Comparing Properties of Massively Multiplayer Online Worlds and the Internet of Things

Kim Nevelsteen and Theo Kanter and Rahim Rahmani
Department of Computer and Systems Sciences
Stockholm University
Borgarfjordsgatan 12, Kista, Sweden
Email: see http://dsv.su.se/en/research/research-areas/immersive

***Abstract*—A virtual world engine at the massively multiplayer scale is a massively multiplayer online world (MMOW); one thing virtual world engines realized when going into the scale of MMOs, is the cost of maintaining a potentially quadratic number of interactions between a massive number of objects, laid out in a spatial dimension. With the rise of the Internet of Things (IoT), this means recognizing the need for architectures to handle billions of devices and their interactions. Research into IoT was fueled by research in wireless sensor networks, but rather than start from a device perspective, this article looks at how architectures deal with interacting entities at large scale. The domain of MMOWs is examined for properties that are affected by scale. Thereafter the domain of IoT is evaluated to see if each of those properties are found and how each is handled. By comparing the current state of the art of MMOWs and IoT, with respect to scalability, it is discussed how research from one domain can possibly be exapted to the other domain and vice versa. A case study of a MMOW interfacing with IoT is presented in closing.**

I. Introduction

A virtual world at the massively multiplayer scale is a Massively Multiplayer Online World (MMOW), of which some are games [1]. One thing virtual world engines realized when going into the scale of MMOs, is the cost of maintaining a potentially quadratic number of interactions between a massive number of objects, laid out in a spatial dimension[2], [3]. Yahyavi et al. [2] explicitly state that the architectures they focus on for MMO games are also applicable to other distributed systems *e.g.,* technology-sustained pervasive games [4].

With the rise of the Internet of Things (IoT) [5], [6], [7], this means recognizing the need for architectures to handle billions of devices and their interactions. Endeavors are already underway in an attempt to create an IoT platform, but a "solution that addresses all the aspects required by the IoT is yet to be designed" [6]. Miorandi et al. [8] summarize a number of research initiatives happening worldwide *e.g.,* HYDRA allowing developers to incorporate heterogeneous devices, and IoT-A concentrating on interoperability. Devices in IoT (*e.g.,* RFID) allow for the mapping of the physical world into the virtual world [7] *i.e.,* using 'non-standard input devices' to blend the virtual and the physical [4] into a pervasive system [9].

Research into IoT was fueled by research in Wireless Sensor and Actuator Networks (WSANs) [5], but rather than start from a device perspective, in Section II, the domain of MMOWs is examined to see how architectures deal with scalability *i.e.,* properties that are affected by scale are gathered from the domain of MMOWs. In Section III, it is then evaluated if each of those properties is found in the domain of IoT and how each property is handled. By comparing the current state of the art of MMOWs and IoT, with respect to scalability, Section IV points to how research from one domain can possibly be 'exapted' [10] to the other domain and *vice versa*. Section V presents a case study as to how a MMOW can interface with IoT *i.e.,* how a MMOW can be a 'mediator' [9] for IoT. And, Section VI summarizes the conclusions.

II. Examining the Domain of MMOWs

Since virtual world engines (implementing MMOWs) have long been dealing with the cost of maintaining a massive number of objects, here properties affected by scale are gathered from the domain of MMOWs. Properties are categorized according to ISO 25010:2011 [11].

