Section A-Research paper

ISSN 2063-5346

Design and Metaphor Evaluation for Extreme Software Development

Ch V Phani Krishna[1,2], Nethravathi P S[2], K Bhargavi[1]

[1]Department of Computer Science & Engineering, Teegala Krishna Reddy Engineering College, Hyderabad, Telangana, India-500097.

[2]Department of Computer Science & Engineering, Institute of Engineering & Technology, Srinivasa University, Mangalore, Karnataka, India-574146.

doi: 10.31838/ecb/2023.12.si6.299

ABSTRACT:

This paper presents the Extreme Software Development (XP) Metaphor Evaluation and Architecture. It provides a straightforward, semiotics-based structural model of system metaphors that provides a fundamental account of how metaphors may bring value to a software system. It identifies activities that teams can use to construct metaphors for their systems as well as evaluation techniques for system metaphors. This results will reassure Extreme Programming teams to carry on using system metaphors to reinforce their growth practices, rather than abandoning them.

KEYWORDS: Extreme Software Development; Metaphor; Semiotics; Evaluation; Architecture

INTRODUCTION

By characterizing a system's logical architecture in terms of something familiar to developers and customers, the Metaphor practice system identifies its logical architecture. By providing a shared vocabulary for discussing system problems and solutions, the customer and developer facilitate the discussion of the project in language made available by a system metaphor. The system metaphor provides developers with additional support for maintaining consistency in the nomenclature of their programs' subsystems, packages, classes, and methods. The system metaphor is a low-cost system design that indicates the main system components and their relationships. It is widely acknowledged that metaphor is a fundamental aspect of communication. Extensive use of metaphor in both art and language.

Metaphor is regarded as a form of language by linguists, cognitive linguists, and relevance theorists. Both theories make significant contributions to our comprehension of the function of metaphor in cognition and communication by presenting precise concepts regarding the mechanisms underlying the comprehension of metaphors. Various theories of metaphor exhibit distinct points of origin, such as linguistics, psychology, and philosophy, and culminate in diverse objectives, including cognitive or pragmatic pursuits. The theories of systemic functional linguistics utilise a social semiotic perspective to examine the mechanisms of meaning construction through language and symbols in scientific literature. This approach offers instances of the proliferation of meaning through the application of metaphor. The present study investigates the distinctions between the semantic attributes and pragmatic implementations of three interconnected cross-modal mechanisms, namely inter-semiotic correspondence, trans-semiotic categorization, and inter-semiotic metaphor.

Section A-Research paper
ISSN 2063-5346

BACKGROUND

Over the course of the last decade, a sizeable body of evidence has accumulated in support of the idea that perception and action reflect common processing pathways with cognition. Several researchers, including Sune Dueholm, Muller, and others, have found empirical evidence for the grounded cognition paradigm via their studies incorporating tangible conceptual representations. These researchers' findings have been presented by these researchers.According to the information presented in reference [1,] the accomplishment of Software Process Improvement (SPI) often calls for complex organisational changes to be made. The research finds a number of criteria, including knowledge direction, theoretical focus, primary audience, geographical origin, and publication level, among others. The major emphasis will be placed on organisational change, with Gareth Morgan's organisational analogies serving as the lenses through which analysis will be conducted. Personal environment controllers provide a fresh and organic way of connection, as stated by Ali. A. Nazari Shirehjini [2], which efficiently link the user, their media, and their surroundings. The purpose of Dan L. Chiappe and Penny Chiappe's [3] formulation of the predication paradigm was to evaluate the capacity of working memory that is involved in the processing of metaphors. The cognitive processes that take place inside the working memory provide significant support in the comprehension of metaphors. Liu and Owyong (4) have highlighted the multiplicative character of metaphors by displaying the processes of meaning-making that cover language and symbols in scientific literature. These meaning-making processes may be found in scientific literature. In order to make this study more accessible, they have suggested a social semiotic framework. The mechanisms of intersemiotic correspondence, trans-categorization, and metaphor are investigated in this study as they pertain to cross-modal communication.

