A Unified Design & Development Framework for Mixed Interactive Systems

Guillaume Bataille, Valérie Gouranton, Jérémy Lacoche, Danielle Pelé, Bruno Arnaldi

▶ **To cite this version:**

Guillaume Bataille, Valérie Gouranton, Jérémy Lacoche, Danielle Pelé, Bruno Arnaldi. A Unified Design & Development Framework for Mixed Interactive Systems. VISIGRAPP 2020 - 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Feb 2020, Valletta, Malta. pp.1-12. hal-02502967

HAL Id: hal-02502967
https://hal.science/hal-02502967

Submitted on 9 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Unified Design & Development Framework for Mixed Interactive Systems

Guillaume Bataille[1,2][a], Valérie Gouranton[2], Jérémy Lacoche[1], Danielle Pelé[1] and Bruno Arnaldi[2]
[1]*Orange Labs, Cesson Sévigné, France*
[2]*Univ Rennes, INSA Rennes, Inria, CNRS, IRISA, France*

Keywords: Human-Machine Interaction, Mixed Reality, Natural User Interfaces, Internet of Things, Cyber-Physical Systems, Hybrid Interactive Systems, Mixed Interactive Systems.

Abstract: Mixed reality, natural user interfaces and the internet of things are complementary computing paradigms. They converge towards news form of interactive systems named mixed interactive systems. Because of their exploding complexity, mixed interactive systems induce new challenges for designers and developers. We need new abstractions of these systems in order to describe their real-virtual interplay. We also need to break mixed interactive systems down into pieces in order to segment their complexity into comprehensible subsystems. This paper presents a framework to enhance the design and development of these systems. We propose a model unifying the paradigms of mixed reality, natural user interfaces and the internet of things. Our model decomposes a mixed interactive system into a graph of mixed entities. Our framework implements this model, which facilitates interactions between users, mixed reality devices and connected objects. In order to demonstrate our approach, we present how designers and developers can use this framework to develop a mixed interactive system dedicated to smart building occupants.

1 INTRODUCTION

In this paper, we aim to enhance the design and development of interactive systems blending the paradigms of mixed reality (MR), natural user interfaces (NUI) and the internet of things (IoT). These systems are called *mixed interactive systems* (MIS) (Dubois et al., 2010). They offer new ways for humans to interact with their environment. MR synchronizes real and virtual worlds and permits to simultaneously interact with them. NUI use the human body as an interface in order to grant natural and intuitive interactions with digital technologies. The IoT supports and improves ubiquitous interactions with everyday things. These paradigms are converging in order to bridge the gap between real and virtual. NUI combined with the IoT are common in our everyday lives. As an example, voice assistants are becoming mainstream products. Combined with MR, NUI mediate our indirect interactions with real entities. For example, we can control a virtual brush by gesture in order to control a painting robot. Beyond the coupling of NUI and the IoT, MR can inform us about the mixed objects surrounding us and mediate our interactions with the IoT. MR could inform us of a broken part of a smart car, how to repair it and help us to order the required spare parts and tools that we are missing.

Blending these domains is still in its infancy and limited to prototypes and proofs of concept. Current techniques like tracking or object recognition bound this convergence. However, the limitations of this convergence are also due to a lack of dedicated tools. We need to drive the production of MIS from handmade prototypes towards industrial processes. As a contribution, we present in this paper a framework to design and develop MIS. We base this framework on our unified model that we name as the design-oriented mixed-reality internal model (DOMIM). Our DOMIM-based framework offers design and development tools to create these systems. Its benefits are simplicity, reusability, flexibility, and swiftness.

In the next section, we present MR, NUI and the IoT, and work related to hybrid interactions blending these domains. The third section presents our unified model and our resulting framework based on it. The fourth section describes the development of a scenario with our framework to validate our approach. This use case is an MIS for smart building occupants. The last section concludes this paper.

[a] https://orcid.org/0000-0002-6751-3914

2 RELATED WORK

In this section, we first define MR, NUI, and the IoT as research domains. Then we present work related to hybrid interactive systems blending these domains.

MR, NUI and the IoT

Mixed Reality (Milgram and Kishino, 1994) is provided for Coutrix et al. by mixed reality systems composed of mixed objects (Coutrix and Nigay, 2006). A mixed object is composed of a real part described by its physical properties, and a digital part described by its digital properties. A loop synchronizes both parts, as presented in figure 1. Current technologies bound this synchronization. They restrict the capture of physical properties and the actuation of digital properties.

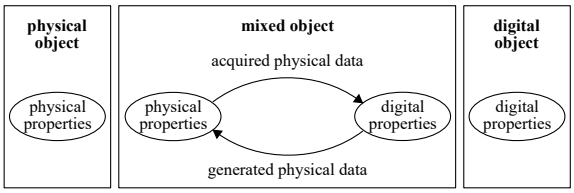

Figure 1: Based on (Coutrix and Nigay, 2006) this figure presents how mixed objects synchronize both their real and virtual properties.

Natural User Interfaces exploit the human body as an interface. They address skills that humans acquired while interacting with the real world (Blake, 2012). NUI mediate interactions between humans and the virtual by mimicking interactions between humans and their real environment. Sensors track human activity, while actuators render virtual properties (Liu, 2010). The main interaction modalities used by NUI are vision, voice recognition and synthesis, touch, haptics, or body motion. For example, users can command voice assistants by speech in order to listen to music, or manipulate a virtual object by captured gestures. NUI provide what we call in this paper pseudo-natural interactions since they partially succeed in reproducing the interactions between humans and the real world (Norman, 2010). A key factor of a NUI success is its short learning curve due to its intuitiveness.

The *Internet of Things* is defined by Atzori et al. (Atzori et al., 2017) as "a conceptual framework that leverages on the availability of heterogeneous devices and interconnection solutions, as well as augmented physical objects providing a shared information base on global scale, to support the design of applications involving at the same virtual level both people and representations of objects.". The IoT connects the physical and digital worlds and transduces their properties in order to interact with humans (Greer et al., 2019). Diverse network services and protocols, like the Zwave protocol, enable and normalize their network communications. Grieves introduced in 2002 the digital twin concept in the smart and connected product systems context, as part of the internet of things (Grieves, 2019). A digital twin is a digital clone of a real product, its physical twin. Both twins are interconnected (Glaessgen and Stargel, 2012) in order to provide supervision and control of the physical twin to manufacturers and users(Grieves, 2019).