The property of "**scalability** can be achieved either by: (1) increasing the resources or by (2) reducing the consumption" [2]. Adding multiple servers to distribute the load of handling interactions is designed to increase the amount of resources. To alleviate the scalability problem, a server cluster can be used instead of the single server *i.e.,* a co-located cluster of servers that collectively act as a centralized unit to serve all clients. To resolve the scalability issue further, current MMO research is looking into pure peer-to-peer (P2P) solutions or a hybrid P2P server cluster combination. Server clusters can be formed in a distributed P2P system by assigning a 'region controller' [12] to supervise over the peers in the cluster. If multiple servers are used, two ways to partition computational space are regions and shards. Regionalization divides space into regions, with a different set of servers responsible for each region, and replication means having multiple copies of the same space *i.e.,* shards. With replicated shards there is no or minimal interaction between shards [2]. A hybrid is also possible *e.g.,* a shard divided into regions. To achieve scalability through a decrease of consumption, 'interest management' can be used in combination with partitioning; the amount of resources an entity consumes is limited by assigning each entity an 'area of interest'. There are many ways to perform interest management, both structured and unstructured, and several challenges remain (see Yahyavi et al. [2] for details).

978-1-5090-0679-3/16/$31.00 ©2016 IEEE

Latency can be defined as "the delay between execution of an update at the primary copy of an object and the replica receiving the object update" [2]. One of the critical aspects of MMOWs is their real-time **responsiveness**, which demands "message latency should be minimized while bandwidth use should be efficient" [3]. From a human perspective, real-time responsiveness is achieved when "the time between the event being generated and the time it is executed and perceived by the user is unperceivable [*sic*]" [13] *i.e.,* responsiveness despite latency. The tolerance threshold for latency in games is between 100 and 300 milliseconds, depending on the game type (ranging from FPS games to RPGs) [2]. Liu et al. [13] report a trade-off between consistency and responsiveness (throughput).

If multiple servers are collaborating to maintain a compute space, then **consistency** between the state of each server must be maintained despite networking delay. Yahyavi et al. [2] report a well-known trade-off between performance (availability) and consistency restrictions. Consistency involves having a primary copy of each object and sending updates (*i.e.,* update dissemination) to replicas[1], in such a way that the causal order of events are consistent [2], [13]. A way to assess a degree of inconsistency is by comparing the (potentially inconsistent) state of each replica against a virtual perfect replica [2].

Considering the prevalence of smartphones, some MMOWs have moved to a mobile platform. But, the limitations in networking and computing power of such devices are a factor [2]. Taking into account whether to use local or remote resources, is important for efficient **resource utilization**. Currently, MMOWs do not have to deal with the complications of having to support (partially) disconnected network architectures *e.g.,* MANets [14].

For MMOWs using a centralized cluster of servers, the **availability** of end nodes in the network is not so much an issue, but in P2P, nodes are "much more prone to failures or unscheduled disconnections" [2], which can adversely affect the network topology [3]. Related to the availability of the network, nodes must be fault tolerant and support data persistence for recoverability. Data persistence means the ability to save and access world states despite disconnections, which remains a challenge in P2P systems [3]. Redundant backup copies of primary objects, redundant network connections and redundant servers [12] are ways to provide fault tolerance.

In games, one of the main concerns is cheating, which corresponds to **security** in other applications. Yahyavi et al. [2] present three categories for cheating: (1) Interrupting Information Dissemination, which includes premature disconnection, flooding of the network, replay attacks and the dropping of updates to peers; (2) Illegal Game Actions, which includes tampering of end nodes of the network, falsifying identity and the use of computer enhanced data where human readings are expected; and (3) Unauthorized Information Access, which includes the tampering of end nodes or network traffic analysis in order to gain access to privileged information. If a P2P communication is used instead of a centralized approach, then dealing with cheating (security) and maintaining control over the game remains a challenge [2], [3] *i.e.,* it is easier to build a game using a centralized architecture [4]. Having more control means that the architecture is easier to manage and maintain [2].

[1]Note the difference between objects being replicated here, and entire server partitions being replicated as with shards.

III. Evaluating Properties in the Domain of IoT

Given the six properties that are affected by scale, the domain of IoT is examined to evaluate if each of those properties is found in the domain of IoT and how each property is handled.

