Artefact: A UML-Based Framework for Model-Driven Development of Interactive Surface Prototypes

Mthunzi, Everett Mondliwethu; Echtler, Florian

Published in:
ISS 2021 - Companion Proceedings of the 2021 Conference on Interactive Surfaces and Spaces

DOI (link to publication from Publisher):
10.1145/3447932.3490523

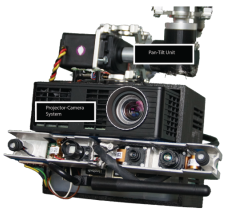
Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

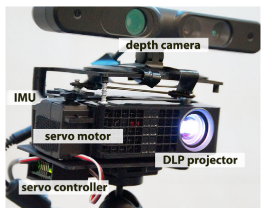
Artefact: A UML-based framework for model-driven development of interactive surface prototypes

Everett M. Mthunzi
mondliwethu.everett.mthunzi@uni-weimar.de
Bauhaus-Universität Weimar
Weimar, Germany

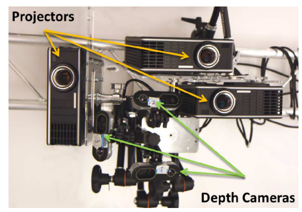
Florian Echtler
floech@cs.aau.dk
Aalborg University
Aalborg, Denmark



(a) Ubeam [12].



(b) AMP-D [35].



(c) LightSpace [32].

(d) Beamatron [33].

Figure 1: Interactive surface prototypes.

ABSTRACT

While interactive surface prototypes may be highly application-specific, existing prototypes hint at common, recurring design considerations. Given the rapid accumulation of near-identical prototypes, there is a need to promote design reuse. In this context, existing research prototypes motivate abstracting generic structures, architectural views, and descriptions to inform future designs. This paper proposes Artefact: a UML-based framework for model-driven development of interactive surface prototypes. We define flexible base models using existing research prototypes: initial hardware and middleware abstractions to support developers in the early design stages. For validation, we use the proposed framework to capture existing research prototypes. We then conduct an interview study to learn expert perceptions towards the captured model representations. Our initial findings highlight three significant benefits: (1) an accessible graphical syntax with unambiguous model representation, (2) a system for capturing arbitrary technical specifications, and (3) flexible model representation with consistent notation. While we can not draw any absolute conclusions, initial results suggest benefits in the model-driven approach.

CCS CONCEPTS

• **Hardware → Design reuse and communication-based design**.

ISS '21 Companion, November 14–17, 2021, Lodz, Poland

ACM ISBN 978-1-4503-8340-0/21/11.
https://doi.org/10.1145/3447932.3490523

KEYWORDS

Interactive surface environments, interactive surface prototypes, UML-based framework

ACM Reference Format:
Everett M. Mthunzi and Florian Echtler. 2021. Artefact: A UML-based framework for model-driven development of interactive surface prototypes. In *Interactive Surfaces and Spaces (ISS '21 Companion), November 14–17, 2021, Lodz, Poland.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3447932.3490523

1 INTRODUCTION

In the research space of interactive surface environments, prototypes are a fundamental contribution. They enable exploring new technologies and are quintessential representations of emerging designs. While interactive surface prototypes are application-specific, there is evidence of recurring design considerations [12, 32, 33, 35]. Given the near-identical implementations and their rapid accumulation, there is a need to promote design reuse. In this context, existing research prototypes enable abstracting generic structures, architectural views, and descriptions to inform future artifact designs. Da Silva and Oliveira suggested adopting UML stereotype-based approaches to promote the reuse of artifact designs. The authors focused on formulating generic abstractions from existing implementations [3]. In [10], Genero et al. emphasized the role of conceptual models in improving domain-specific applications. The authors put forward a literature review (1999—2009), highlighting a need to *"get the model right"* towards improving artifact designs. To learn the general perception of developers toward using UML, Torre et al. conducted a case study. Their findings highlighted widespread adoption of UML-based object-oriented models [28], pointing to rapidly expanding open-source repositories of UML models [7, 14]. Ho-Quang et al. suggested that the rapid adoption of UML-based model-driven approaches to be driven by a need to promote communicating implementations unambiguously. In the same context, Becker and Schäfer underlined lack of modeling support [2].

