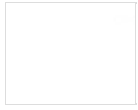


Pattern Language for Human Centered Architectural Evolution



View/Open

 Moore_washington_0250O_18219.pdf (16.99Mb)

Author

Moore, Raeanna

Metadata

[Show full item record](#)

Abstract

All design projects- infrastructure, architecture, furniture- are, by nature, unfinished. Buildings and built spaces are initially designed and constructed to meet a specific set of needs, typically as laid out by a client, but may eventually be adapted to accommodate new programs, occupants, urban environments, or technologies. Unless these things (program, occupancy, cultural requirements) remain static, eventually the building will need to evolve. While the process of construction is linear, the continued use and perceived 'success' of a built project relies on its ability to adapt to the changing needs of its occupants, whether that adaptation happens as a part of the original design or an applied change post construction. Despite the prevailing attitude that once a building is constructed it is 'finished', this thesis argues that post-occupancy adaptations, detailing, and design provisions for future flexible needs are what make a building complete. A well finished space both acknowledges the human scaled needs of its occupants as well as adapting to the programmatic shifts that may take place in the future. While this may not be an argument that pertains to the whole of the built environments, certain spaces, particularly public institutions and educational facilities frequently demonstrate this occurrence. Architecture is not simply a box filled with things, but rather, a structure that is both inhabitable and engaging, flexible and growing, adaptive and responsive to its occupants and environment, growing with generations. Using methods of observation and survey, this thesis examines structures that have undergone post-construction changes in the Seattle area and assess their perceived level of 'completion' as well as the perceived 'success' of the building. By the central argument that all design projects are unfinished, these are simply current iterations and will evolve again in the future, yet they offer lessons on built relationships and adaptive design that can be studied and applied to other, less complete projects. Drawing on the work of Christopher Alexander's Pattern Language, this thesis has developed a set of fifty areas of opportunity for design evolution that focuses on the human occupant. These patterns are not meant as a 'cookbook' of design but rather observations in how buildings speak to their designers and occupants about what they were, what they are, and what they want to be. In understanding that architecture is never truly finished, and that all buildings are simply in line for their next iteration, it suggests a method of crafting a more sustainable future for the way we design.

URI

<http://hdl.handle.net/1773/41678>

Collections

[Architecture \[467\]](#)

DSpace software copyright © 2002-2015 [DuraSpace](#)

[Contact Us](#) | [Send Feedback](#)

Theme by



@MIRE

Human evolution took place as new genetic variations in early ancestor populations favored new abilities to adapt to environmental change and so altered the human way of life. Dr. Rick Potts provides a video short introduction to some of the evidence for human evolution, in the form of fossils and artifacts. Page last updated: January 16, 2019. Introduction to Human Evolution. Lesson Plans. For College Students. Fun Facts. Human Origins Glossary. PATTERN-ORIENTED SOFTWARE ARCHITECTURE On Patterns and Pattern Languages. Volume 5. The book you either are about to read, are reading, or have finished reading—the one in which this foreword is embedded—is about pattern languages largely in this context. Evolution! Thompson's work includes the design of an analog circuit that can discriminate between 1kHz and 10kHz square waves using a fieldprogrammable gate array designed using a genetic algorithm.¹ The result is described as 'probably the most bizarre, mysterious, and unconventional unconstrained evolved circuit yet reported,' in large part because the researchers were unable, in the end, to understand how. Architecture-centric software evolution (ACSE) is considered as an approach to support software adaptation at a controllable level of abstraction in order to survive in the uncertain environment. This requires evolution in system structure and behavior that can be modeled, analyzed and evolved in a formal fashion. Existing research and practices comprise a wide spectrum of evolution-centric approaches in terms of formalisms, methods, processes and frameworks to tackle ACSE as well as empirical studies to consolidate existing research. However, there is no unified framework providing systematic