Markus Tendahl and Raymond W. Gibbs, Jr.[5] discussed the future of metaphor in linguistics, cognitive linguistics, and relevance theory.In addition to presenting the function of metaphor in cognition and language use, the article proposes detailed hypotheses on metaphor comprehension that are undoubtedly part of a comprehensive metaphor theory. Yair Neuman and Ophir Nave [6] presented an innovative approach for determining the meaning of a target term by analyzing metaphors containing the term. Vinoski, S. [7] described the XP and MDA architectures as well as defining system paradigms and interfaces or contracts with an abstract language akin to IDL. Chi -hua Hsiao and Lily I -wen Su [8] investigated hyperbolic expressions using a variety of metaphor-based syntactic devices, as well as how hyperbole adopts an affective stance to demonstrate intersubjectification. Jiunde Lee [9] examined metaphors, which can function as effective instruments to facilitate the mental modeling processes of users. Diane Pecher et al. [10] examined metaphorical mappings derived from conceptual metaphor theory, where image schemas structure and provide sensory-motor foundations for concepts that are abstract. Within the context of knowledge-based sales management, Sreedhar Madhavaram and Robert E. McDonald [11] presented a novel theoretical metaphor. Narda G. Robinson [12] presented the Sense of the Metaphor, or the Neurophysiological Workings of Acupuncture. Ulrich Flemming et al. [13] reported the results of a comprehensive study of Computer-Aided Drafting (CAD) system users. Mark J. Landau et al. [14] proposed the metaphoric transfer technique as an empirical method for determining whether metaphors influence the processing of social information. Ali Baykal [15] presented the instructional design metaphor of opensystems. Dall'Agnol,

M et al. (2016) conducted a comparative analysis between Lean Management and Extreme Programming. XP, a software development approach, is based on the principles of Agile Methodologies.Agile refers to the capacity to both initiate and adapt to change to succeed in a turbulent business environment. The formulation of XP, according to West, D., and Solano, M. [17], elevated metaphor to a level of essential activity. XP was the sole agile methodology that utilized metaphor in such a transparent and fundamental manner - it was unique.

Upon careful examination of the aforementioned works, it is evident that there exists a limited body of literature pertaining to the selection and application of metaphors. In particular, the research on the system metaphor

Section A-Research paper

ISSN 2063-5346

appears to be relatively scant in comparison to the extensive research conducted on other XP practises, such as Pair Programming, Test Driven Development, and the Planning Game.

METAPHOR

As a means of communication between team members and consumers, metaphor can be described as a communication tool. The majority of XP teams choose to disregard the use of metaphor because it is inadequately understood. As a specific for adopting the explanation point, metaphor is used. A metaphor paradigm facilitates straightforward verbal communication between customer and developer regarding the development of a system. The method Metaphor practise is a method for describing the plausible architecture of a system in terms that are familiar to both the developer and the consumer. System metaphor also promotes consistency in naming programme elements, such as subsystems, packages, classes, and methods. Metaphor is weak as a design, and as team size increases, the metaphor's capacity to be sufficiently decreased. This, coupled with pair programming's diminishing ability to meet the team's communication needs, results in a team that diverges rather than converges swiftly.

Among the XP practises, the metaphor has demonstrated the lowest level of efficacy. The XP methodology endeavours to establish a unified metaphor that encompasses the entirety of the project. It is advisable to generate a metaphorical representation for each individual to whom the model is being elucidated. Individuals hailing from diverse backgrounds may require distinct metaphors. Metaphors are subject to reuse, albeit exclusively among individuals who share a common background.

The utilisation of metaphorical language in programming necessitates the ability to translate it into concrete programme code, thereby producing code that is both coherent and uncomplicated. It is imperative that metaphors accentuate all primary constituents and operations of the system. The utilisation of metaphors ought to encompass a lexicon that is capable of accurately and efficiently delineating the system.