In this paper, we propose Artefact: a UML-based framework for model-driven development of interactive surface prototypes. We define flexible models based on existing research prototypes: initial hardware and middleware abstractions to support developers in the early design stages. For validation, we express existing research prototypes using the proposed framework. We then conduct an interview study to learn expert perceptions towards the captured model representations proposed model-driven approach.

2 THE ARTEFACT FRAMEWORK

The proposed framework employs UML to establish a flexible, reusable, and expressive syntax for modeling interactive surface prototypes. For flexibility, UML notation enables technology-agnostic representation of abstract components regardless of domain. UML also defines a formal notation for modeling constructs, promoting consistent logical analysis as well as unambiguous model representation. For expressiveness, UML builds on generalization hierarchies [24], and is inherently expressive. A significant benefit to adopting UML is its extensive documentation.

Artefact leverages metaclasses, stereotypes, profiles, and tag values; extension mechanisms that enable extending the normative UML to describe a specific domain [24]. In a broad sense, a Model is an abstract description of a system's environment, i.e., the organization of its members and member associations. As such, for a multi-layered interactive surface prototype, the hardware and middleware systems can be described as complementary models. Metaclasses are abstract Class representations that extend the notion of a conventional UML Class. Profiles are necessary to realize concrete instances of metaclasses using stereotypes. Profiles can also be described as package modules containing stereotypes (i.e., concrete Metaclass extensions). Profiles are reusable, can extend to other profiles, and can be factored for reuse. Within a profile, a Stereotype can extend one or multiple metaclasses, enabling notation in place of and in addition to an extended Metaclass. Note that stereotypes can only be used by extending a pre-defined Metaclass, where extensions indicate that properties of a Metaclass extend to a Stereotype. Akin to a conventional Class, a Stereotype has structural compartments for describing attributes, members, and tag values [24].

3 EMPIRICAL VALIDATION

Given that there is no standard method for determining the absolute correctness of a model [24], for validation, we adopted a two-fold approach. First, the hardware and middleware models were used to capture prototypes presented in [8, 9, 11, 16] and [36]. The criteria for the selected prototypes considered peer-reviewed studies published from 2019 onwards, focusing on current prototypes. Services, formats, transmission rate, vendors, models, modes, quantities, process speeds, sampling rares, and communication interfaces we captured and expressed using tag values, directed relations, cardinalities, stereotype compartments. Once captured, we used the resulting model descriptions for validation. In [27], Sargent suggested evaluating the correctness of a model using *"face validation"*, i.e., subjecting a model to scrutiny of experts. Following this guideline, we conducted an interview study to determine whether experts considered the proposed modeling framework reasonable and acceptable. The study was also used to gain insights about perceptions towards adopting a model-driven approach to support developing interactive surface research prototypes. A call for participation was sent out to research centers at KTH Royal Institute of Technology, the University of Regensburg, and Bauhaus-Universität Weimar. We also invited industry experts from Extend3D GmbH, Munich. Of twelve experts, five met the requirement of at least two years working with interactive surface environments or components associated with interactive surface environments (e.g., projector-camera systems). The interviews were conducted over Skype, and an online form was used to collect expert perceptions. Each interview was structured as follows: First, the research aim was discussed. Then, we asked the experts to confirm the suitability of their experience given the context of the research. Afterward, experts were introduced to the proposed modeling framework. Thereafter, we introduced the formulated hardware and middleware models. After discussing the rationale and objectives of each model, we presented captured model representations of prototypes from [8, 9, 11, 16] and [36]. We asked the experts to scrutinize and remark on the efficacy of using the framework to model prototypes. Lastly, we discussed whether a model-driven approach would benefit the research community for developing prototypes in the early design stages. Discussion with all experts was structured using questionnaires. All questions were open-ended towards deeper discussions, as led by the experts. This approach promoted collecting experts' perceptions and learning their opinions on setbacks, benefits, and implications of a model-driven approach using the proposed framework.