Hybrid Interactions

Researchers have created hybrid interactions combining MR, NUI, or the IoT for over a decade. In related work we found several non-reusable prototypes melting MR, NUI, and the IoT. We also found models and frameworks related to MIS. The complexity of MIS requires the use of models and tools in order to support their design and development.

Non-reusable Prototypes

Petersen et al. (Petersen and Stricker, 2009) studied the continuous synchronization of virtual and real by using NUI. Mistry et al. developed Sixthsense (Mistry and Maes, 2009) to augment the real world with digital content. They use gestures to interact with the content. Rendering is provided by spatial augmented reality (SAR). Sulisz et al. combined AR and smart devices to support mobile users' interactions (Sulisz and Seeling, 2012). Lin et al. developed Ubii to pseudo-naturally interact with real objects in AR (Lin et al., 2017). Using GUI limits the naturalness of their interactions. Their main interest is to allow distant interactions between devices in the user neighborhood. These interactions require scanning the RFID tag of objects on a short-range. This constraint also limits the naturalness of these interactions. Kritzler et al. extended the concept of digital twins to virtual twins as interactive renderings of industrial objects in order to control smart factories (Kritzler et al., 2017). Normand et al. (Normand and McGuffin, 2018) enlarged a smartphone screen with a video see-through device. This device renders around a smartphone its virtual properties. It also allows pseudo-natural interactions by in-air gestures with these properties. Mann et al. presented the concept of All Reality (Mann et al., 2018) which blends mixed reality and sensors in order to augment humans-objects interaction. Kim et al. studied user interactions with a voice assistant mediated by an embodied virtual agent (Kim et al., 2018). Norouzi et al. (Norouzi et al., 2019) surveyed the merging between autonomous agents, the IoT and

Table 1: Comparison between interactive systems blending MR, NUI or the IoT under the prism of models and frameworks.

Work	real/virtual synchronization	real/virtual decorrelation	MR devices templates	natural interactions library	flexibility
MacWilliams et al. 2003	yes	yes	no	no	static
Kelaidonis et al. 2012	yes	yes	no	no	dynamic
Dubois et al. 2014	yes	yes	no	no	dynamic
Nitti et al. 2016	yes	yes	no	no	no
Bouzekri et al. 2018	yes	no	no	no	no
Pfeiffer et al. 2018	simulated	simulated	no	no	no
Lacoche et al. 2019	yes	yes	no	no	static

augmented reality (AR). This is an anthropomorphic trend of the convergence between MR, NUI, and the IoT. But virtual agents are - in these cases - virtual mediators. They do not complete the mediated object itself with a rendered virtual part. These works were also handmade and produced non-reusable prototypes.

Models and Frameworks

Multi-agent models aim at factoring a system into a collection of components: the agents. MVC (Model-View-Control) and PAC (Presentation-Abstraction-Control) separate the functional and visual/interaction features of interactive systems (Hussey and Carrington, 1997). However, they both fail in describing low-level aspects of human-machine interaction and in decorrelating real and virtual.

MIS are mainly studied under the convergence of ubiquitous computing, tangible user interfaces, and MR. MacWilliams et al. (MacWilliams et al., 2003) presented a development framework for interactive systems mixing AR, ubiquitous and wearable computing and tangible user interface called DWARF. While this system did not provide a generic model unifying all these domains through the real/virtual interplay, software templates or a natural interactions library, it provided tools to support the design and development phase of an MIS. Jacob et al. (Jacob et al., 2008) presented a set of thematic guidelines in order to study MIS interactions but no generic architecture or development tools dedicated to MIS. Dubois et al. (Dubois et al., 2014) presented a model-based MIS development framework. However, this framework does not consider all entities composing an MIS as mixed entities or the interplay of mixed entities interacting in both real and virtual worlds. Kelaidonis et al. (Kelaidonis et al., 2012) presented a framework to virtualize real objects and manage the interplay with their virtual part in the IoT context. Nitti et al. surveyed virtual objects in the IoT (Nitti et al., 2016) and presented a generic virtualization architecture. However, they do not cover natural user interactions. Bouzekri et al. presented a generic architecture for cyber-physical systems (Bouzekri et al., 2018) which considers users, interfaces, communications, software including behavioral models and hardware. They provide an architecture that may be adapted to MR, NUI and the IoT but does not precisely model behaviors or decorrelate real and virtual. Pfeiffer et al. presented their approach to design in VR mixed interactive systems (Pfeiffer and Pfeiffer-Leßmann, 2018). But their approach is limited to the design of simulated MIS systems. They do not provide any model to easily deploy the simulation results in a real environment. Lacoche et al. presented a PAC model to simulate and develop smart environments (Lacoche et al., 2019). But this model does not cover natural user interactions.

The Table 1 compares related work relative to interactive systems mixing MR, NUI or the IoT and their contribution to unified models or frameworks.

Related Work Analysis

In conclusion, merging MR, NUI and the IoT in these works simplifies user interactions. But they implement non-sustainable software in order to develop proofs of concept and prototypes mixing these domains. This software has a short lifetime with no reusability, interoperability or potential support.

However, the complexity of MIS design and development requires models and tools to support them. Existing models, architectures and frameworks may cover MIS design and development but do not satisfy designers and developers' need for generic real/virtual synchronization and decorrelation, software templates and natural interaction libraries. Tools would help to design and develop MIS. However, new tools require new models.

In this paper, we present our solution to these problems.

3 THE DESIGN-ORIENTED MIXED-REALITY INTERNAL MODEL

A growing amount of interactive devices and connected objects, the expanding volume of data to communicate, and a more accurate synchronization between real and virtual contribute to the increasing complexity of mixed interactive systems (MIS). We need to create abstractions, methods, and tools to support MIS developments. And we require new models to produce these tools. Our approach consists in providing a common model for NUI, MR, and the IoT. This model supports an MIS design from its architectural design to its implementation. We, therefore, propose to name this model the design-oriented mixed-reality internal model (DOMIM). This model is implemented into a framework ready to use by developers. The resulting framework splits an MIS into reusable application templates and components, providing simplicity, reusability, flexibility, and swiftness. We now describe our mixed entity model step by step, then the relations between mixed entities, and finally our framework based on this model.

