Considerations on the Acquisition and Preservation of eBook Mobile Apps

Maureen Pennock
The British Library
United Kingdom
maureen.pennock@bl.uk
https://orcid.org/0000-0002-7521-8536

Peter May
The British Library
United Kingdom
peter.may@bl.uk
https://orcid.org/0000-0001-8625-9176

Michael Day
The British Library
United Kingdom
michael.day@bl.uk
https://orcid.org/0000-0002-1443-5334

Abstract – In 2018 and 2019, as part of the UK Legal Deposit Libraries' sponsored 'Emerging Formats' project, the British Library's digital preservation team undertook a program of research into the preservation of new forms of content. One of these content types was eBooks published as Mobile Apps. Research considered a relatively small number of apps in an attempt to better understand the preservation challenges associated with them and make recommendations for a way forwards. It found that whilst the content landscape is extremely varied, the technical challenges are similar for both Android and Apple apps. The greatest challenges appear to lie in the acquisition and access areas, particularly in enabling delivery to appropriate rendering environments and devices, though other challenges remain around the use of device sensors and content stored on remote servers rather than the user's device. Whilst these challenges may be surmountable with sufficient resource and investment, the lack of growth in this content area in recent years makes large scale investment questionable.

Keywords – mobile apps, challenges, access, acquisition, digital preservation

Conference Topics – Exploring New Horizons

I. Introduction

Mobile device ownership has increased dramatically over the past two decades. In 1996, when the Office of National Statistics first began collecting data on mobile phone ownership in UK households, only 16% of homes in the UK were known to own a mobile phone. [1] Twenty years later that figure had risen to 95%, many of which were 'smartphones', i.e. highly advanced phones with computer and internet capabilities, touchscreen, and other built-in components such as GPS or gyroscopes.

Smartphones began to appear in the early 2000s and allowed users to download, install, and run programs or 'apps' directly on their mobile phone. In the same manner, apps could also be downloaded to mobile 'tablet' computers, the popularity of which was similarly increasing. Early apps were often productivity or game based, but by the early 2010's traditional publishing houses had begun to explore how smartphone apps could be used to deliver new literature experiences. Faber and TouchPress (now known as Amphio) were early adopters, launching T. S. Eliot's 'The Waste Land' app in 2011 to widespread acclaim [2]. More apps followed, most of which had a clear link to existing analogue publications, but 2015 saw Amphio release a groundbreaking new work conceived wholly to be experienced as an app: Ian Pears' 'Arcadia'. During this time several other publishing houses began to deliver book-like content in app form, including Penguin, Random House, Nosy Crow, and Oxford University Press. Software houses such as Inkle and Simogo, alongside other smaller independent parties, also began to release book-like apps in which the textual work was the primary content type. Collectively, publishing and software houses enhanced the world of electronic books with a new way to experience titles both old and new.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

This 'new way of experiencing titles' poses a challenge for memory organizations tasked with preserving cultural heritage and electronic literature. The app genre is relatively unknown in most collecting libraries, and the overall eBook mobile app content landscape poorly mapped – how do we even begin to deconstruct the challenge and how should an organization know where to begin? If memory organizations are to collect and preserve apps, what technical issues must be addressed, and what access issues must be resolved? This paper presents work undertaken by the British Library as part of the UK Legal Deposit Libraries' 'Emerging Formats' project in an effort to begin answering those questions.

II. Exploring the Genre

Deconstructing the challenge requires a thorough understanding of the content type. During its first year, alongside exploration of other types of 'emerging formats', the project analyzed just three mobile app titles: The Solar System (Faber/TouchPress, first released in 2010); The Waste Land (Faber/TouchPress, first released in 2011), and Goldilocks and Little Bear (Nosy Crow, first released 2015). All of these were published on the iOS platform and have already been discussed within the larger context of the 'Emerging Formats' project. [3] The second year of the project increased the size of the sample in order to provide greater insight into the different types of eBooks published as mobile apps, the range of content features found within different types of eBook mobile apps, and experience of apps developed for consumption on an Android device. This increase allowed for a more in-depth evaluation of the challenges of collecting, preserving and providing access to mobile apps.

A total of twenty-two apps were selected for analysis in year two of the project. Each was chosen by content experts as an exemplar eBook mobile app, representative of different categories of content, different publishers, or notable particularly for award-winning features. Content was accessed either on a Samsung S7 phone running Android 8.0 or one of two Apple iPads running iOS9 and iOS11. Apps and their features are discussed by category, below.