Because of the projected size of the IoT, **scalability** is expected to be a major issue and often mentioned in literature [7], [5], [14], [8]. Literature is divided on whether centralized cloud computing, decentralized P2P, or a hybrid architecture will achieve the needed scalability (through the addition of resources) for IoT. Arguments for cloud computing are based on the advantages of abundant cloud storage and processing capabilities, while being tightly controlled for efficient energy usage and reliability [5]. Counter arguments state that decentralized P2P architectures show promise [5], [7] and that a centralized architecture cannot lead to a truly scalable solution [14] *i.e.,* wanting to reap the "seemingly endless amount of distributed computing resources and storage owned by various owners" [5]. If billions of devices will be soon connected to the Internet, producing a potentially quadratic number of interactions [2], then somehow IoT needs to facilitate those interactions [6]. Currently, many sensor network solutions are sense-only solutions, with many-to-one interactions *i.e.,* only a small fraction of solutions are sense-and-react applications (using actuators), with many-to-many interactions [15]. To achieve scalability through reducing consumption, there seems to be only one construct mentioned in IoT literature. From the domain of WSANs, the construct of 'clustering', for both physical devices and also objects present on those devices, has been suggested for scalability; either abstract regions are formed [15] or a few nodes are elected as 'cluster heads' to act as decentralized authorities for each cluster [14]. A problem with current clustering protocols being that mobility of nodes has hardly been considered [14].

To achieve the appropriate **responsiveness**, IoT is said to require two classes of traffic: throughput and delay tolerant elastic traffic; and, bandwidth and delay sensitive inelastic (real-time) traffic. These two types can be further discriminated into various levels of quality of service [5]. Perera et al. [6] mention that "real-time data processing is essential", referring to two different classes of traffic, namely 'event driven' versus 'time driven', which correlates to 'event-triggered' and 'periodic', respectively, in WSAN literature [15].

"The synchronisation [*sic*] of data across the architecture" (*i.e.,* **consistency**) is envisioned to be a big challenge in IoT [14]. In WSAN literature, four approaches emerge on how to provide access to data: (1) similar to a relational database, (2) as remotely accessible variables or tuples, (3) through

mobile code, or (4) by message passing [15]. A number of projects in IoT use 'participatory sensing', where sensors on people are used to obtain readings local to the user [5]. A difficultly in IoT, is that (partially) disconnected networks exist, so "there is a potential for various non-homogeneous copies of object data across the architecture", making support for 'one-copy' [primary copy] equivalence problematic *i.e.,* "the value of all copies should be identical after a transaction" [14].

The aim of IoT is to include many platforms and devices with various resource limitations (*e.g.,* cloud server clusters, desktop computers, mobile devices, sensor networks and everyday objects), with access to resources through various network infrastructure, such as a wireless personal area network (*e.g.,* Bluetooth), wireless local area network (*e.g.,* Wi-Fi), wireless wide area network (*e.g.,* 2G and 3G networks), or satellite network (*e.g.,* GPS) [6]. IoT can be discussed from two perspectives, 'Internet' centric and 'Thing' centric [5], [7]; leading to different architectures *i.e.,* centered on cloud computing ('remote'), or centered on the user ('local'), respectively [5]. Not all 'Things' in the IoT have sufficient energy and processing power to do comprehensive data processing (*e.g.,* sensors in a WSAN); in that case, data must be networked to more powerful remote devices and processed there [6]. In participatory sensory, people are centric [5], which can be generalized such that interactions for an object in the IoT are highly dependent on the object's local surroundings *e.g.,* presence of other objects or people [6]. If (partially) disconnect networks are present, interaction with the local environment is most likely still possible, contrary to remote access. Because billions of IoT devices will potentially be communicating with each other, IoT research has noted the importance of optimizing computation in the various 'layers' of the IoT infrastructure [6] *i.e.,* handling **resource utilization**. Another example of this being the "notion of distributing computation in order to reduce the communication overhead, which is generally termed in-network processing or in-network computing" [5].