3.1 Mixed Entities

MR, NUI and the IoT all mix virtual and real. MR blends real and virtual from a user-centric point of view. The IoT allows ubiquitous interactions between humans and things through networks. NUI mediate interactions between humans and virtual entities. We need to define a transparent relation between the real and the virtual, suitable for all MR, NUI, and the IoT. In order to define such a relation, we need a unified model for these domains. We first extend the Coutrix et al. (Coutrix and Nigay, 2006) model of a mixed object, as shown in Figure 1, to both living and non-living entities. This allows us to model users, interfaces, and objects with a single model. Figure 2 describes the mixed entity model. This model is useful to decorrelate and synchronize the virtual and the real properties of a mixed entity. This decorrelation allows to observe or control how these properties evolve separately when they are not synchronized and complementarily when they are. Their synchronization requires sensors, actuators, and infrastructures to support and communicate virtual properties. A real and a virtual part compose mixed entities. A loop synchronizes these parts. This loop translates a subset of the real properties of an entity into virtual ones and vice versa. As an example, we consider humans behaving in a mixed building. The real building itself is its real part. Interconnected sensors like cameras, actuators like smart plugs, and controllers like computers compose the real interaction loop of this mixed building. The virtual twin of a mixed building is its virtual part. This virtual part mirrors the real presence of objects and humans in the building. Turning on the virtual twin of a mixed fan will turn on the real fan.

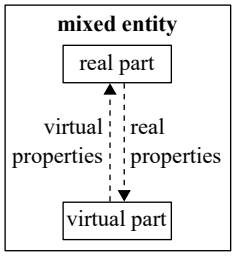

Figure 2: The mixed entity external model synchronizes the properties of the real and virtual parts of a mixed entity.

Completing the Mixed Entity Model

DOMIM combines the mixed entity model with the perception-cognition-action loop, in order to model the capability of each part to behave. Mixed entities can behave in both real and virtual environments. Typically, smart objects (Sánchez López et al., 2012)(Poslad, 2011) synchronize their virtual and real behaviors in real-time. We present our resulting model in Figure 3. The internal interactions of a mixed entity occur:

- inside each part of a mixed entity. Sensors, actuators, and controllers capture, transmit and process the properties of the mixed entity,
- between the parts of a mixed entity. They synchronize and complete their knowledge and behavior in order to generate a mixed behavior.

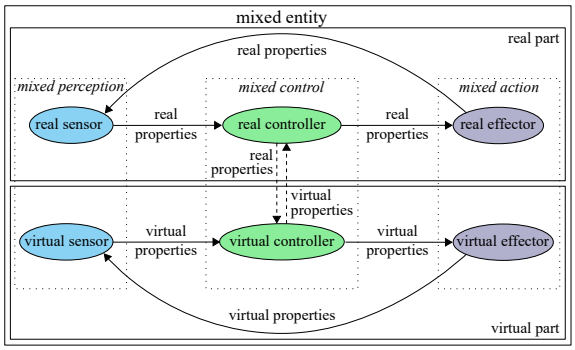

Figure 3: Our design-oriented mixed-reality internal model of a mixed entity, composed of two synchronized interactions loops.

Real sensors capture the properties of the real world, for example, cameras. Real controllers execute applications, in order to process the real properties captured. For example, a real controller can rely on an

algorithm that would detect the location of the mixed entity. Real effectors act on real properties. Typically, a smart plug can switch on/off a lamp. Virtual sensors capture virtual properties. For example, a real presence sensor perceives humans in its neighborhood, while a virtual one perceives humanoid presence in its virtual neighborhood. Virtual controllers process virtual properties, like an algorithm in charge of processing the semantics of a received message. Virtual effectors modify virtual properties. For example, a virtual brush can paint a virtual space. When developed, these components are added to the DOMIM application templates in order to reuse them.

Mixed entities may self-support the communication between their real and virtual parts. They are in this case *active mixed entities*. This is the case of vacuum robots. These robots can perceive their environment with sensors and reconstruct it in 3D with simultaneous location and mapping (SLAM) algorithms (Leonard and Durrant-Whyte, 1991)(Marchand et al., 2016). *Passive mixed entities* need environmental sensors and controllers in order to maintain their virtual part. They cannot synchronize their real and virtual parts without the IoT. Mixed humans hybridize the notions of avatar and agent. Their avatar clones their skeleton movement and position. When non-synchronized, the virtual part of mixed humans can behave as an autonomous agent. For example, the virtual part of a sleeping person can turn off a TV. The virtual part of mixed users is supported either by the NUI they use or their connected environment, in order to interact with their environment. The real part of mixed users can also interact with the real part of their environment.

DOMIM enables the coexistence between real and virtual. With DOMIM, the real can affect the virtual and vice versa. In the next subsection, we describe the relations between the mixed entities constituting an MIS.

3.2 Interactions between Mixed Entities

The user-interface-environment interaction graph described in Figure 4 shows our approach to the interactions occurring between the entities of an MIS. Mixed entities interact externally through natural, pseudo-natural and virtual world interactions in such systems, as shown in Figure 5:

- between the real part of mixed entities: natural, pseudo-natural or network interactions,
- between the virtual part of mixed entities: virtual world interactions.

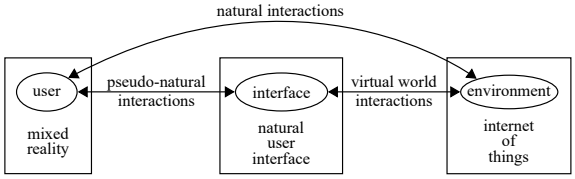

Figure 4: Mixed Reality, Natural User Interfaces and the Internet of Things interaction graph presents the different types of interaction involved and how they complete each other in MIS.