A. Interactive narrative apps

Interactive narrative mobile apps are defined here as works which require choices to be made by the reader in order to establish the direction of the story, which can vary meaningfully between readers and reading experiences.

Text, particularly in narrative form, was the driving feature of all apps in this category. Some apps, such as Ian Pears' 'Arcadia' from Amphio [4], had a complex branching narrative with relatively few additional features beyond navigational aids, for example in the form of a map or index. Others such as '80 Days' from Inkle [5], 'The Kai Chronicles' from Project Aon [6], or 'The 8th Continent' from Ben Garrett [7], had significantly more game-like features built around the narrative, including functionality to accumulate and trade items, accrue funds, maintain health scores, and engage in combat. This increased functionality was often accompanied by extensive imagery, both background and foreground, and most apps in our sample also included background audio tracks. Advertisements appeared on occasion, particularly within free apps with a heavy emphasis on gameplay.

The apps in this category seemed to make limited use of built-in hardware features such as gyroscope, camera or microphone, and progress was typically made by swiping or tapping the screen.

B. Young children's apps

eBook apps for Young Children are understood here as apps aimed at young and early readers. All apps in this category made extensive use of color and imagery (as one might expect with children's literature). Most of the sample originated from Nosy Crow publishing: 'Snow White' [8] and 'Jack & the Beanstalk' [9] both provided a modern re-telling of a classic fairytale, whilst the 'Rounds' [10] series introduced young readers to the lifecycle of animals and 'Axel Scheffler's Flip Flap Pets' [11] was a modern take on traditional 'lift the flap' books. 'The Flitlits: the Funny Fair' [12] from Flitwits Ltd was the only work examined here from a publisher other than Nosy Crow, providing a bi-lingual exploration of a magical land that could be used to support teaching of the England and Wales Key Stage 1 and Key Stage 2 educational curriculum.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

All Nosy Crow apps were highly interactive. Tapping and swiping the device touchscreen were the most frequent ways to progress the story or complete in-book challenges and games, though some apps also made use of the device gyroscope, camera and microphone. All featured background music and the option for an automated 'read aloud' experience, which highlighted subtitled phrases on the screen when sounded out by the corresponding audio track. Some of the Nosy Crow apps included tracks from award-winning musicians or images from famous illustrators. In comparison, the single FlitLits app analyzed had limited potential for enhanced interaction, though the audio and textual tracks were available in English, Welsh, and American.

C. *Simple language apps*

In a similar manner to the young children's apps, the simple language apps analyzed were visually appealing, with heavy use of images and color, and relatively little text. 'Go!Gaelic'[1] was one such example from the Learning Centre of North Lanarkshire Council, which included a very limited range of English words and phrases that could be translated into Gaelic. The Interactive Welsh alphabet app 'Cyw a'r Wyddor', from Welsh-language broadcaster S4C [13], also contained a limited amount of content intended to support young children learning the language anew. Both apps made frequent use of audio, helping users to understand how translated words were pronounced. The interactive functionality of these apps was limited mainly to querying and listening to audio playbacks.

D. *Music apps*

Music apps are defined here as apps where music is the main feature of the app rather than a supporting component. Apps analyzed in this category included 'Beethoven's 9th Symphony' from Amphio [14] and 'Tido Music' from Tido [15]. Whilst music was the key feature in these apps, it was expressed in many different ways from audio-visual recordings, musical notation and descriptive textual information, to digitized content and graphical visualizations. One app, 'Steve Reich's Clapping Music' from Amphio [16], included a game as a central feature, played by tapping the touchscreen in time with the music.

The appearance and functionality of apps in this category varied, though a notable shared characteristic was that not all content was contained in the app package, with each requiring an internet connection or a fee to be paid in order to unlock access to additional content.

E. *Reference works*

Structured textual data was the predominant feature of the two reference works analyzed, though in some instances it was accessed directly from an online database rather than downloaded to the user's device. The 'Oxford Dictionaries Quick Search' app from Oxford University Press and MobiSystems [17], for example, included over 350,000 words and meanings, but whilst the full (ad-free) version cost £27.99 per year and included the option to download the entire database to your own device, the free version of the app provided limited access and required connection to a remote database. The GPC (Geiriadur Prifysgol Cymru) 'Welsh dictionary' app from the Centre for Advanced Welsh and Celtic Studies [18] offered similar install options, but in this case the full database could be downloaded directly to the app for free. Functionality was primarily limited to general searching and querying, though the Oxford Dictionary also retained a search history and allowed users to 'favourite' words.