Expert #1 remarked on the rationality of the models, "The models are rational and clear.". **Expert #2** gave merit to unambiguous technical outlines, "…specifications simplify prototyping.". **Expert #3** pointed to capitalizing existing prototypes, "…developers can obtain structured information about existing prototypes…". **Expert #4** remarked on learnability, "…it took me ~5 minutes to grasp the framework.". **Expert #5** underlined inherent fostering of artifact replication, "…full specifications simplify replicating prototypes.".

Data collected during the interviews was coded and analyzed by the first author. Our findings suggest that experts found the proposed framework to be beneficial with perceptions converging toward three main benefits: (1) a generic and simple syntax for prototyping, (2) an approach to systematically compare different prototypes, and (3) a convenient starting point for developing interactive surface prototypes.

4 DISCUSSION

Our validation approach suffers from the absence of a standard methodology for evaluating the *"absolute"* validity of models [24, 27]. Although an argument can be made for the simple representation of structurally complex systems, the modeling framework we propose is not comprehensive by design. Absent an extensive set of all possible abstractions, the minimal set presented in this paper is insufficient for non-experts who may also seek to employ the modeling framework as a knowledge base. Face validation has demonstrated the role of experts in the iterative development of

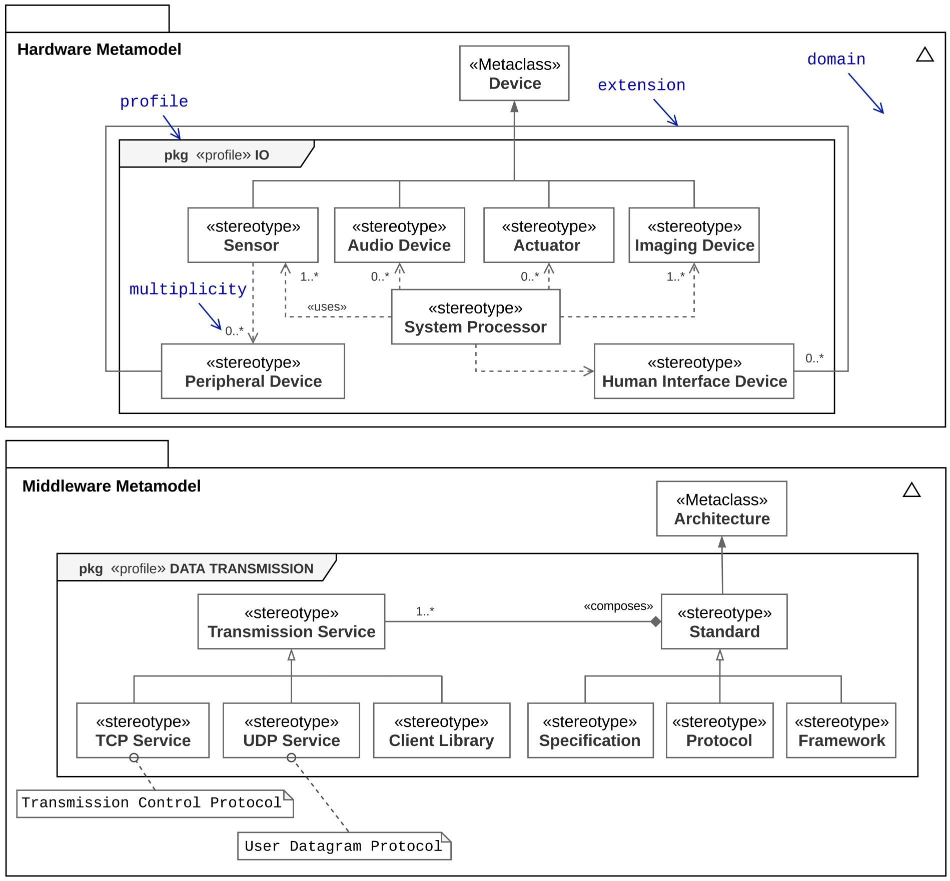


Figure 2: Flexible base models for the hardware and middleware layers (abstracted in Tab. 1). Profiles and metaclasses are used to describe a domain's environment. Properties and tagged values are utilized to convey arbitrary technical specifications. Last, directed relationships and cardinality are used to capture dependencies. Each model is suggestible and can be modified for variable application scenarios.

the modeling infrastructure. There is much promise in conducting face validation on a larger scale and learning expert perceptions towards *"models as end products"* [1], i.e., in the space of interactive surface environments. Suggestions from experts point to benefit in open discourse about design considerations for the modeling framework. Conducting workshops with experts and developers would be one possible approach to promote such a discussion.

