

Creative Impact of an Event-Driven Visual Programming Tool

Praneet Mathur^{1,2}

¹*Co-Founder, iiterate Technologies GmbH, Adenau, Germany**

²*Founder, ARPM Design and Research LLP, Gurgaon, Haryana, India†*

(Dated: September 15, 2022)

Computational design is gaining global prominence. With the increase in demand for technologically capable designers, we find more designers understanding computers better, learning programming languages and molding technologies to fit their needs. This has led to multidisciplinary communities forming around visual scripting tools (VSTs) like Grasshopper3D, Dynamo, etc. These communities consist of many users from creative fields who find it easier to learn a visual scripting language than a programming language. However, function-driven programming and various quirks of these tools delimit their application to a closed spectrum of use-cases. This further limits the users' capabilities and forces many to hack their way around basic programming language paradigms like loops, event handling, etc.

VSTs seem to promote a creative affinity to programming, while also making it more approachable and accessible. To understand the creative impact of a more powerful VST, this paper outlines the development and use of an agnostic event-driven VST - one based on MVVM software architecture and linked list data structures, written entirely in C# (WPF) with minimal dependencies. With features like plugin extensibility and interoperability with 3D software (e.g. Rhinoceros3D), this new tool is built to aid creative programming driven by events and data. This implies enhanced capabilities for the user and enables interactive computation of data in real-time.

User experience inferences are derived from diverse user studies, with a focus on students and professionals in the design and AEC industries. Various parameters and test scenarios are used to objectively assess the impact of enabling event-driven programming for creative use.

CONTENTS			
I. INTRODUCTION	1	IX. Building Default Libraries	3
A. Related Work	2	X. Interoperability Architecture	3
B. Preliminaries and Definitions	2	A. Local Interop	3
II. Significance of Visual Programming Languages	2	1. Rhinoceros3D Interop	3
III. Significance of Event-Driven Programming	2	B. Network Interop	3
IV. Studying Existing VSTs	2	1. Cloud Applications	3
V. Specifications of an Ideal VPL	2	XI. Headless Computation	4
VI. Specifications of an Ideal VPL-IDE	2	XII. Multithreaded performance	4
VII. VPL Core Architecture	2	XIII. Performance Testing	4
A. Data Structures	2	XIV. User Studies	4
B. Computation	2	XV. Inferences and Conclusions	4
C. IDE Canvas Rendering	2	XVI. Future Work	4
D. Extensibility with Plugins	3	Acknowledgements	4
VIII. Windows Desktop IDE Application	3	APPENDIX 1 – PREPARING A CONTRIBUTION IN LATEX	4
A. OpenFrameworks-based Embedded Infinite Canvas	3	REFERENCES	4
B. WPF-based Embedded Infinite Canvas	3	References	4
1. WPF and Abstraction	3		
C. Packaging Components	3		

I. INTRODUCTION

Here give an introduction to your research.

* praneet@iiterate.de

† praneet@arpmdesignandresearch.com

A. Related Work

Related work should be in two parts - One at the beginning of the paper and one at the end, to help the reader draw meaningful inferences.

B. Preliminaries and Definitions

Your contribution should be preceded by a *short* Abstract of not more than 150 words, written as a single paragraph, as above.

II. SIGNIFICANCE OF VISUAL PROGRAMMING LANGUAGES

Symbols of variables should be typed in math mode. For example, to get the Greek letter alpha, type `\alpha`; it should come out looking like α .

Equations should be numbered consecutively, with the number enclosed in parentheses and placed flush with the right-hand margin.

Here is an example of an equation, The Schroedinger equation for the wave function $\psi(\vec{x}, t)$

$$-\frac{\hbar^2}{2m}\nabla^2\psi + V\psi = E\psi. \quad (1)$$

If you want to refer to the equation, you can the `label`, with (`\ref{label}`). For example, the above equation is (1).