With the existence of partially or permanent disconnected networks in IoT [14], robustness and fault tolerance (*i.e.,* **availability**) will become fundamental research topics in IoT [8]. Considering the 'extremely large scale' of IoT and the 'high level of dynamism in the network', self-organization is suggested as a solution [8] *e.g.,* supporting merges and splits of the network [14]. In WSAN literature, very little research has been done with respect to having mobile nodes in a network [15]. And, programming approaches for WSANs provide for only limited guarantees in the face of the various types of hardware faults [15].

Perera et al. [6] state that the "IoT paradigm will intensify the challenges in **security** and privacy". Security is related to concepts such as authentication, privacy and integrity. For IoT, cryptographic algorithms commonly used in authentication are typically problematic in devices with various resource limitations (*e.g.,* sensor networks), using large amounts of energy and bandwidth [7]. Authentication usually involves identifying people, but in IoT identities are associated with objects [8]. If authentication is to be possible in (partially) disconnected environments, the procedure will have to be possible locally [14]. Privacy refers to the access of data related to an individual [8]. While surfing the Internet, individuals usually play an active role in their privacy, but in IoT sensors are expected to collect information about individuals passively, without them actively using an IoT service and without control over what information is being collected [7], [6]. Once the information is generated, it will most like be retained indefinitely, unless a mechanism is in place to allow for 'digital forgetting' [7]. The purpose of data integrity is to prevent adverse modification of data without detection [7], [14]. A criticism with WSANs and thus IoT, is that hardware components are easy to physically attack, because they are largely unattended and if wireless communication is used, it can be eavesdropped [7]. Security and privacy are very much open issues in IoT [6], [8].

IV. Comparing Key Properties of MMOWs and IoT

By comparing the current state of the art of MMOWs and IoT, with respect to scalability, it becomes clear how research from one domain can possibly be exapted to the other domain and *vice versa*.

To achieve **scalability** through adding resources, cloud computing and P2P solutions are being considered in the domain of IoT, but research in the domain of MMOWs is more advanced, with running platforms utilizing partitioning schemes such as regionalization and replication in combination with interest management. To reduce consumption, clustering has been suggested in IoT, for devices and objects present on those devices. Clustering for devices in IoT can be equated to the partitioning techniques of regions and shards for MMOWs; and, clustering for objects in IoT can be equated to interest management for MMOWs [2], [14]. For example, López et al. [14] suggest using clusters determined through the use of context information, which is very similar to the existing MMOW project called *Donnybrook* using 'interest sets' [2]. There seems to be an overarching opinion that a single architecture or middleware will enable IoT, but there is a risk of fragmentation [8]. With many working on their own platform, one can speculate if IoT will be a collection of clouds, also facing interoperability problems found in multicloud computing *e.g.,* the locking in of clients [16]. If IoT does fragment and considering the advances in the domain of MMOWs, this means solutions from the domain of MMOWs (specifically the Metaverse; see Dionisio et al. [17]) could be exapted to the domain of IoT.

In the IoT literature consulted, there was no mention of the consistency and **responsiveness** trade-off found in MMOWs. Real-time data is one of the properties that "distinguish virtual worlds [MMOWs] from other distributed systems" [13]. Real-time traffic will not be required in all applications of IoT, but those applications that require delay sensitive inelastic traffic would be in the same class as a MMOW. Contrary to cloud-based MMOWs, response times in IoT can be kept lower, if

interactions are kept in a local environment rather than with remote resources, due to reduced networking latency.

Also not in IoT literature, was any mention of the performance (availability) and consistency restrictions, in a partitioned space *i.e.,* a distributed system. A metric for **consistency** in MMOWs is 'drift distance' [3]; for a moving entity in a virtual world, the drift distance is, the absolute value of: the distance between the position of the entity locally and its position remotely. When dealing with the physical world, the drift distance could be calculated between the local position of the entity and its physical position. This means that given a physical entity with multiple virtual replicas (*i.e.,* multiple versions of reality) the copy most similar to the local physical entity is the one most consistent.

