adequately capture cross-system life-history aspects as described in figure 6. These enterprise models help to distinguish the relationship between Business process, domain process, enterprise activities, functional entities and functional operations. In this phase enterprise business modellers will work closely with the enterprise stakeholders to delimit the scope of their project, in particular helping them to identify the goals, targets, and vision for your enterprise.

C. Portfolio Management Discipline

Enterprise success factor depends on the overall efficiency and the effectiveness of managing a diversified mix of software portfolio (collection of software systems). Portfolio and Program management (maintaining a software systems inventory snap-shot) must be a continuous activity throughout the enterprise system lifecycle as it to mitigate enterprise risks.

D. Enterprise Architecture Discipline

The Enterprise Architecture discipline defines state-of-the art enterprise architecture which includes architectural frameworks, reference architectures and architectural standard & guidelines.

E. Strategic Reuse Discipline

Reusing wide varieties of software assets starting from source code to architectural frameworks is one of the best practices within the software industry.

F. People Management Discipline

The People Management discipline focuses the human oriented management aspect development of any IT organization. HR strategies that aligned with the organization's unique IT strategy is one of the most important success factors of today's IT industry that helps IT people to grow wishfully, and motivate people to ensure their hearty contribution towards the projects.
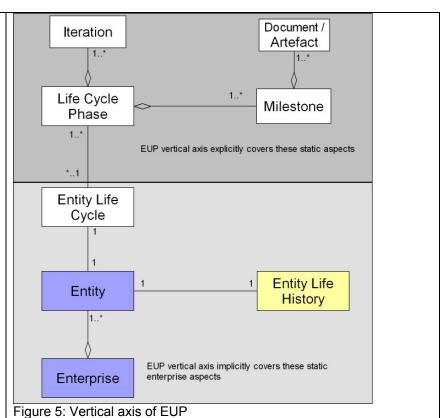
G. Enterprise Administration Discipline

The Enterprise Administration discipline covers the enterprise level administrative responsibilities and initiatives to provide an active support towards the project teams.

Static aspects of EUP

The vertical axis of EUP represents the static aspects of enterprise like activities, artefacts, workers and workflows. The horizontal axis represents time and dynamic aspects of EUP. From that point of view the EUP is described in cycles, phases, iterations and milestones and could be depicted as figure 5:

Each phase is associated with number of milestones (exit criteria) and therefore with different number of deliverable documents (may contain models, diagrams, reports, analysis, audits etc).

Figure 5: Vertical axis of EUP

EUP disciplines, roles, deliverables and relevant tools

Disciplines	Participant Roles	Deliverable Artifacts	Tools
Operations and Support	Operator, Support Manager, Customer Representative, Operations Manager, Support Developer, Support Manager, System Support Representative	Alert, Defect Report, Fix, Hot Fix, Usage report, Backups, Disaster Recovery Plan, System, System Support Plan, System Operation Plan, Service Pack	IBM Tivoli (www.tivoli.com), Operations-oriented tools by Opsware, Inc. (www.opsware.com), Remedy (www.remedy.com) ,Revelation Help Desk (http://gg1.revelationhelpdesk.com)
Enterprise Business Modelling	Enterprise Business Modeller, Enterprise Stakeholder,	Enterprise business rules specification, Enterprise business process model, Enterprise domain model, Enterprise mission statement, Enterprise vision, Organization model, Project proposal, Modelling Guidance, Enterprise Business Glossary, Project Proposal, Enterprise Goals and Targets	Whiteboards, Modelling tools, Diagramming tools, Business activity monitoring (BAM) software
Portfolio Management	Vendor Manager, Portfolio Manager, Enterprise Configuration Control Board, Program Manager, Program Reviewer	Portfolio, Portfolio Plan, Trouble Ticket, Contract, Program, Program Plan, Enterprise Risk List, Enterprise Risk Management Plan, Enhancement Request, Request For Information, Request For Proposals	Artemis 7 (us.aisc.com), Primavera Enterprise (www.primavera.com), PMOffice (www.systemcorp.com), UMT (www.umt.com)
Enterprise Architecture	Enterprise Architect, Application Architect, Technical Reviewer, Reuse Engineer, Technical Reviewer	Candidate Architecture, Enterprise Architecture Model, Enterprise Business Architecture Requirements, Reference Architecture, Enterprise security Plan, Enterprise Technical Requirements	Modelling tools like System Architect from Popkin (www.popkin.com) and Rational Rose/XDE from IBM Rational (www.ibm.com), Diagramming tools such as Microsoft Visio (www.microsoft.com) and CorelDraw (www.corel.com), Interactive presentation tools like Webex (www.webex.com), Non-interactive communication tools: like Groove (www.grove.net) or Wikis (www.wiki.org) and Whiteboards
Strategic Reuse	Reuse Engineer, Reuse Manager, Reuse Registrar, Reuse Consumer	Asset Criteria, Asset Rework Plan, Configured Asset, Announcement, , Published Asset, Robust Asset, Reuse Guidance, Reuse Measurement Plan, Reuse Program Plan	Shared directory, Version control tool, Reuse repository tool like Flashline (www.flashline.com) , Logidex (www.logiclibrary.com)
People Management	Human Resource Manager	Benefits Package, Contract, Education Plan, Individual Career Plan, Position	Web-based training tracking and management system Intelex

