Chapter 4
Possible Extensions

4.1 Challenges and Open Issues

The problems, caveats and disadvantages mentioned (Sect. 3.4) in the case study serve to highlight challenges and open issues for the creation of a would-be pervasive game engine. These challenges include: (1) using distributed and decentralized architectures; (2) extending ubiquitous computing; (3) interoperability; and, (4) creating game master interfaces and tools.

4.1.1 Distributed and Decentralized Architectures

Exemplified by Demeure, Gentes, Stuyck, Guyot-Mbodji, and Martin (2008) (see also Sect. 2.3.2), fully decentralized architectures exist, where the game state is not centrally controlled and only shared with other clients when opportune. A challenge exists pertaining to the extent that a decentralized architecture can be utilized for games. Issues arise as how to: maintain security, maintain a shared data space and prevent cheating (Yahyavi & Kemme, 2013); gather and persist data (e.g., for monitoring); or, build dynamic user interfaces (see Sect. 3.4.8). To deal with the scalability issue in Sect. 3.4.2, virtual world engines already exist that use a centralized distributed system of servers for load balancing (BigWorld, 2014), so utilizing such techniques for pervasive games seems evident.

Possibly the most direct extension of the work contained in this book, would be to implement the feature set from the survey in a modern virtual distributed world engine (e.g., Big World Technology (BigWorld, 2014)), and analyze its ability as a pervasive game engine. Another approach being to analyze architectures of large scale pervasive games, based on distributed computing e.g., Ingress (Niantic Labs, 2014).

K.J.L. Nevelsteen, *A Survey of Characteristic Engine Features for Technology-Sustained Pervasive Games*, SpringerBriefs in Computer Science,
DOI 10.1007/978-3-319-17632-1_4

4.1.2 Extending Ubiquitous Computing

Devices and systems have the potential to offer richer context information for context-awareness e.g., the incorporation of body metrics or social relations. Ubiquitous computing remains a challenge, with open issues: increased utilization of context-awareness; reduction of soft events (e.g., in communication, see Sect. 3.4.9); focusing on technology that can be effectively pushed into the background (e.g., for ubiquity of access and diegetic communication); and, obtaining ubiquitous persona and presence (Dionisio, Burns III, & Gilbert, 2013) (see Sect. 3.4.3). The latter recognizing that a player's identity is made up of the sum of their interactions with the game e.g., crossmedia through different devices or interfaces. The amount of uncertainty in ubiquitous computing has been reduced considerably; early writings on pervasive games include much on mobile networking issues, which are solved in mainstream technologies today, but some issues are still critical e.g., losing connectivity by switching between WLAN and mobile networks. A partial solution could be that of delay-tolerate network communication, used in FinN (Akribopoulos et al., 2009), to obtain an eventually consistent game state in their distributed system.

4.1.3 Interoperability

A device abstraction layer is suggested in this book, but without a concrete design. In 2004, Greenhalgh, Izadi, Mathrick, Humble, and Taylor (2004) set out to interconnect heterogeneous devices with the EQUIP/ECT technologies. Broll, Ohlenburg, Lindt, Herbst, and Braun (2006) state *interoperability* in pervasive games to be a 'well-known problem'. And, a number of years later, Branton, Carver, and Ullmer (2011) dedicate an entire publication to deal with the 'important challenge' of interoperability through standardization. Many innovations, such as new languages or middleware, are sited by Branton et al. (2011) as partial solutions, but 'compatibility' (ISO, 2011) between web services was noted as 'largely lacking'. Since some game engines and service-oriented architectures are already distributed systems, and they interact (see Sect. 3.4.3), then interoperability is an issue between heterogeneous distributed systems as well i.e., similar to multi-cloud network communication (Singhal et al., 2013). Interoperability remains a challenge with the amount of heterogeneous devices and systems increasing and becoming more diverse.

4.1.4 Game Master Interfaces and Tools

This book discusses the challenge of building a reusable pervasive game engine. It seems reasonable to infer that reusable game master interfaces and authoring tools should also exist (e.g., see Broll et al. 2006). Some game master tools have already

been created (e.g., for mobile games (Paelke, Oppermann, & Reimann, 2008) and authoring tools for location-based games (Oppermann, 2009)), but a more general reusable approach remains a challenge (Guerrero Corbi, 2014; Benford, Giannachi, Koleva, & Rodden, 2009) (see Sect. 3.4.7). Open issues are: capturing soft events and entering them in the game state; reducing the potential overload of data into a human consumable format; creating interfaces and visualizations that are applicable to a wide variety of games; and generating interfaces and visualizations that cater to the activity of game mastering rather than just presenting information.