F. *Miscellaneous apps*

A small number of apps did not fit easily into any of the categories identified. These displayed features akin to those seen across the other categories, but without some key characteristics. For example, 'American Interior' from Penguin Random House [19] and 'The 39 Steps' from Faber & TouchPress [20] were both narrative-based but absent of interaction that influenced development of the storyline, whilst the multimedia content seen in 'Shakespeare's Sonnets' [21] app was akin to that in Amphio's 'Beethoven's 9th Symphony' app [14], but with literature rather than music as the key feature.

III. Technical Dependencies

As is evident from the previous section, apps can contain an extremely wide range of content and functionality. The method by which this is packaged together and delivered to devices for consumption by readers is a container format: for iOS this is the

[1] Go!Gaelic was withdrawn from both the Apple App store and the Google Play store in early 2019 for 'critical updates'

IPA format; for Android it is the APK format. These containers hold not only the content that will populate the app, but also metadata and the software program (or application, from which 'app' is clearly derived) that will run on the access device's operating system and allow users to experience the content.

The structure of both container formats is well defined within official developer documentation. Both are based on the ZIP format, thought the APK specification is more closely aligned to the JAR format (a zip-like container format used to package together java class files, metadata and other resources prior to their distribution), and both can be unpacked on a desktop computer by simply amending the file extension to ZIP.

Both container formats are tightly linked to their relative platforms for deployment: IPA apps can only be run in an Apple environment, and APK in an Android environment. Within those environments, apps are optimised for specific versions of operating systems. Each operating system iteration typically brings about new APIs available for use by apps and together these essentially create a minimum version of the operating system environment that a given app will work in. Development best practice is to target the earliest version of the operating system that supports all the APIs required, in order to maximize the number of devices that will support the app in development. Both Android and Apple operating systems are updated relatively frequently. There have been 17 major versions of the Android operating system since its first release in 2008[1], and 12 releases of Apple's iOS since 2007[2], though early versions were limited to phones only. This rapid rate of development poses a challenge for apps with specific operating system dependencies, as these apps may not function as intended once a device's underlying operating system is updated.

Versioning dependencies should be documented in the app metadata. Specific hardware dependencies (including those upon built-in hardware features such as camera and microphone) can also be documented within the metadata file, though developers may not always utilise this feature. The handheld hardware market for Android is more varied than Apple and as a result the broader scope of hardware options means that some apps may not have full functionality (or even work) on all Android devices, which may explain the difficulty we had in getting one particular app to work properly. The dependency between apps and supplementary peripherals such as the iPad Pencil needs further research, but was not an issue for the content considered during this assessment.

The Google and Apple validation services, a pre-requisite for publishing apps via the respective app stores, should theoretically mean that all content types packaged within the app can be rendered on a subset of Android and Apple environments contemporary to the app's publication, as identified in the app metadata. Whilst all apps in our assessment were downloaded from the official app stores, some Android apps can also be downloaded from 'unknown' or 'unauthorized' sources. These may not have been through this validation check.

IV. Availability of Preservation tools

A. *Validation and identification*

Both APK and IPA files can be identified by Apache Tika. Neither currently have PRONOM entries so cannot be identified by tools reliant on the PRONOM database. APK files typically use the mime type application/vnd.android.package-archive, whilst IPA files might use variations of the MIME type application/zip, application/octet-stream, or a purposefully defined MIME type. The depth of validation applied by Google and Apple in their validation service (described above) has not been reviewed from a preservation perspective so it is unclear to what depth (i.e. individual files?) this may apply.

B. *Metadata extraction*

XML metadata is held in both IPA and APK containers. Our sampling indicated that much of it is human-readable so it could theoretically be extracted and used within preservation workflows without significant difficulties, assuming sufficient technical skills are available to write the requisite parser.