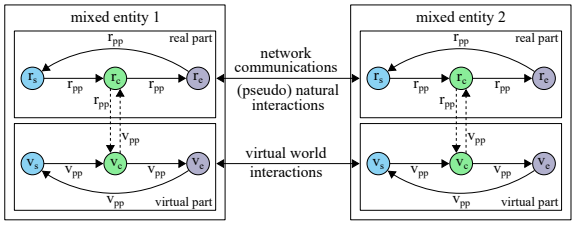

Figure 5: External and internal interactions in a system of two mixed entities.

DOMIM allows designers and developers to:

- separate the constitutive entities of an MIS: this is useful to design and architect an MIS,
- decorrelate the real and the virtual in order to design and implement their complementary interactions,

DOMIM can support a simple modeling of mixed humans, NUI, and connected objects and their interactions as shown in section (4). Compared to reference interaction models like MVC or PAC, DOMIM allows describing low-level aspects of interactions. The resulting model is flexible since it allows to easily reuse and adapt previous models and their implementation to other mixed entities. It is used by our framework while designing, architecting and implementing MIS. It is useful to break an MIS into pieces in order to segment its complexity.

We now describe the DOMIM framework.

3.3 The DOMIM Framework

As previously mentioned, an MIS interconnects a pool of mixed entities and can be considered as a distributed user interface. These interfaces allow users to interact naturally and intuitively with the mixed entities composing it. To produce such interfaces, our DOMIM-based framework supports the development of interconnected applications enabling the mixed entities composing an MIS. Interconnected platforms linked to devices by services constitute an MIS. Platforms like the Hololens run these applications. Each application implements one or more mixed objects. An MIS typically requires connected objects man-

agement services. For example, one service controls all the Zwave-compatible devices of an MIS. This service can collect sensor events from the connected devices and controls their actuators. It allows synchronizing each application supporting a mixed entity with its real part through the Zwave protocol. Simultaneously, a Hololens application supports both the mixed user and the Hololens as a natural user interface, a mixed interface.

Our framework requires Unity[1]. Extending supported development environments to the Unreal Engine or even an entirely dedicated development environment represents an important development cost but is still feasible in order to provide a completely independent solution to developers. Our framework provides DOMIM-based application templates for Windows and MR devices like Hololens or Android smartphones supporting ARCore. These templates are Unity 2018.3 projects containing:

- a Petri net editor. Our plugin integrates this editor to Unity (Bouville et al., 2015)(Claude et al., 2015). Developers easily complete a pre-defined Petri net by connecting the scene graph entities and their components to it. It allows to design software during implementation and is simpler than coding, noticeably for designers,
- a generic implementation of the DOMIM model as an editable Petri net. Developers complete this Petri net in order to configure a mixed entity behavior by clicking on buttons and drop-down lists,
- pre-developed network and natural interaction components. Developers easily associate these components with the mixed entities composing the scene graph thanks to our Petri net editor.

Our DOMIM-based application templates provide components shown by Figure 6. We provide pre-developed gesture recognition components like gaze and tap gesture, the tangible touch of virtual objects, and hybrid tangible, in-air and tactile interactions with virtual objects. Our network synchronization components enable the synchronization of mixed entities' properties. Mixed entities are interconnected in order to produce distributed interactions or synchronize their location. We provide a Zwave service and based on the Zwave .NET library [2]. Other IoT services are under development, as a service based on the LoRa protocol. The semantic analysis component parses the network messages received by a DOMIM-based application. The semantic description component formats the network messages sent by the application, typically to synchronize and distribute the MIS interface. For example, a mixed object and a mixed user need to synchronize their location, in order to interact in the virtual world. The semantic components are crucial to normalize the network communications between the mixed entities composing an MIS. This normalization allows interconnectivity between MIS and consequently the scalability of DOMIM-based MIS. The computer vision component integrates transparently the Vuforia engine[3] or an alternative AR middleware like ARToolKit[4] to our interaction and colocation components. The Vuforia engine is an AR development platform, providing vision-based pose estimation algorithms.

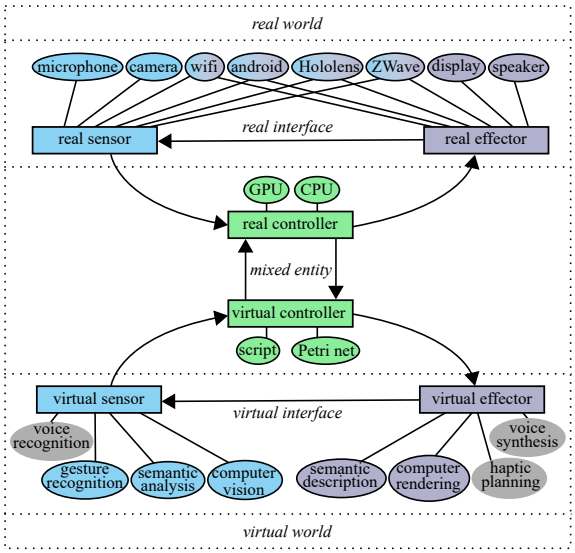

Figure 6: This Figure is a detailed version of the Figure 3. Our DOMIM-based framework provides non-greyed components. They are available in DOMIM application templates and easy to connect to the scene graph thanks to our Petri net editor.

We call a Petri net configuration a scenario. A generic DOMIM scenario implements a DOMIM-based mixed entity. This scenario separates the real and virtual parts of the mixed entity, implements for each part its interaction loop and synchronizes both parts. Figure 7 shows in our Petri net editor a mixed fan scenario based on a generic DOMIM scenario. A real fan wired to a smart plug and an application make up a mixed fan. The application controls the mixed fan behavior. This scenario runs on the Hololens. Developers assign object-oriented methods or properties of the components to the generic scenario transitions. These components belong to the mixed entities of the Unity scene graph:

[1]https://unity.com/fr

[2]https://github.com/genielabs/zwave-lib-dotnet

[3]https://developer.vuforia.com/

[4]https://github.com/artoolkit

- set real power state: updates the real fan properties from the virtual ones by calling a method of the mixed fan if the user switched the virtual button of the mixed fan,
- network communication: commands the Zwave server to update the real fan state according to its real properties, this commands the smart plug of the mixed fan,
- check real power state: asks the Zwave server the real fan state to update the real properties of the mixed fan, this checks the power state of the smart plug,
- set virtual power state: this transition effector calls a method of the mixed fan. This method updates the virtual fan properties with the real ones,
- semantics: parses messages received from other mixed entities of the MIS, formats the messages to send to these entities, for example, to synchronize their locations,
- check virtual power state: observes if the virtual properties of the mixed fan have changed if the mixed user interacted with the virtual power button of the mixed fan.