4.2 Conclusion

That a game engine can be repurposed, has already been shown by Lewis and Jacobson (2002). To identify if a game engine could be repurposed to stage pervasive games (that make use of virtual game elements), a component feature set, for such an engine, has been distilled in the survey of Chap. 2. The feature set has been verified against the definitions of pervasive games and related work in Sect. 2.5. A virtual world engine has been selected, as candidate in the same product line as a would-be pervasive game engine in Sect. 3.4. To validate the resulting feature set and the chosen game engine, the pervasive game Codename: Heroes was implemented as proof-of-concept. CN:H was extended to support all features of the set. Although CN:H is not the first to implement a pervasive game using a virtual world engine (e.g., Ambient Wood (Thompson, Weal, Michaelides, Cruickshank, & Roure, 2003)), the production of CN:H gave needed first-hand experience, highlighting features of particular importance and any open issues. CN:H was successfully implemented, reaping the benefits of using the selected architecture; development time was low, spanning just a few months, with a third of the development resources spent on the game client. Although not all pervasive games will utilize all the features described, the aim is for the engine to support a wide variety of pervasive games. An additional aim for the feature set, is that it can be used to expand large scale virtual world engines into large scale pervasive game engines.

References

Akribopoulos, O., Logaras, M., Vasilakis, N., Kokkinos, P., Mylonas, G., Chatzigiannakis, I., & Spirakis, P. (2009). Developing multiplayer pervasive games and networked interactive installations using ad hoc mobile sensor nets. In *Proceedings of the international conference on advances in computer enterntainment technology* (pp. 174–181). New York, NY, USA: ACM.

Benford, S., Giannachi, G., Koleva, B., & Rodden, T. (2009). From interaction to trajectories: designing coherent journeys through user experiences. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 709–718). New York, NY, USA: ACM.

BigWorld. (2014). *Bigworld technology overview*. Retrieved from http://bigworldtech.com

Branton, C., Carver, D., & Ullmer, B. (2011). Interoperability standards for pervasive games. In *Proceedings of the 1st international workshop on games and software engineering* (pp. 40–43). New York, NY, USA: ACM.

Broll, W., Ohlenburg, J., Lindt, I., Herbst, I., & Braun, A.-K. (2006, October). Meeting technology challenges of pervasive augmented reality games. In *Proceedings of 5th ACM SIGCOMM workshop on network and system support for games* (p. 28). Singapore: ACM.

Demeure, I., Gentes, A., Stuyck, J., Guyot-Mbodji, A., & Martin, L. (2008). Transhumance: a platform on a mobile ad hoc network challenging collaborative gaming. In *Collaborative technologies and systems, 2008. CTS 2008. international symposium on* (pp. 221–228). Irvine, CA, USA: IEEE.

Dionisio, J. D., Burns III, W. G., & Gilbert, R. (2013). 3D virtual worlds and the metaverse: current status and future possibilities. *ACM Computing Surveys*, *45* (3), 34:1–34:38.

Greenhalgh, C., Izadi, S., Mathrick, J., Humble, J., & Taylor, I. (2004, September). ECT: a toolkit to support rapid construction of ubicomp environments. In *Proceedings of workshop on system support for ubiquitous computing*. UbiSys. Nottingham, UK: Springer Verlag.

Guerrero Corbi, V. (2014, July). *Development of a web based interface for game-masters of pervasive games*. (Master's thesis, Universitat Politècnica de Catalunya).

ISO. (2011). *ISO/IEC 25010:2011 systems and software engineering – systems and software quality requirements and evaluation (SQuaRE) – system and software quality models*.

Lewis, M., & Jacobson, J. (2002, January). Game engines in scientific research. *Communications of the ACM*, *45* (1), 27–31.

Niantic Labs. (2014). *Ingress*. Retrieved from https://www.ingress.com/

Oppermann, L. (2009, April). *Facilitating the development of location-based experiences*. (Doctoral dissertation, The University of Nottingham).

Paelke, V., Oppermann, L., & Reimann, C. (2008). Mobile location-based gaming. In *Map-based mobile services- design, interaction and usability* (Chap. 15, pp. 310–334). Springer Berlin Heidelberg.

Singhal, M., Chandrasekhar, S., Ge, T., Sandhu, R., Krishnan, R., Ahn, G.-J., & Bertino, E. (2013). Collaboration in multicloud computing environments: framework and security issues. *Computer*, *46* (2), 76–84. doi:10.1109/MC.2013.46

Thompson, M. K., Weal, M. J., Michaelides, D. T., Cruickshank, D. G., & Roure, D. C. D. (2003). *MUD slinging: virtual orchestration of physical interactions*. ECSTRIAM03-007.

Yahyavi, A., & Kemme, B. (2013). Peer-to-peer architectures for massively multiplayer online games: a survey. *ACM Computing Surveys (CSUR)*, *46* (1), 9.