5 CONCLUSION

This paper identifies the accelerated accumulation of comparable prototypes and proposes a model-driven approach to promote design reuse. Existing prototypes have been leveraged to define a generic UML-based framework for modeling hardware and middleware layers of interactive surface prototypes. The proposed framework has been applied to capture existing prototypes, and a face validation study has been conducted to learn experts' perceptions towards the captured model representations. Our initial findings highlight three significant benefits: (1) an accessible graphical syntax with unambiguous model representation, (2) a system for capturing arbitrary technical specifications, and (3) flexible model representation with consistent notation. While no absolute conclusions can be drawn, initial results suggest benefits in the proposed framework.

ACKNOWLEDGMENTS

This paper was supported by the German Federal Ministry for Education and Research (BMBF) through grant 16SV8288 as part of the VIGITIA project.

REFERENCES

[1] C Atkinson and T Kuhne. 2003. Model-driven development: a metamodeling foundation. *IEEE Software* 20, 5 (2003), 36–41. https://doi.org/10.1109/MS.2003.1231149

[2] Martin Becker and Andreas Schäfer. 2020. Variability Realization in UML/SysML Models. In *Proceedings of the 24th ACM Conference on Systems and Software Product Line*. ACM, New York, NY, USA, 1. https://doi.org/10.1145/3382025.3414975

[3] Leandro Flores Da Silva and Edson Oliveira. 2020. Evaluating Usefulness, Ease of Use and Usability of an UML-Based Software Product Line Tool. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*. ACM, New York, NY, USA, 798–807. https://doi.org/10.1145/3422392.3422402

[4] Florian Echtler. 2018. Surfacestreams: A content-agnostic streaming toolkit for interactive surfaces. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 10–12. https://doi.org/10.1145/3266037.3266085

Table 1: Abstraction of hardware and middleware modeling constructs. Components identified in reviewed body of literature are generalized to stereotypes. The stereotypes are then abstracted to high-level metaclasses.

Metaclass	Stereotype	Component	Body of literature
Device	System Processor	Workstation, Server	[4–6, 12, 13, 21–23, 25, 26, 29–35]
		Microprocessor	[13, 25]
Device	Sensor	Camera Sensor	[4, 5, 12, 13, 21, 23, 25, 26, 29, 30, 33–35, 37]
		Depth Sensor	[4, 5, 13, 23, 25, 31–33, 35, 37]
		Infrared Sensor	[22, 29]
Device	Imaging Device	Projector, Display	[4, 6, 12, 13, 22, 23, 25, 26, 29, 30, 32–35, 37]
Device	Actuator	Actuator	[12, 22, 26, 29, 33, 35]
Device	Human Interface Device	Mobile Device	[13, 21, 23, 25, 26, 30, 33, 35]
Device	Peripheral Device	Phicon	[29]
		Marker	[21, 22, 26]
		Mirror	[6, 22, 29]
Device	Audio Device	Transducer	[6]
Architecture	Transmission Service	Framework	[4, 19]
		Specification/Protocol	[17, 18, 20]

[5] Florian Echtler and Raphael Wimmer. 2014. The interactive dining table, or pass the weather widget, please. In *Proceedings of the 9th ACM International Conference on Interactive Tabletops and Surfaces*. ACM, New York, NY, USA, 419–422. https://doi.org/10.1145/2669485.2669525

[6] Senem Ezgi Emgin, Amirreza Aghakhani, T. Metin Sezgin, and Cagatay Basdogan. 2019. HapTable: An interactive tabletop providing online haptic feedback for touch gestures. *IEEE Transactions on Visualization and Computer Graphics* 25, 9 (2019), 2749–2762. https://doi.org/10.1109/TVCG.2018.2855154

[7] Robert B France, James M Bieman, Sai Pradeep Mandalaparty, Betty H C Cheng, and Adam Jensen. 2012. Repository for Model Driven Development (ReMoDD). In *2012 34th International Conference on Software Engineering (ICSE)*. IEEE, Zurich, Switzerland, 1471–1472. https://doi.org/10.1109/ICSE.2012.6227059