III. SIGNIFICANCE OF EVENT-DRIVEN PROGRAMMING

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

IV. STUDYING EXISTING VSTS

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

V. SPECIFICATIONS OF AN IDEAL VPL

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

VI. SPECIFICATIONS OF AN IDEAL VPL-IDE

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

FIG. 1. Example of a figure; but it's empty until you provide an eps (encapsulated postscript) file. (See the source page.)

VII. VPL CORE ARCHITECTURE

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

A. Data Structures

References should be cited in the text using numbers within square brackets: 'example [1], example [1, 2], example [1–5]', or alternatively as 'Ref. [1], Refs. [1] to [5]', and 'Reference' in full if this word occurs at the beginning of a sentence. They should *appear in consecutive numerical order* and should be listed at the end of the text. Unless you are near the bottom of the last page of text, do *not* start a new page for the list of references, but continue on the same page. Note that in the list of references it is unnecessary to state the title of an article or chapter in proceedings or in a collection of papers unless a page number cannot be quoted, e.g. for future publications.

For example, here's a reference to the paper listed in the bibliography (at the end):

B. Computation

All figures should be quoted in consecutive numerical order in the text and should, for example, be referred to as 'Fig. 3, Figs. 3-5', etc., or 'Figure' at the beginning of a sentence. All figures, diagrams, etc., must remain within the same area.

Figure captions should be brief and, if possible, go *below* the illustration, e.g. 'Fig. 3 A short title'. They should be typed in point size 9. Very detailed illustrations may require a full page; if necessary, they may be placed sideways on the page; when this is done, no text may appear on that page, and the caption must also read sideways.

Where possible, figures should be prepared electronically. We can handle Encapsulated Postscript.

Here is an example of a figure.

C. IDE Canvas Rendering

Tables should be referred to in the text as Table 1, Tables 2–7. They should be centred on the page width, with the table number, followed by a brief caption in point size 9, typed *above* them if possible. For the positioning of

tables, follow the same rules as those for numbered figures.

D. Extensibility with Plugins

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

VIII. WINDOWS DESKTOP IDE APPLICATION

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

A. OpenFrameworks-based Embedded Infinite Canvas

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

B. WPF-based Embedded Infinite Canvas

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

1. WPF and Abstraction

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

C. Packaging Components

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred

to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

IX. BUILDING DEFAULT LIBRARIES

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

X. INTEROPERABILITY ARCHITECTURE

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

A. Local Interop

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

1. *Rhinoceros3D Interop*

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

B. Network Interop

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

1. *Cloud Applications*

These should be laid out as the sections in the text, except that each appendix should start on a new page. They should be numbered consecutively and be referred to as Appendix 1, Appendices 1–3, etc. Equations, figures and tables should be quoted as Eq. (A.1.1) and Fig. A.1.1, etc.

XI. HEADLESS COMPUTATION

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

XII. MULTITHREADED PERFORMANCE

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

XIII. PERFORMANCE TESTING

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

XIV. USER STUDIES

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

XV. INFERENCES AND CONCLUSIONS

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

XVI. FUTURE WORK

If unavoidable, footnotes should be placed at the bottom of the page in which they are referred to.

ACKNOWLEDGEMENTS

If required, acknowledgements should appear as a section immediately before the reference section. The acknowledgements section should not be numbered.

APPENDIX 1 – PREPARING A CONTRIBUTION IN LATEX

We can accept contributions prepared using Latex.

REFERENCES

-
- [1] Neil H. E. Weste and Kamran Eshraghian, *Principles of CMOS VLSI Design*, 2nd ed. Reading, MA: Addison-Wesley, 1993.
 - [2] R. A. Lincoln and K. Yao, "Efficient Systolic Kalman Filtering Design by Dependence Graph Mapping," in *VLSI Signal Processing, III*, IEEE Press, R. W. Brodersen and H. S. Moscovitz Eds., 1988, pp. 396–410.
 - [3] C. H. Bischof and G. M. Shroff, "On Updating Signal Subspaces," *IEEE Trans. on Signal Processing*, vol. 40, no. 1, pp. 96–105, Jan. 1992.