[1] As outlined at https://en.wikipedia.org/wiki/Android_version_history

[2] As outlined at https://en.wikipedia.org/wiki/IOS_version_history

C. *Migration and Emulation*

The research identified no tools that could easily unpackage and migrate IPA or APK apps for use in alternative environments. App updates that are undertaken to maintain compatibility with new versions of operating systems are typically done directly by the developer or publisher. These are often based on results from the compatibility testing function within the Software Development Kit (SDK). It is not possible for the standard user to 'update'apps in this way, though users can download new versions from the app store once the developer has released an update.

Both Android and Apple SDK's provide emulator/simulator software designed for accessing app content on a PC for development and testing purposes. The Android emulator, for example, allows developers to test apps on a variety of device configurations (e.g. screen resolutions and display sizes) and Android API levels without needing to have each physical device. The emulator provides many of the capabilities of a real Android device and can be installed via the SDK tools tab of the SDK Manager. Many other emulators are also available that would allow Android apps to be run on a PC, such as that from the Android x-86 open source project[1]. Initial experiences of using this emulator to access apps at the Koninklijke Bibliotheek in 2014 indicated that whilst it had potential, more work was required to understand its limitations. [22] Questions are also yet to be explored regarding integration of emulator software into a reading room environment and user experiences.

The iOS SDK simulator is used for development and testing purposes. This could in theory be used to deliver content on a PC but would require permission from the publisher and extensive user testing. There would also be challenges in enabling the simulator independently of the full SDK environment. The iOS operating system is closed and highly proprietary, limiting the availability of alternative emulator sources.

V. Discussion

The technical dependencies identified above, and the limited availability of preservation tools, make preserving this type of content a challenge. Furthermore, the rate at which app operating systems are updated is significantly more frequent than the rendering environments used to provide access to our other digital collection content. This has clear implications for the frequency of preservation planning activities and associated resource requirements. Any preservation solution for this content type would need emulators for all versions of the operating systems upon which a given collection of apps is dependent.

The limitations of the current generation of emulators must also be considered, particularly if they are to be deployed in a reading room environment utilising desktop computers rather than handheld devices. Emulators such as those used within Software Development Kits typically allow the developer to develop and run programmes within the emulator that generate 'mock' device functionality such as GPS or a gyroscope. Whilst this is sufficient for testing purposes, it would not work as a functional GPS or gyroscope were the emulator be deployed in a live environment. Features of an app which relied on these components would therefore not function correctly. Several of the apps we reviewed also had a dependency on built-in hardware components such as the camera and microphone, and most used the device speakers to deliver audio content. These components are more likely to be supported by a desktop machine so it could be possible to tap directly into that functionality, though it is likely this would need further development – alternatively we could take a parsimonious approach and wait for a future state emulator to emerge from the development community, though the risks of that are clear.

We note, however, that aside from this, the problem of app dependencies on particular processor architectures is an as yet unresolved issue. As discussed by Viticci [23], iOS 11 (released in Sept 2017) *only runs on modern 64-bit devices and doesn't support legacy 32-bit apps*. Viticci even notes that after the introduction of iOS 11, 'the company [Apple] began preventing apps that hadn't been updated for 64-bit devices from being launched altogether, thus rendering 32-bit apps that were still installed on users' devices useless', and that 'the biggest problem facing App Store preservation

[1] See http://www.android-x86.org/ for more details

today is the lack of any kind of emulation for old 32-bit titles that are no longer supported on modern hardware.' The proprietary nature of iOS remains at this point an unresolved challenge that limits the viability of emulation as a preservation approach for IPA apps and which will require positive engagement with Apple to reach a satisfactory conclusion.

Setting aside the challenge of access for a moment, we should remind ourselves that an access solution without content to access is no solution at all. The first line of defence against content loss is acquisition of the object, preferably in the most robust and complete form available.

Users typically acquire apps by downloading them from the app store directly to their device. This is a satisfactory approach for the immediate usage of apps, but it would require a device to be networked in order to subsequently transfer content to the repository, and even then may need third party software to perform this task. Download in this fashion also typically introduces Digital Rights Management (DRM) to an object, which is subsequently used to limit end user access to the object – this is certainly the case for IPA apps though further research is required to clarify the situation with APK apps. Acquisition-related DRM would cause problems should the content subsequently need to be accessed by different users (as likely in a reading room environment), and certainly also over the long-term when the platform's authentication mechanism is eventually changed.