It is imperative that metaphors are devoid of ambiguity and do not imply the existence of non-existent system components or behaviour. The diagram presented in Figure 1 illustrates a comprehensive perspective of metaphors, which are highly effective tools for educational purposes. They are employed across a diverse range of academic fields. Metaphors are utilised to enhance understanding. In order to effectively convey a novel idea, the communicator must establish a mutual understanding of the context with their audience. The XP methodology endeavours to establish a metaphor that can be universally applied.

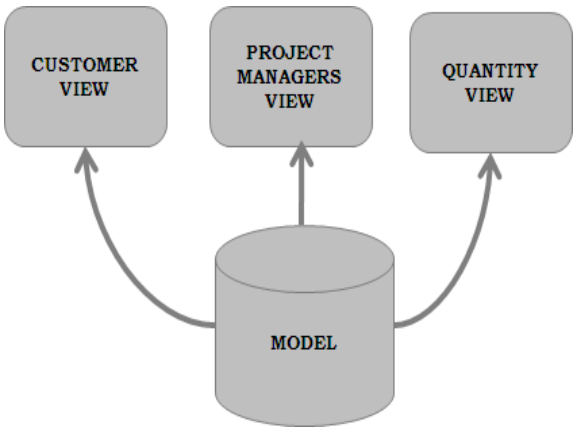

Figure1: MetaphorSystemView

ISSN 2063-5346

METAPHOREVALUATION

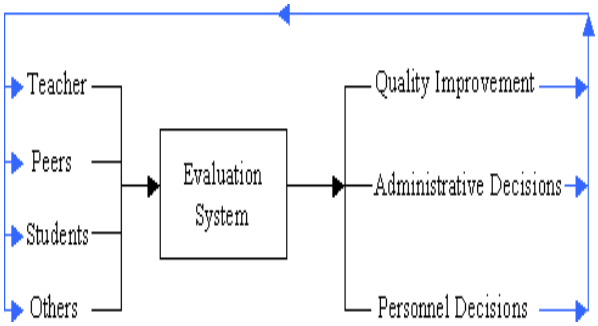

Figure 2: Metaphor Evaluation System

The use of metaphor is emphasised in the Extreme Programming (XP) methodology.The validity of the real computer code is being evaluated metaphorically. The metaphor successfully characterises the key features of the system and how they work separately. The metaphor supplies the language for articulating the system.

Whether the metaphor accurately describes the entire system.The metaphor implies nonexistent elements of the system or nonexistent actions. The metaphor adds unwarranted complexity to the procedure. Figure 2 depicts the rating scheme for metaphors. As shown in the diagram, the evaluation process includes quality control, organisational decisions, and user input.

METAPHORARCHITECTURE

The desktop architecture for PC software and reviewed how closely the then current offerings matched that architecture. The new applications development and was compatiblewiththestandardcomputerthatwerethen being purchased, itonlyprovidedsomeofthe layersinthearchitectureandshowninfigure3

The figure shows that the application suites give the object model. Object model gives thedesktopmanager.The desktop manager gives its icons, dialog manager and data exchange. It all includes in the windows manager.Windowsmanager controls the imaging manager. Finally it gives the systems software.

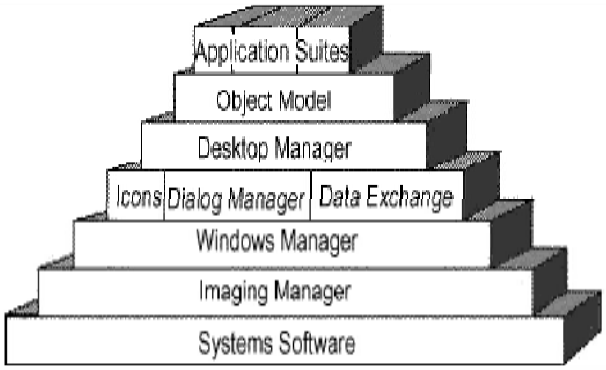

Figure 3: Metaphor Architecture

Section A-Research paper

ISSN 2063-5346

The values of Extreme Programming (XP) are enhanced through the use of metaphor. Communication can be conceptualised as a system. The utilisation of metaphorical language or architectural design is known to augment the effectiveness of communication. A framework designed to manage and organise narratives and their interconnections, which facilitates the progress of developmental endeavours.