		definition, Staffing Request, Staffing Forecast, Long Term Succession Plan, Needs Assessment, Organization Assessment	(www.intelex.com/training) , PeopleSoft and several Open-source options
Enterprise Administration	Information Administrator, Enterprise Administrator, Network Administrator, Facilities Administrator, Security Administrator	Enterprise Data model, Enterprise Intellectual Property, Enterprise Data Guidance, Enterprise Facilities Guidance, Enterprise Network and Hardware Guidance, Enterprise Security Guidance, Federal Regulations, Legacy Data Source Documentation, Enterprise Computing Infrastructure Plan, Enterprise development Guidance, Industry Guidance, Enterprise Facilities Plan	IBM Tivoli (www.tivoli.com), 360Facility (www.360facility.com) and ManageSoft (www.managesoft.com)
Software Process Improvement	Process Engineer	Existing Process, IT Process, Process Implementation Plan, Process Vision, Development Case, Organizational-Level Development Case	RUP Modeler, RUP Process Workbench, RUP Builder, IRIS Process Author (www.osellus.com) with its RUP bridge, Waypointer (www.jaczone.com) , Wiki (www.wiki.org), Groupware and collaboration like eRoom (www.documentum.com) or Groove Virtual Office (www.groove.net)

From the above table it is clear that EUP not only extends the RUP disciplines, it also helps us with the recommended tools. EUP followers get this tool guidance as an added advantage which helps the user to select their tools from a wide range of selection as part of their process.

Comparing GERA and EUP

From our previous discussion of GERA process recommendations and various EUP disciplines we can derive following comparison matrix:

GERA Recommendations	EUP Disciplines	Special Comments
GERA Life Cycle phases		
Entity Identification	Enterprise Business Modeling	Introduced within EUP. For a very brief introduction please refer to EUP Extension section
Entity Concept	Enterprise Business Modeling	
Entity Requirement	Requirements, Business Modeling	Please refer any RUP manuals for further details about these disciplines.
Entity Design	Analysis & Design	
Entity Implementation	Implementation, Test, Deployment	
Entity Operation	Operation and Support discipline, Production Phase	This is one of the important enterprise environment considerations of EUP to focus on software production.
Entity Decommissioning	Retirement Phase	EUP extended this phase. EUP shows how to extract different software assets from this phase to be considered in Portfolio Management and Strategic Reuse discipline.
GERA Enterprise life history	Enterprise Business Modeling, Configuration & Change	Introduced within EUP. For a very brief introduction please refer to EUP

	Management	Extension section.
GERA Entity Types	Enterprise Business Modeling	
Enterprise modelling with integrated model representation and model views		
Content View	Portfolio Management, Enterprise Administration, Strategic Reuse	Introduced within EUP. For a very brief introduction please refer to EUP Extension section.
Purpose view	Support & Operations, Configuration & Change Management	
Implementation View	Project Management, People Management,	
Physical Manifestation View	Environment, Support & Operations,	
Other aspects of GERA		
Human oriented Concepts	Project Management, People Management, Roles,	Introduced within EUP. For a very brief introduction please refer to EUP Extension section.
Process Oriented Concept	Software Process Improvement	
Technology Oriented Concept	Portfolio Management, Enterprise Architecture, Enterprise Administration, Strategic Reuse	

Few missing chapters and EUP future directions

The analysis performed in this paper has shown that EUP covers almost all of the process oriented concepts recommended by GERA; however it is still weak in few areas. Following areas are still needed more coverage:

a) **Entity Life History**

   GERA recommended entity life history aspect of enterprise helps to capture cross-system interactions. It helps to maintain traceability between cross-cutting aspects of entity, one of the very important features to capture cross-cutting concerns. EUP enterprise business modeling and configuration management open a little bit opportunity to address this issue but really not that much helpful. EUP should be able to address these issues.