An alternative approach is to consider the app primarily as software – as previously noted, apps are after all a combination of software program and content together in a single container – and to target the acquisition towards an earlier stage in the production process. Acquisition of source code direct from publishers would be one such way to avoid issues introduced by downloading content from an app store, though it would require additional deposit of a compiler in order to generate a usable app for access by readers. Whilst in theory, multiple compiler deposits supporting different platforms could allow organizations to subsequently produce apps for different platforms, in practice the source code is typically designed for either Android or iOS: the same app running on both Android and iOS is typically two separate code bases (as they are two different languages). Apps coded specifically for cross-platform development, using meta-languages to build apps that can then be cross-compiled to both Android and Apple apps, are likely to be more suitable for this process, though each app would still require production of a compilation script for each different operating system, requiring a high degree of technical skill to produce.

The problem with this is that each representation is essentially a new Intellectual Entity[1]. If an app is considered primarily as a software program, and that program is modified to work with a new version of Android (for example) then this may introduce changes to the functionality of the app: if, for example, cross-compilation from Android to iOS is performed, an organization has essentially then created something completely new. The implications of this on the perceived authenticity of the object need to be more fully considered.

Should a collecting organization pursue this approach regardless, the acquisition of technical documentation about the app would be necessary, as noted by both the National Film and Sound Archive of Australia [24] and the Library of Congress [25]. Ultimately, a 'belt and braces' approach – i.e. acquisition of the published app, acquisition of source code, compiler(s), emulators, and associated technical documentation - would keep the most options open for different access solutions at a later date, but for many organizations this may be more than is feasible.

Even then, our content analysis indicates that preservation may still not be assured. If the app is to be acquired in the most robust and complete form possible then we must find some way to

[1] 'Intellectual Entity' is an Open Archival Information System (OAIS) term defined within PREMIS as 'A set of content that is considered a single intellectual unit for purposes of management and description: for example, a particular book, map, photograph, or database. An Intellectual Entity can include other Intellectual Entities; for example, a Web site can include a Web page; a Web page can include an image. An Intellectual Entity may have one or more digital representation'. See http://www.digitizationguidelines.gov/term.php?term=intellectualentity for more information.

deal with apps which have an inherent reliance on content hosted externally to the app. These are likely to lose their integrity over time, particularly as linkage to archived web content does not yet (if at all) appear to have become standard practice in apps. Such a problem would certainly manifest significantly in apps that require a subscription (such as the 'Tido Music' app); other apps which require an active connection to an online resource would also be affected, though the degree to which this impacted on the end user and the object integrity would be variable dependent on the importance of the content affected. Loss of access to the online resource would be a high issue in, for example, the free version of the Oxford English Dictionary app. The '8th Continent', on the other hand, which used an active internet connection only to deliver advertisements, would be less badly affected. If collecting at scale, then identifying those apps with an inherent reliance on externally hosted content could be an issue of its own. If we consider this issue alongside the limitations of working directly with downloaded content that were identified earlier in this section, collectively they suggest that direct deposit of app content from the publisher is likely to result in a more 'preservable' item than if we seek to acquire content using a harvesting approach of the sort used to gather content for web archives.

Finally, we must consider the future of the mobile eBook app market. How big is the market already, and in which direction might it evolve going forwards? A report produced for the British Library in 2017 suggested that 'the total number of UK Apps which are in effect books or are book like, is probably no more than thousands and certainly less than tens of thousands'. [26] However, this figure is a cumulative one, and several are likely to have already become unavailable or unworkable. Even during the course of this project, one of our sample apps was withdrawn and another could not be made to function properly. Informal discussions with publishers has suggested that if anything, the app market is shrinking, with several withdrawing from the app market due to the high-costs of development and maintenance, especially when compared to standard eBook titles. There is also a concern that it may be difficult to ensure mobile eBook apps comply with new EU accessibility rules, and some publishers have reported an interest in ePub4[1] as an alternative. Utilizing ePub4 would effectively move us to a scenario where content is "streamed" to mobile devices (rather than held within an app), a scenario which may become more prevalent particularly with 5G mobile services. This is a different preservation challenge again. Put simply, apps may be too costly to develop and support in all but high value/niche markets. How much then should organizations invest in the development of a solution to support this type of content?

VI. Conclusion and Recommendations

Preservation of mobile apps is a relatively under-explored subject within the field of digital preservation literature. We have sought to begin to remedy that in this paper by sharing our experiences with a sub-set of the eBook mobile app genre, our exploration of the technical environment in which this type of content exists, and the preservation issues that institutions will face should they choose to begin collecting this type of content.