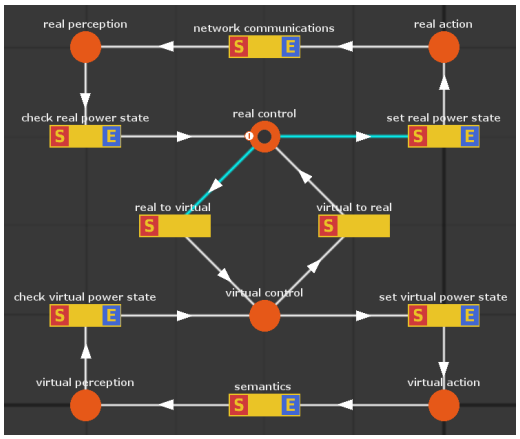

Figure 7: The Petri net scenario of a mixed fan, provided by our Petri net editor. Orange circles are places. Yellow boxes are transitions, the E letter indicates a transition effector, the S letter indicates a transition sensor. The black dot is the token at its initial place.

The process we just described is the same for any MIS application. Developers apply this process to all their MIS applications with the support of our DOMIM-based framework. They declare the properties of each mixed entity. They associate network components with them and tune their communications with other mixed entities composing the MIS. They provide each mixed entity with a 3D model, their virtual twin eventually completed by metaphors as the geometry of their virtual part, and their DOMIM-based behavior description. Finally, they associate each mixed entity with interaction techniques.

DOMIM simplifies the design and development of an MIS. It allows breaking an MIS into comprehensible parts. It distinguishes the entities constituting it, classifies their interrelations and spells their internal real-virtual interplay out. It allows reusing software by splitting it into reusable and generic components. Our DOMIM-based framework provides an integrated design and development tool in order to make flexible pre-developed application templates and components. Our framework also provides pre-developed software in order to accelerate the design and development of an MIS, bringing swiftness to the development of complex interactive systems.

4 USE CASE IMPLEMENTATION

Our validation use case consists of an MIS for smart building occupants. We chose this context since a building contains a large number of disparate connected objects. In this building, humans interact with a fan, a lamp, a thermometer, and a presence detector. We assume that objects are stationary. They are tracked by detecting the pose of a texture whose location is pre-defined in the building virtual twin. We model rooms and objects offline by hand for rendering performance reasons. This use case aims at controlling and monitoring mixed objects inside or outside the building. We propose the following interactions:

- inside a room:
  - interaction A, in-air gestures: an air tap gesture turns on and off the power of the fan and the lamp. We first focus on an object with head gaze, then validate our action with the tap gesture. Figure 13 shows this interaction. This interaction validates the capability of DOMIM to model an MIS using in-air gestures to interact in real-time with smart objects surrounding the user. It also validates the capability of our DOMIM-based framework to implement in-air gestures and interpret them to command mixed objects,
  - interaction B, tangible interactions: the simultaneous touch of both their real and virtual parts turns on and off the power of the fan and the lamp. We touch with the index finger the virtual button displayed by the Hololens on the surface of the real fan to command it, as shown in Figure 14. Compared to previous interaction, this

implementation validates our framework capability to implement tangible interactions,

- outside the room, interaction C combines tangible, tactile and in-air gestures. We use a smartphone as a tangible and tactile device in association with a Hololens for stereovision and in-air gestures in order to:
  - manipulate the virtual twin of a room represented as a world in miniature (WIM) (Alce et al., 2017),
  - turn on/off the fan and the lamp by touching the smartphone screen, as shown in Figure 15.

  Compared to previous interactions, this implementation validates our framework capability to use a smartphone as a tangible interface in order to implement and hybrid interaction mixing tangible, tactile and in-air gestural interactions,

We use DOMIM to design this use case and our DOMIM-based framework to develop it in order to validate our approach.

4.1 Design

We consider in this section the design of an MIS architecture and its software. Regarding DOMIM, the first step is to enumerate the entities composing the setup, shown in Figure 8:

- a mixed user enabled by:
  - a Hololens,
  - a smartphone Xiaomi Mi8 supporting ARCore,
- a mixed room containing:
  - an Alienware Area 51 running a Zwave server,
  - network enablers:
    - a Netgear R6100 wifi router to connect the PC, the Hololens and the smartphone,
    - a Z-Stick S2 Z-Wave USB antenna plugged to the laptop, making it our ZWave gateway,
  - mixed objects:
    - a fan plugged into a Fibaro Wall Plug, switching it on and off,
    - a lamp plugged into a Everspring AD142 plug, switching it on and off,
    - a Fibaro Motion Sensor, monitoring presence inside the room,
    - a Everspring ST 814 temperature/moisture detector, monitoring them inside the room.

We now detail how our framework provides the claimed services to designers.

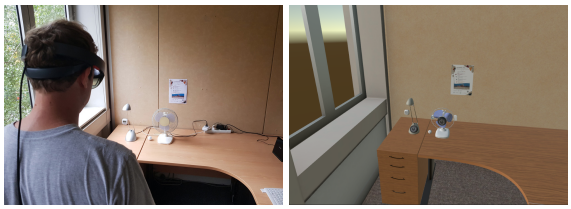

Figure 8: On the left, a capture of the use case setup. On the right, a capture of the room's virtual twin.

Simplicity
DOMIM allows us to segment an MIS complexity into several sub-MIS and to distinguish real and virtual interactions between mixed entities. Figure 9 shows the sub-MIS composed of the mixed user, a Hololens and the mixed fan. This MIS allows interactions A and B. Interactions are clearly identified, categorized and described.