b) **Aspect oriented model driven concept and generating more reusable assets**

   EUP evolved initially as an extension to the RUP. Therefore it is understandable why EUP is not likely to provide any recommendations on different best practice software development approaches and modelling techniques. However from strategic reuse and portfolio management point of view EUP should provide us some light on this type of emerging techniques those are associated with strategic reuse and portfolio management techniques. Aspect oriented Software Development techniques is a paradigm shift from traditional object oriented software development towards concern driven enterprise. Aspect Oriented Development has introduced the concept of Crosscutting Concerns, the secret window of ample of opportunities. Aspect Oriented, Concern Driven software development approach profoundly change the concept of reusable, concern-oriented code development practice. On the other hand, EUP should encourage Agile Model Driven Design techniques to generate reusable model assets.

c) **Process derivation techniques**

   During our software development phase or enterprise development phase, we may need to create a set of other processes like code audit process, software performance management process, capacity planning, test management process etc. EUP guidance with few derivative process examples on this aspect of process derivation techniques would benefit current IT industry in a greater level. EUP may provide a process inheritance framework in this respect.

d) Quantitative Process management

Use of statistical techniques to control and improve processes has become a driver of process improvement process. SEI has introduced Quantitative Process Management [QPM1, QPM2] as part of the CMM level 4 certification. EUP guidelines on this quantitative process management technique would be great help for any enterprise process management group and project management team.

Conclusions

The Enterprise Unified Process (EUP) augments RUP to address the enterprise features rather than just software development which is a part of enterprise lifecycle. EUP not only enhances the RUP disciplines with the addition diversified disciplines like enterprise business modeling, portfolio management, enterprise architecture, strategic reuse, people management, enterprise administration, software process improvement and operations & support, at the same time it also introduces a proper enterprise project lifecycle within the enterprise. At the moment EUP strives to be a complete enterprise process with a few missing bits. However EUP is continuously aiming to reduce the process complexities and increasing perpetuating best practices. EUP is evolving very fast and is highly likely that agile nature of EUP would eventually address all those few missing issues and process recommendations by GERAM and incorporate all other process oriented best practices. EUP has definitely extended almost all RUP shortcomings and promised to assist our current enterprise software development success and advancement of enterprise engineering. Current IT industry will be greatly aided by necessary EUP impetus.

Acronyms

CIMOSA	Computer Integrated Manufacturing Open Systems Architecture
TOGAF	The Open Group Architecture Framework
AOSD	Aspect Oriented Software Development
MDA	Model Driven Architecture

Copyright Declarations

a) Figure 1, 3 sources are [GERAM]. Used with kind permission of Prof P Bernus
b) Figure 4 source is [EUP Concept]. Used with kind permission of Scott W Ambler

Acknowledgements

An enterprise could be succeeded only because of the dedicated work of many great thinkers', some of whom I have mentioned in my little bibliography while many others are still remaining as anonymous. To all those great persons who have influenced me and encouraged my understanding through their writing, I acknowledge my indebtedness and express my great appreciation. The author is sincerely thankful to Scott W Ambler for his valuable review comments which helped me to finish this article. The author is grateful to his suggestions.

Disclaimer

All information provided by The Ultimate People Company Ltd (UPCO), including, without limitation, offerings, statement of future directions, "white papers" or other technical or marketing materials (collectively, "information") is intended for informational purposes only and is subject to change or withdrawal by UPCO at any time without notice. UPCO assumes no responsibility for the accuracy or completeness of the information. The information is provided "as is" without warranty of any kind including (without limitation any implied warranties of fitness for a particular purpose. In no event will UPCO be liable to any party for any direct, indirect, special or other consequential damages for any use of the information, including without limitation, any lost profits, business interruption, loss of programs or other data on your information handling systems or otherwise.