and directness in its design approach. The system ought to possess the qualities of swift and effortless modification, as well as conceptual comprehensibility at a glance and minimal reading comprehension requirements. Aside from enabling uncomplicated alterations, the design ought to encompass the ability to represent the state or progression of individual actions, narratives, or subsystems.The architectural form ought to exhibit a boldness that distinguishes the undertaking and inspires enthusiasm.

UNDERSTANDINGOFXPMETAPHOR

Agile software development is a novelmethod; it provides faster software delivery time and communication. Metaphoris the face-to-face discussions among team and team members and client instead of depending on formal certification. ExtremeProgramming(XP)systemmetaphorisanalyzingwithsemiotics seems like an idealtoolanditisaccurateto facilitatestructuredanalysis.

The application of semiotics to the metaphors utilised in XP allows for the utilisation of the extensive research conducted within the broader field of semiotics. XP developers employ semiotics directly to analyse their own metaphors, and the System Metaphor practise is patterned after semiotic theory. The proposed model aims to address the existing gap in the comprehension and interpretation of metaphors by developers. It is designed to provide guidance for developers to effectively incorporate metaphors into their systems, as well as to facilitate their evaluation and assessment.

SEMIOTICS:

Signs and sign operations, indication, designation, similarity, analogy, metaphor, symbolism, signification, and communication are referred to as semiotics. Figure 4 demonstrates the semiotic model of metaphor.

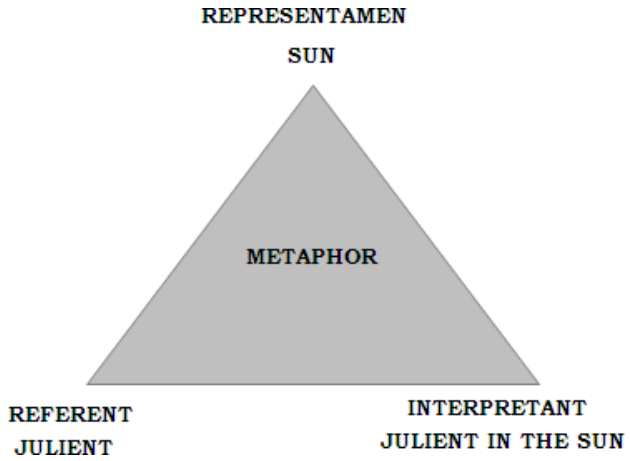

Figure4: Semiotic Model of Metaphor

The depicted image serves to illustrate that the sun functions as a symbol. The referent of the sign is the tenor of the metaphor, and the representa men-referent relationship is characterized by reference. In this particular instance, Juliet serves as the referent, whereby the sun functions as a representation, reference, or symbol for her. The interpretant of a sign can be understood as the complete metaphorical meaning that an observer may

Section A-Research paper

ISSN 2063-5346

derive upon encountering the representation in the referential context. The denotative interpretant is a semiotic concept that denotes implicit meaning within the literal meaning of a sign. According to this interpretation, Juliet is metaphorically represented as the sun.

Semiotics is widely acknowledged to possess considerable anthropological dimensions. The examination of semiotic theories centres on the topic of signs or systems of signs.Semiotics is commonly categorised into three distinct subfields.The correlation between signals and meanings.

Syntactics is the semiotic branch that focuses on the formal characteristics of symbols and signs. The present study aims to investigate the correlation between signs and their impact on the individuals who employ them. The semiosis process is examined in relation to its biotic components.