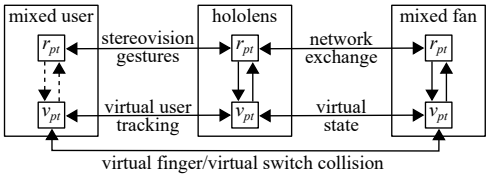

Figure 9: The external model based on DOMIM of the use case subsystem composed of a user, a Hololens and a mixed fan in the case of interaction B.

Flexibility
DOMIM eases the adaptability of an MIS to its evolutions, thanks to its power of abstraction. For example, the previous sub-MIS is abstract enough to replace a Hololens by a Magic Leap, a mixed fan by a mixed lamp, and a gesture interaction by a voice interaction effortlessly, as shown in Figure 10.

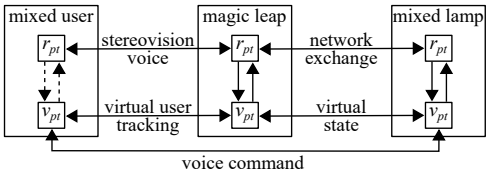

Figure 10: Adapting the MIS presented in Figure 9 to an MIS composed of a user, a Magic Leap and a mixed lamp, with vocal interactions.

Reusability
We now present how to reuse an MIS external model. The ability of DOMIM to produce simple and comprehensive models appears here. We extract from Figure 9 the mixed user, the Hololens and their interactions. We complete this sub-MIS with a smartphone in order to design interaction C, as described in Figure 11. Both smartphone and Hololens are self-located, and their locations are synchronized.

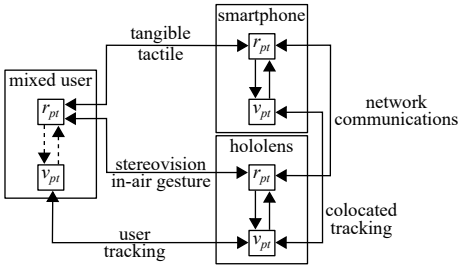

Figure 11: Interaction C, the external model of a mixed user enabled by a Hololens and an ARCore smartphone. ARCore provides the location of the smartphone, while the Hololens embeds its own SLAM in order to locate it.

The designed and implemented MIS validates the use of DOMIM as a simple, flexible and reusable model.

4.2 Implementation

Interactions inside each mixed entity application produce its behavior. This behavior enables its interactions with other mixed entities composing the MIS. Developers produce the applications supporting each mixed entity by using DOMIM application templates and pre-developed components provided by the DOMIM framework. For each application, developers create their mixed objects graph, declare for each mixed entity its properties, locate them in the virtual twin of the real space, and associate them with pre-developed pseudo-natural and network interactions. Our network components provide inter-platforms communications and IoT-based services like a ZWave service. Interaction components offer pseudo-natural interactions, like in-air gestures, described in this subsection and available as components in our framework. Figure 12 presents the relations between the mixed objects and devices of our MIS. The Zwave server commands devices and collects their real properties in order to synchronize their virtual part. Typically, the virtual part of a mixed fan knows if its real part is powered or not.

Interaction A: In-air Interactions with Mixed Objects. Our framework provides a DOMIM-based application template for the Hololens. The mixed user is supported by this application. The Hololens perceives the mixed user location with its embedded SLAM and tracks his hands with its depth camera. It renders the virtual properties of the mixed objects surrounding him as metaphors. It detects which mixed object the mixed user observes with gaze tracking, and the in-air tap gesture to switch on and off the mixed object. When the Hololens detects the mixed user intention to switch on and off a mixed object, it

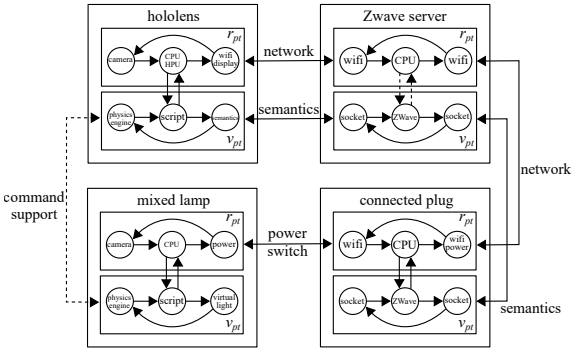

Figure 12: The internal model based on DOMIM of the use case subsystem composed of a Hololens, a ZWave server, a smart plug, and a lamp.

communicates to the Zwave server the command. The Zwave server then switches on/off the smart plug of the mixed object. Figure 13 presents the interaction A.

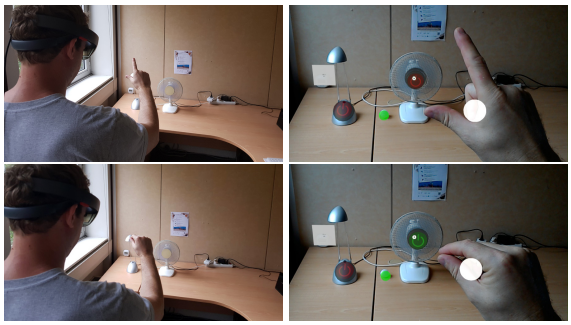

Figure 13: Interaction A: in-air interactions with mixed objects. At the top, the user's gaze focuses on the virtual power button of the mixed fan. On the bottom, the user uses the tap gesture to switch on the mixed fan.

Interaction B: Tangible Interactions with Mixed Objects. This tangible interaction allows the mixed user to switch on and off the smart plug of a mixed object by touching it. The Hololens DOMIM-based application detects the collision between the virtual twin of the user's hand and the virtual button augmenting the mixed object. The virtual button is displayed above the mixed object surface. When the user touches the virtual button he also touches the real object, making the virtual part of the mixed object tangible. The mixed user application developed for interaction A is modified by adding this virtual collision detection as an interaction modality in order to command the mixed object. Figure 14 shows interaction B, provided by our framework as a component.

This MIS implementation shows the capability of our DOMIM-based framework to design and develop MIS rapidly and efficiently.