[8] Eisuke Fujinawa, Kenji Goto, Atsushi Irie, Songtao Wu, and Kuanhong Xu. 2019. Occlusion-aware hand posture based interaction on tabletop projector. In *Proceedings of the 2nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans, LA, USA, 113–115. https://doi.org/10.1145/3332167.3356890

[9] Shun Fujita, Naoki Yanagihara, Buntarou Shizuki, and Shin Takahashi. 2019. Ray-casting based interaction using an extended pull-out gesture for interactive tabletops. In *Proceedings of the 31st Australian Conference on Human-Computer-Interaction*. ACM, New York, NY, USA, 546–549. https://doi.org/10.1145/3369457.3369528

[10] Marcela Genero, Ana Fernández-Sáez, H Nelson, Geert Poels, and Mario Piattini. 2011. A Systematic Literature Review on the Quality of UML Models. *J. Database Manag.* 22 (2011), 46–70. https://doi.org/10.4018/jdm.2011070103

[11] Yizheng Gu, Chun Yu, Zhipeng Li, Weiqi Li, Shuchang Xu, Xiaoying Wei, and Yuanchun Shi. 2019. Accurate and low-latency sensing of touch contact on any surface with finger-worn IMU sensor. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans, LA, USA, 1059–1070. https://doi.org/10.1145/3332165.3347947

[12] Jan Gugenheimer, Enrico Rukzio, Pascal Knierim, and Julian Seifert. 2014. UbiBeam: An interactive projector-camera system for domestic deployment. In *Proceedings of the 9th ACM International Conference on Interactive Tabletops and Surfaces*. ACM, New York, NY, USA, 305–310. https://doi.org/10.1145/2669485.2669537

[13] Chris Harrison, Hrvoje Benko, and Andrew Wilson. 2011. OmniTouch : Wearable multi-touch interaction everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 441–450. https://doi.org/10.1145/2047196.2047255

[14] Regina Hebig, Truong Ho Quang, Michel R V Chaudron, Gregorio Robles, and Miguel Angel Fernandez. 2016. The Quest for Open Source Projects That Use UML: Mining GitHub. In *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*. ACM, New York, NY, USA, 173–183. https://doi.org/10.1145/2976767.2976778

[15] Truong Ho-Quang, Regina Hebig, Gregorio Robles, Michel R V Chaudron, and Miguel Angel Fernandez. 2017. Practices and Perceptions of UML Use in Open Source Projects. In *Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Practice Track*. IEEE, Buenos Aires, Argentina, 203–212. https://doi.org/10.1109/ICSE-SEIP.2017.28

[16] Nikhita Joshi and Daniel Vogel. 2019. An evaluation of touch input at the edge of a table. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 2865–2874. https://doi.org/10.1145/3290605.3300476

[17] Martin Kaltenbrunner. 2009. ReacTIVision and TUIO: A Tangible Tabletop Toolkit. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. ACM, New York, NY, USA, 9–16. https://doi.org/10.1145/1731903.1731906

[18] Martin Kaltenbrunner, Till Bovermann, Ross Bencina, and Enrico Costanza. 2005. TUIO: A protocol for table-top tangible user interfaces. In *Proceedings of the 6th international conference on Gesture in Human-Computer Interaction and Simulation*. ACM, Berder Island, France, 1–5.

[19] Martin Kaltenbrunner and Florian Echtler. 2015. Shared Infrastructures for Tangible Tabletops & Interactive Surfaces. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*. ACM, New York, NY, USA, 499–500. https://doi.org/10.1145/2817721.2835070

[20] Martin Kaltenbrunner and Florian Echtler. 2018. The TUIO 2.0 protocol: An abstraction framework for tangible interactive surfaces. *Proceedings of the ACM on Human-Computer Interaction* 2, EICS (jun 2018), 35. https://doi.org/10.1145/3229090

[21] Hirokazu Kato, Mark Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. 2000. Virtual object manipulation on a table-top AR environment. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*. IEEE, New York, NY, USA, 111–119. https://doi.org/10.1109/ISAR.2000.880934