References

Standards

1. [GERAM] GERAM: Generalised Enterprise Reference Architecture and Methodology. Version 1.6.3 (http://www.cit.gu.edu.au/~bernus/taskforce/geram) also in P.Bernus, L.Nemes and G. Schmidt (Eds) Handbook on Enterprise Architecture, Berlin : Springer (2003) pp 22-64.
2. [TOGAF] TOGAF (The Open Group Architecture Framework) Version 8.1 "Enterprise Edition". Available from http://www.opengroup.org
3. [EUP concept] www.ambysoft.com
4. [EUP 2005] Enterprise Unified Process: Extending the Rational Unified Process by Scott W. Ambler, John Nalbone, and Michael Vizdos, published Prentice Hall PTR
5. [PERA master plan] A HANDBOOK ON MASTER PLANNING AND IMPLEMENTATION FOR ENTERPRISE INTEGRATION PROGRAMS Based On The Purdue Enterprise Reference Architecture and the Purdue Methodology Purdue Laboratory for Applied Industrial Control Edited by Theodore J. Williams,Gary A. Rathwell,Hong Li. February 2001 (Revised from July 1999 PERA website version)
6. [CIMOSA] http://www.pera.net/Methodologies/Cimosa/CIMOSA.html
7. [IEEE] IEEE Recommended Practice for Architectural Description of Software- Intensive systems. Institute of Electrical and Electronics Engineering Std 1471-2000
8. [ISO 9126] Software Quality: The Elusive Target by Barbara Kitchenham and Shari Lawrence Pfleeger, The Software, IEEE, January 1996 (Vol. 13, No. 1) pp. 12-21
9. [ICEIMT GERAM] The Contribution of the Generalised Enterprise Reference Architecture to Consensus in the Area of Enterprise Integration by Peter Bernus, School of Computing and Information Technology, Griffith University, Nathan (Brisbane) QLD 4111, Australia <bernus@cit.gu.edu.au> and Laszlo Nemes, Division of Manufacturing Science and Technology, CSIRO, Locked Bag No 9., Preston, VIC., 3072, Australia <lnm@dmt.csiro.au>, Published for ICEIMT GERAM, on 21 August, 1997
10. [ISO 15704] http://www.iso.org
11. [QPM 1] http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr010.pdf
12. [QPM 2] http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr037.pdf

Bibliographies

13. Enterprise Unified Process, The: Extending the Rational Unified Process By Scott W. Ambler, John Nalbone, Michael J. Vizdos, Published by Prentice Hall PTR, ISBN: 0-13-191451-0
14. Handbook on Enterprise Architecture, by Bernus, P., Nemes, L. and G. Schmidt (eds.) , Springer, (2003), ISBN is 3540003436
15. How to survive in the jungle of Enterprise Architecture frameworks, Second Edition, by Jaap Schekkerman, published by Trafford, ISBN 141201607-X
16. Balancing Agility and Discipline: A Guide for the Perplexed by Barry Boehm, Richard Turner, Published by Addison Wesley
17. Performance Engineering as a Part of Development Life Cycle for Large-Scale Software Systems by Gregory Fox, TRW Federal Systems Group, Fairfax, VA
18. Performance oriented software architecture engineering: an experience report, by Chung-Horng Lung, Anant Jalanpukar and Asham El-Rayess, Software Engineering Analysis lab, Nortel, Canada
19. Performance Engineering: State of the Art and Current Trends, LNCS, Vol 2047, Springer 2001
20. Evaluating the Performance Engineering Process: Andreas Schmietendorf, Andre Scholz, Claus Rautenstrauch, University of Madgdenburg, School of Computer Science
21. Process Models for the software development and performance engineering tasks: Andreas Schmietendorf, Evgeni Dimitrov, Reiner R. Dumke

Research Papers and Publications

22. Quantitative Analysis of Enterprise Architectures by Maria-Eugenia Iacob (MariaEugenia.Iacob@telin.nl) and Henk Jonkers (Henk.Jonkers@telin.nl) Telematica Instituut, P.O. Box 589, 7500 AN Enschede, the Netherlands

23. An architecture for defining the processes of the software and systems life cycles by Terence P. Rout, Software Quality Institute and Peter Bernus, School of Computing and Information Technology, Griffith University Queensland, Australia.
24. PERA AND GERAM--ENTERPRISE REFERENCE ARCHITECTURES IN ENTERPRISE INTEGRATION by Theodore J. Williams,Institute for Interdisciplinary Engineering Studies,Purdue University and and Hong Li, Senior Consultant, Claremont Technology Group, Inc
25. Concepts for Modelling Enterprise Architectures by Henk Jonkers1, Marc Lankhorst1, René van Buuren[1], Stijn Hoppenbrouwers[2], Marcello Bonsangue[3], Leendert van der Torre[4] where [1]Telematica Institute, [2]University of Nijmegen, Nijmegen, the Netherlands, [3]Leiden Institute for Advanced Computer Science, Leiden, the Netherlands, [4]CWI, Amsterdam, the Netherlands

Materials from Internet

26. Enterprise Modelling by Marks S. Fox and Michael Gruninger (This is a publication of The American Association for Artificial Intelligence)
27. Role of Model-Driven Architecture in Business Integration by Sundar Vaidyanathan, Business Integration Journal, September 2004
28. http://www.agilemodeling.com/essays/mda.htm
29. http://www.enterpriseunifiedprocess.com/
30. http://aelinik.free.fr/cmm/tr25_l4a.html
31. http://www.enterpriseunifiedprocess.com/essays/zachmanFramework.html