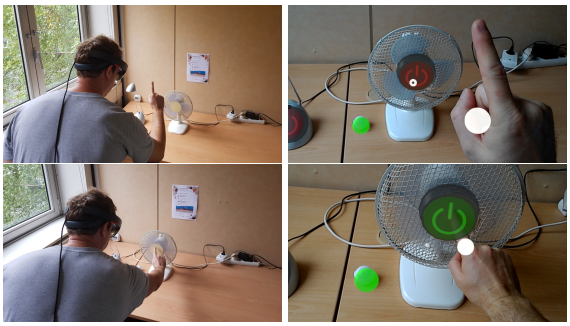

Figure 14: Interaction B: tangible interactions with mixed objects. At the top, the user is about to touch the virtual power button of the mixed fan. On the bottom, the user touches the virtual button in order to switch on the mixed fan.

Interaction C: Hybrid In-air, Tangible and Tactile Interactions with Tangible Virtual Objects. This interaction combines two user interfaces, the Hololens, and the smartphone. The smartphone is used as a tangible and tactile interface, a tracking device and a tangible mediator of the virtual world surfaces but not as a visual rendering device. The smartphone allows to tangibly manipulate the virtual twin of the room displayed on the smartphone by manipulating the smartphone itself. Our framework provides an application template for each device. The Hololens is aware of its location thanks to its embedded SLAM and of the location of the smartphone thanks to the smartphone ARCore SLAM. The smartphone continuously transmits its location to the Hololens. We synchronize both coordinate systems when the Hololens estimates the smartphone pose. The smartphone displays a texture. The Vuforia engine runs on the Hololens. When Vuforia detects the texture on the smartphone, it computes the transformation matrix of the smartphone. We then estimate a transition matrix with the help of this transformation matrix. This transition matrix defines a common coordinate system for both Hololens and smartphone locations. The sum of the Hololens location and the smartphone pose estimated by the Vuforia engine on Hololens is compared to the smartphone location estimated by ARCore in order to provide this transition matrix. Our framework provides this interaction as a component, including the transition matrix estimation. When the mixed user wants to interact with the virtual twin of a mixed object, the Hololens detects his free hand location and focuses on the closest virtual object twin. When the mixed user touches the tactile screen of the smartphone, the smartphone sends to the Hololens the touch event. The Hololens then sends to the Zwave service the command to the focused mixed object. Our framework provides this Zwave service hosted by a computer included in this MIS. We show interaction C in Figure 15.

Figure 15: Interaction C: hybrid in-air, tangible and tactile interactions with tangible virtual objects. At the top, the user manipulates the smartphone in order to manipulate the room's virtual twin displayed by the Hololens, the virtual twin of a mixed lamp is focused by his hand. On the bottom, the user touches the smartphone tactile screen in order to switch on the mixed lamp.

If we want to extend or change an MIS, we complete or change the DOMIM implementation of developed mixed entities, and eventually add new mixed entities based on DOMIM application templates and tune their interactions with the MIS.

5 CONCLUSION

In this paper, we presented our solution for the development of hybrid interactive systems blending MR, NUI, and the IoT that we call mixed interactive systems. Our approach responds to the need for appropriate models and tools to abstract and develop these complex systems. Our main contribution is our design-oriented mixed-reality internal model (DOMIM) of a mixed entity. This model supports the architecture, design and implementation of MIS. We used this model to produce our DOMIM-based framework providing simplicity, reusability, flexibility, and swiftness. A DOMIM framework-based project breaks an MIS into pieces in order to segment its complexity, and highly structure and factor them. Interactions between different platforms are provided by network, which allows a a high interoperability of DOMIM-based MIS. Our framework also allows to design software during implementation.

For future work, first, our DOMIM-based framework needs to be completed with additional interaction techniques, devices, and IoT services to improve its interoperability. Additional services could also be developed and integrated to ease the digital twin initialization and the discovery of its connected objects. Then, the model and the framework must be con-

fronted with other users cases at a larger scale such as smart factories and smart cities services. We also aim at evaluating the benefits of our solution. First, its benefits must be evaluated from a designer and developer point of view. We aim to demonstrate the efficiency of our framework and tools for the development of MIS compared to state of the art solutions. Second, its benefits must also be evaluated from an end-user point of view. Indeed, we could wonder if MR would facilitate interaction between users and their surroundings. A first step could be to compare user's appreciations and performances between an MIS and a single smartphone application for controlling the same connected environment.

REFERENCES

Alce, G., Roszko, M., Edlund, H., Olsson, S., Svedberg, J., and Wallergård, M. (2017). [POSTER] AR as a User Interface for The Internet of Things #x2014;Comparing Three Interaction Models. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 81–86.

Atzori, L., Iera, A., and Morabito, G. (2017). Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm. *Ad Hoc Networks*, 56:122–140.

Blake, J. (2012). The natural user interface revolution. In *Natural User Interfaces in. Net*, pages 1–43. Manning publications edition.

Bouville, R., Gouranton, V., Boggini, T., Nouviale, F., and Arnaldi, B. (2015). #FIVE : High-level components for developing collaborative and interactive virtual environments. In *2015 IEEE 8th Workshop on Software Engineering and Architectures for Realtime Interactive Systems (SEARIS)*, pages 33–40, Arles, France. IEEE.

Bouzekri, E., Canny, A., Martinie, C., and Palanque, P. (2018). A Generic Software and Hardware Architecture for Hybrid Interactive Systems. In *EICS 2018 – Workshop on Heterogeneous Models and Modeling Approaches for Engineering of Interactive Systems*, Paris, France.

Claude, G., Gouranton, V., and Arnaldi, B. (2015). Versatile Scenario Guidance for Collaborative Virtual Environments. In *Proceedings of 10th International Conference on Computer Graphics Theory and Applications (GRAPP'15)*, berlin, Germany.

Coutrix, C. and Nigay, L. (2006). Mixed reality: a model of mixed interaction. In *Proceedings of the working conference on Advanced visual interfaces - AVI '06*, page 43, Venezia, Italy. ACM Press.

Dubois, E., Bortolaso, C., Appert, D., and Gauffre, G. (2014). An MDE-based framework to support the development of Mixed Interactive Systems. *Science of Computer Programming*, 89:199–221.