[22] Hideki Koike, Yoshinori Kobayashi, and Yoichi Sato. 2001. Integrating paper and digital Information on EnhancedDesk: A method for realtime finger tracking on an augmented desk system. In *ACM Transactions on Computer-Human Interaction*, Vol. 8. ACM, New York, NY, USA, 307–322. https://doi.org/10.1145/504704.504706

[23] Sundar Murugappan, Vinayak, Niklas Elmqvist, and Karthik Ramani. 2012. Extended Multitouch: Recovering touch posture and differentiating users using a depth camera. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 487–496. https://doi.org/10.1145/2380116.2380177

[24] Object Management Group. 2017. A specification defining a graphical language for visualizing, specifying, constructing, and documenting the artifacts of distributed object systems. https://www.omg.org/spec/UML/About-UML/

[25] Yoon Jung Park, Hyocheol Ro, Jung-Hyun Byun, and Tack-Don Han. 2019. Adaptive Projection Augmented Reality with Object Recognition Based on Deep Learning. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*. ACM, New York, NY, USA, 51–52. https://doi.org/10.1145/3308557.3308678

[26] Jun Rekimoto and Masanori Saitoh. 1999. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York,

NY, USA, 378–385. https://doi.org/10.1145/302979.303113

[27] R G Sargent. 2010. Verification and validation of simulation models. In *Proceedings of the 2010 Winter Simulation Conference*. IEEE, Baltimore, MD, USA, 166–183. https://doi.org/10.1109/WSC.2010.5679166

[28] Damiano Torre, Yvan Labiche, Marcela Genero, Maged Elaasar, and Claudio Menghi. 2020. UML Consistency Rules: A Case Study with Open-Source UML Models. In *Proceedings of the 8th International Conference on Formal Methods in Software Engineering*. ACM, New York, NY, USA, 130–140. https://doi.org/10.1145/3372020.3391554

[29] Brygg Ullmer and Hiroshi Ishii. 1998. The metaDESK: Models and prototypes for tangible user interfaces. *Proceedings of the 10th annual ACM Symposium on User Interface Software and Technology* 8 (1998), 223–232. https://doi.org/10.1145/263407.263551

[30] Andrew D. Wilson. 2005. PlayAnywhere: A Compact Interactive Tabletop Projection-Vision System. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 83–92. https://doi.org/10.1145/1095034.1095047

[31] Andrew D. Wilson. 2010. Using a Depth Camera as a Touch Sensor. In *ACM International Conference on Interactive Tabletops and Surfaces, ITS 2010*. ACM, New York, NY, USA, 69–72. https://doi.org/10.1145/1936652.1936665

[32] Andrew D. Wilson and Hrvoje Benko. 2010. Combining multiple depth cameras and projectors for interactions on, above, and between surfaces. In *23rd ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 273–282. https://doi.org/10.1145/1866029.1866073

[33] Andrew D. Wilson, Hrvoje Benko, Shahram Izadi, and Otmar Hilliges. 2012. Steerable augmented reality with the Beamatron. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 413–422. https://doi.org/10.1145/2380116.2380169

[34] Benjamin Wingert, Isabel Schöllhorn, and Matthias Bues. 2017. Prodesk: An interactive ubiquitous desktop surface. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. ACM, New York, NY, USA, 366–371. https://doi.org/10.1145/3132272.3135078

[35] Christian Winkler, Julian Seifert, David Dobbelstein, and Enrico Rukzio. 2014. Pervasive information through constant personal projection: The ambient mobile pervasive display (AMP-D). In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 4117–4126. https://doi.org/10.1145/2556288.2557365

[36] Qin Wu, Jiayuan Wang, Sirui Wang, Tong Su, and Chenmei Yu. 2019. MagicPAPER: Tabletop interactive projection device based on tangible interaction. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*. ACM, Los Angeles, CA, USA, 2. https://doi.org/10.1145/3306214.3338575

[37] Robert Xiao, Scott Hudson, and Chris Harrison. 2017. Supporting Responsive Cohabitation Between Virtual Interfaces and Physical Objects on Everyday Surfaces. *Proceedings of the ACM on Human-Computer Interaction* 1, EICS (jun 2017), 17. https://doi.org/10.1145/3095814