CONCLUSION

This manuscript delineates the assessment and framework of Extreme Programming (XP) Metaphor, accompanied by a plethora of illustrations. The proposed model presents a clear and semiotics-grounded framework for analysing the structural aspects of system metaphors, thereby offering a foundational explanation for the ways in which metaphors can enhance software systems.The aforementioned study outlines a set of procedures that teams can employ to generate metaphors for their systems, along with methods for assessing the efficacy of such metaphors. The present analysis aims to promote the adoption of system metaphors by Extreme Programming teams as a means to reinforce their development practises, rather than discarding them.

REFERENCES

1. Sune Dueholm Müller., Lars Mathiassen and Hans Henrik Balshøj "Software Process Improvement as organizational change: A metaphorical analysis of the literature" Journal of Systems andSoftware,Vol.83,Issue11,November 2010,pp2128-2146.
2. Ali. A. Nazari Shirehjini "A novel interaction metaphor for personal environment control: direct manipulation of physical environment based on 3Dvisualization"Computers&Graphics,Vol.28,Issue5,October2004, pp667-675
3. DanL.ChiappeandPennyChiappe"Theroleofworkingmemoryinmetaphor production and comprehension" Journal of Memory and Language,Vol.56,Issue2,February 2007,pp172-188.
4. YuLiuandYuetOwyong"Metaphor,multiplicativemeaningandthesemiotic construction of scientific knowledge" LanguageSciences,Vol.33,Issue5,September2011,pp822-834
5. Markus Tendahl and Raymond W. GibbsJr "Complementary perspectives on metaphor: Cognitive linguistics and relevance theory" Journal of Pragmatics, Vol.40,Issue11,November2008,pp 1823-1864.
6. YairNeumanandOphirNave"Metaphor -based meaning excavation" InformationSciences,Vol.179,Issue16, 20July2009,pp2719-2728.
7. Vinoski,S"Doyouknowwhereyour architecture is?" IEEE Internet Computing,Vol.7, Issue5, September 2003, pp 86 -88.
8. Chi-hua Hsiao and Lily I -wen Su "Metaphor and hyperbolic expressions of emotion in Mandarin Chinese conversation" Journal of Pragmatics, Vol.42, Issue5, May2010, pp1380-1396.

Section A-Research paper

ISSN 2063-5346

9. Jiunde Lee "The effects of visual meta phorand cognitivestyle for mental modeling in a hypermedia-based environment" Interacting with Computers, Vol. 19, Issues 5-6 December2007, pp614-629.
10. Diane Pecher, Inge Boot and Saskia Van Dantzig "Sensory -Motor Grounding, Metaphors,andBeyond"PsychologyofLearningandMotivation,Vol.54,2011, pp217-248.
11. SreedharMadhavaram and RobertE. McDonald "Knowledge -based sales managementstrategyandthegraftingmetaphor:Implicationsfortheoryand practice" Industrial Marketing Management,Vol.39,Issue7,October 2010, pp 1078 -1087
12. Narda G. Robinson "Making Sense of the Metaphor: How Acupuncture Works Neurophysiologically" Journal of Equine VeterinaryScience,Vol.29,Issue8,August2009,pp642-644
13. UlrichFlemming.,SureshK.Bhavnani and Bonnie E. John "Mismatched metaphor:uservssystem moeelin computer -aided drafting" Design Studies, Vol.18,Issue4,October1997,pp349-368.
14. MarkJ.Landau.,BrianP.Meierand LucasA.Keefer"AMetaphor -EnrichedSocial Cognition"Psychological Bulletin, Vol.136,Issue6,November2010,pp1045-1067.
15. AliBaykal"Opensystemsmetaphorininstructionaldesign"Procedia -Social and Behavioral Sciences, Vol. 1, Issue 1, 2009, pp 2027 -2031.
16. Dall'Agnol,M.,Janes,A.,Succi,Gand Zaninotto, E"Leanmanagement -A meta phorfor extreme programming?" Lecture Notes in Computer Science, Vol.2675, 2003, pp 26 -32.
17. West,DandSolano,M"Metaphorsbewithyou!(MetaphorSystem)"Proceedings-AGILE Confernce 2005, Vol.2005, Article number1609799,24-29th,July2005,pp3-11.