Until rather recently MMOW architectures have had the luxury of only having to support the desktop class of devices, connected to a server cluster, with MMOWs on mobile devices only recently becoming more common. In IoT, efficient **resource utilization** is still very much a challenge. Some sensors are able to produce a continuous stream of data and networking that data to more powerful devices or the cloud might outweigh the benefit, given the limited computing power of sensors and mobile devices. If faced with (partially) disconnected networks, the benefit of accessing local resources must be considered. Premature disconnections and decentralized P2P for MMOWs has a likeness to partially disconnected networks in IoT. And, if MMOWs are to interface with IoT (see Section V), this means MMOWs must also face fully disconnected networks. To allow for the use of local resources in a disconnected state, authority over local resources can be delegated to the local environment; a caveat being that, although MMOWs have considered P2P or hybrid architectures, engines have typically been centralized clusters, rather than geographically distributed systems.

Similar to resource utilization, IoT research has already taken into account (partially) disconnected networks, but if MMOWs are to interface with IoT, issues with **availability** must be dealt with. If a shard is disconnected, delay tolerant networking can be used to merge the shard when a connection is established [1], [18].

Authentication is not mentioned as an issue for MMOWs as long as a central authority is used. Similarly, a centralized approach allows for control *e.g.,* for maintenance. Missing from the IoT discourse, is the issue of maintenance; if IoT is to be enabled through a single platform, how will updates to the platform be managed? Miorandi et al. [8] state that IoT needs to move away from centralized approaches, to a fully distributed and dynamic approach. If decentralized P2P is used for either IoT or a MMOW, **security** remains an issue. Similar to IoT, end nodes in a MMOW cloud are also easily attacked and the networking tampered with. Privacy in a MMOW is similar to authentication, except that rather than trying to access private player data, attackers are trying to gain sensitive information pertaining to game state. Most game cheats are primarily against data integrity.

V. As a Behind the Scenes Resource

In previous sections, properties have been found in the domain of MMOWs, that overlap with the domain of IoT. To show that there are applications situated in both domains, a case study is presented here that combines MMOWs with IoT. The case study exemplifies that a virtual world can be used as a mediator for the physical world.

Nevelsteen [4] concluded that a virtual world engine is in the same product line as a game engine for pervasive games. Considering pervasive games need a sensory system to monitor the physical world (through the use of non-standard input devices), IoT could potentially serve as such a sensory system. Since a MMOW is a virtual world at a massive scale, this means that a MMOW can be exploited "as a 'behind the scenes' resource for coordinating and managing devices and interaction in the physical space" [19].

Furthermore, according to Rehm et al. [9], the concept of IoT has been recently been replaced by the concept of Cyber-Physical Systems (CPS), which are the integration of computation, networking, and physical processes; "embedded computers and networks monitor and control the physical processes, with feedback loops where physical processes affect computations and *vice versa*" [20]. Rehm et al. believe that "virtual worlds can serve as platforms to facilitate the integration required by CPS". And, by extrapolation, they "conceive of a unified platform, the Metaverse, built on VW [virtual world] technologies that allow for the integration of technological, physical, and human elements of CPS".

VI. Conclusions

In this article it has been shown how research from the MMOW domain can possibly be exapted to the IoT domain and *vice versa*. Six properties, specific to a massive number of entities interacting, have been identified; first in the domain of MMOWs, second in the domain of IoT, and then compared in discussion. IoT can clearly learn from advances, in availability with respect to P2P systems, and scalability, from the domain of MMOWs. When virtual worlds start to incorporate IoT, blending the virtual and the physical, MMOWs can clearly learn from advances made, in resource utilization, availability, and responsiveness with respect to (partially) disconnected networks, from the domain of IoT. For consistency and security, there seems to be advances in both domains that can cross over to the other domain. The rise of the IoT means recognizing the need for architectures to handle billions of devices and their interactions; this includes virtual worlds (MMOWs and the Metaverse) used as mediators for the physical world (via IoT), in pervasive applications and CPS.