Dubois, E., Gray, P., and Nigay, L. (2010). *The engineering of mixed reality systems*. Human–Computer Interaction Series. Springer Science & Business Media.

Glaessgen, E. and Stargel, D. (2012). The digital twin paradigm for future NASA and US Air Force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA*, page 1818.

Greer, C., Burns, M., Wollman, D., and Griffor, E. (2019). Cyber-physical systems and internet of things. Technical Report NIST SP 1900-202, National Institute of Standards and Technology, Gaithersburg, MD.

Grieves, M. W. (2019). Virtually Intelligent Product Systems: Digital and Physical Twins. *Complex Systems Engineering: Theory and Practice*, pages 175–200.

Hussey, A. and Carrington, D. (1997). Comparing the MVC and PAC architectures: a formal perspective. *IEE Proceedings - Software Engineering*, 144(4):224–236.

Jacob, R. J., Girouard, A., Hirshfield, L. M., Horn, M. S., Shaer, O., Solovey, E. T., and Zigelbaum, J. (2008). Reality-based Interaction: A Framework for post-WIMP Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 201–210, New York, NY, USA. ACM.

Kelaidonis, D., Somov, A., Foteinos, V., Poulios, G., Stavroulaki, V., Vlacheas, P., Demestichas, P., Baranov, A., Biswas, A. R., and Giaffreda, R. (2012). Virtualization and Cognitive Management of Real World Objects in the Internet of Things. In *2012 IEEE International Conference on Green Computing and Communications*, pages 187–194, Besancon, France. IEEE.

Kim, K., Bölling, L., Haesler, S., Bailenson, J., Bruder, G., and F. Welch, G. (2018). Does a Digital Assistant Need a Body? The Influence of Visual Embodiment and Social Behavior on the Perception of Intelligent Virtual Agents in AR. pages 105–114, Munich.

Kritzler, M., Funk, M., Michahelles, F., and Rohde, W. (2017). The Virtual Twin: Controlling Smart Factories Using a Spatially-correct Augmented Reality Representation. In *Proceedings of the Seventh International Conference on the Internet of Things*, IoT '17, pages 38:1–38:2, New York, NY, USA. ACM.

Lacoche, J., Le Chenechal, M., Villain, E., and Foulonneau, A. (2019). Model and Tools for Integrating IoT into Mixed Reality Environments: Towards a Virtual-Real Seamless Continuum. In *ICAT-EGVE 2019 - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*, Tokyo, Japan.

Leonard, J. J. and Durrant-Whyte, H. F. (1991). Simultaneous map building and localization for an autonomous mobile robot. In *Proceedings IROS '91:IEEE/RSJ International Workshop on Intelligent Robots and Systems '91*, pages 1442–1447 vol.3.

Lin, S., Cheng, H. F., Li, W., Huang, Z., Hui, P., and Peylo, C. (2017). Ubii: Physical World Interaction Through Augmented Reality. *IEEE Transactions on Mobile Computing*, 16(3):872–885.

Liu, W. (2010). Natural user interface- next mainstream product user interface. In *2010 IEEE 11th International Conference on Computer-Aided Industrial Design Conceptual Design 1*, volume 1, pages 203–205.

MacWilliams, A., Sandor, C., Wagner, M., Bauer, M., Klinker, G., and Bruegge, B. (2003). Herding Sheep: Live System Development for Distributed Augmented Reality. In *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '03, pages 123–, Washington, DC, USA. IEEE Computer Society.

Mann, S., Furness, T., Yuan, Y., Iorio, J., and Wang, Z. (2018). All Reality: Virtual, Augmented, Mixed (X), Mediated (X,Y), and Multimediated Reality. *arXiv:1804.08386 [cs]*. arXiv: 1804.08386.

Marchand, E., Uchiyama, H., and Spindler, F. (2016). Pose Estimation for Augmented Reality: A Hands-On Survey. *IEEE Transactions on Visualization and Computer Graphics*, 22(12):2633–2651.

Milgram, P. and Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays. *IEICE Transactions on Information Systems*, E77-D.

Mistry, P. and Maes, P. (2009). SixthSense: A Wearable Gestural Interface. In *ACM SIGGRAPH ASIA 2009 Art Gallery & Emerging Technologies: Adaptation*, SIGGRAPH ASIA '09, pages 85–85, New York, NY, USA. ACM.

Nitti, M., Pilloni, V., Colistra, G., and Atzori, L. (2016). The Virtual Object as a Major Element of the Internet of Things: A Survey. *IEEE Communications Surveys & Tutorials*, 18(2):1228–1240.

Norman, D. A. (2010). Natural User Interfaces Are Not Natural. *interactions*, 17(3):6–10.

Normand, E. and McGuffin, M. J. (2018). Enlarging a Smartphone with AR to Create a Handheld VESAD (Virtually Extended Screen-Aligned Display). In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 123–133, Munich, Germany.

Norouzi, N., Bruder, G., Belna, B., Mutter, S., Turgut, D., and Welch, G. (2019). A Systematic Review of the Convergence of Augmented Reality, Intelligent Virtual Agents, and the Internet of Things. In Al-Turjman, F., editor, *Artificial Intelligence in IoT*, pages 1–24. Springer International Publishing, Cham.

Petersen, N. and Stricker, D. (2009). Continuous Natural User Interface: Reducing the Gap Between Real and Digital World. In *Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '09, pages 23–26, Washington, DC, USA. IEEE Computer Society.

Pfeiffer, T. and Pfeiffer-Leßmann, N. (2018). Virtual Prototyping of Mixed Reality Interfaces with Internet of Things (IoT) Connectivity. *i-com*, 17(2):179–186.

Poslad, S. (2011). *Ubiquitous computing: smart devices, environments and interactions*. John Wiley & Sons.

Sánchez López, T., Ranasinghe, D. C., Harrison, M., and Mcfarlane, D. (2012). Adding Sense to the Internet of Things. *Personal Ubiquitous Comput.*, 16(3):291–308.

Sulisz, C. and Seeling, P. (2012). An Off-the-shelf Wearable HUD System for Support in Indoor Environments. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, MUM '12, pages 60:1–60:4, New York, NY, USA. ACM.