Acknowledgment

Research was made possible by a grant from the Swedish Governmental Agency for Innovation Systems to the Mobile Life Vinn Excellence Center.

REFERENCES

[1] K. J. L. Nevelsteen, "Virtual world, defined from a technological perspective, and applied to video games, mixed reality and the metaverse," submitted 2015. [Online]. Available: http://arxiv.org/abs/1511.08464

[2] A. Yahyavi and B. Kemme, "Peer-to-peer architectures for massively multiplayer online games: A survey," *ACM Computing Surveys (CSUR)*, vol. 46, no. 1, p. 9, 2013.

[3] S.-Y. Hu, H.-F. Chen, and T.-H. Chen, "VON: a scalable peer-to-peer network for virtual environments," *Network, IEEE*, vol. 20, no. 4, pp. 22–31, 2006.

[4] K. J. L. Nevelsteen, *A Survey of Characteristic Engine Features for Technology-Sustained Pervasive Games*, ser. SpringerBriefs in Computer Science. Switzerland: Springer International Publishing, 5 2015.

[5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[6] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 1, pp. 414–454, 2014.

[7] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[8] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of Things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[9] S.-V. Rehm, L. Goel, and M. Crespi, "The metaverse as mediator between technology, trends, and the digital transformation of society and business," *Journal For Virtual Worlds Research*, vol. 8, no. 2, 2015.

[10] P. Johannesson and E. Perjons, *An Introduction to Design Science*. Switzerland: Springer International Publishing, 2014.

[11] ISO/IEC JTC 1/SC 7, "ISO/IEC 25010:2011 systems and software engineering – systems and software quality requirements and evaluation (SQuaRE) – system and software quality models," International Organization for Standardization (ISO), Tech. Rep., 2011.

[12] T. Hampel, T. Bopp, and R. Hinn, "A peer-to-peer architecture for massive multiplayer online games," in *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*, no. 48. New York, NY, USA: ACM, 2006.

[13] H. Liu, M. Bowman, and F. Chang, "Survey of state melding in virtual worlds," *ACM Computing Surveys (CSUR)*, vol. 44, no. 4, p. 21, 2012.

[14] T. S. López, A. Brintrup, M.-A. Isenberg, and J. Mansfeld, *Architecting the Internet of Things*. Berlin Heidelberg: Springer-Verlag Berlin Heidelberg, 2011, ch. Resource Management in the Internet of Things: Clustering, Synchronisation and Software Agents, pp. 159–193.

[15] L. Mottola and G. P. Picco, "Programming wireless sensor networks: Fundamental concepts and state of the art," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 19, 2011.

[16] M. Singhal, S. Chandrasekhar, T. Ge, R. Sandhu, R. Krishnan, G.-J. Ahn, and E. Bertino, "Collaboration in multicloud computing environments: Framework and security issues," *Computer*, vol. 46, no. 2, pp. 76–84, 2013.

[17] J. D. Dionisio, W. G. Burns III, and R. Gilbert, "3D virtual worlds and the metaverse: Current status and future possibilities." *ACM Computing Surveys*, vol. 45, no. 3, pp. 34:1 – 34:38, 2013.

[18] I. Demeure, A. Gentes, J. Stuyck, A. Guyot-Mbodji, and L. Martin, "Transhumance: a platform on a mobile ad hoc network challenging collaborative gaming," in *Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on*. Irvine, CA, USA: IEEE, 2008, pp. 221–228.

[19] C. Greenhalgh, S. Izadi, T. Rodden, and S. Benford, "The EQUIP platform: Bringing together physical and virtual worlds," Mixed Reality Laboratory - University of Nottingham-UK, Tech. Rep., 2001. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.3793

[20] CHESS. (2016) Center for hybrid and embedded software systems – cyber-physical systems. [Online]. Available: http://cyberphysicalsystems.org