Though the range of content and functionality one might find in an app is not dissimilar from the range of content to be found, for example, in a web archive or a personal digital archive deposit, mobile eBook apps are certainly far removed from a typical PDF eBook. The mobile eBook apps reviewed in this research may be considered to have more similarities with eBooks in an ePub format, particularly given that the ePub format is also a container that can mask significant 'under-the-hood' complexity, but the executable combination of both content files and computer program in an app set it apart from most of our other digital collection types. Should it therefore first and foremost be understood – and preserved - as software? The many technical dependencies identified and discussed in this paper, and the challenges these subsequently pose for collecting institutions, would suggest that this would be a logical approach. Unlike mobile apps, the subject of software preservation has received significant attention in the digital preservation field over recent years, for example,

[1] ePub4 is described by EDR Lab, a W3C member involved in development and promotion of the standars as a 'Packaged Web Publication', see https://www.edrlab.org/epub/introduction-to-epub-4/ for more details.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

via the Software Heritage Foundation[1] and the Software Preservation Network[2], and it is perhaps in that direction that we must turn in order to identify a way forwards.

What then are our next steps and what recommendations can we make in order to begin preservation in the meantime? In terms of target formats for acquisition, we reach the undeniable conclusion that acquisition of the app in its packaged form (either an IPA file or an APK file) is optimal for ensuring organisations at least acquire a complete published object for preservation. Whilst this is a compressed form and would limit organisations to delivery of the app on a specific version of iOS or Android, this form should at least also include sufficient metadata about inherent technical dependencies to understand what is needed to meet them. Avoidance of embedded DRM in the apps may be best achieved by direct publisher deposit, as opposed to download, and direct deposit would further provide opportunity for ensuring that apps which are acquired are in the most complete and rigorous form possible – i.e. free from subscription or dependencies on significant content hosted remotely.

Some organizations may choose to acquire source code rather than compiled apps. In this scenario, acquisition of the uncompressed source code of the app could potentially allow more flexibility for updating and maintaining the app for use on different platforms and different versions of operating systems, though this would require access to appropriate technical skills and is likely only to be effective when apps are coded initially for cross-platform purposes. Dependent on the end access solution, apps may still need to be authenticated before they can be installed, and the limits of emulator-based access, particularly on a fixed PC, need to be carefully considered: will it ultimately provide end users with a sufficiently 'authentic' experience? If not, then should wholly alternative solutions such as recording or documentation be more seriously pursued? We recall also the observations regarding uncertain growth of this content type. It is not unfeasible to consider a scenario whereby collection of mobile eBook apps remains relatively small scale. Small scale collecting would, particularly for larger collecting organisations where this content type is only a small percentage of the overall collection, suggest that a non-standard approach to access could be envisaged, for example on dedicated and non-standard reading room terminals, reducing the need for potentially costly technical solutions and integration.

Acquisition of as many technical components as possible, including the published app, source code, compiler(s) and emulators, alongside associated technical documentation, would seem to keep the most options open for different access solutions at a later date. Engagement with software providers will be essential however in ensuring that content is acquired in its most complete form and that emulators are available for use as needed. Despite this, the functionality represented in an emulator is one step removed from actual device function, and reservations regarding the use of emulation to deliver an 'authentic' user experience still need to be more thoroughly considered. Expanding our sample and technical analysis from year one to year two of the project has been invaluable in highlighting these kinds of issues.

[1] The Software Heritage Network focuses on acquisition and preservation of publicly available source code. For more information see https://www.softwareheritage.org/#

[2] The Software PreservatioN Network is a US-based organisation established to 'advance software preservation through collective action'

Acknowledgments

Our thanks go to Ian Cooke, David Russo, Caylin Smith and Neil Wilson for their perspectives on some of the issues raised in this paper during the course of the Emerging Formats project.

References

[1] Statista, 'Percentage of households with mobile phones in the UK from 1996 to 2017', (data derived from Office for National Statistics), January 2018, https://www.statista.com/statistics/289167/mobile-phone-penetration-in-the-uk/

[2] Stuart Dredge, ' The Waste Land iPad app earns back its costs in six weeks on the App Store', article in The Guardian, 8 August 2011, https://www.theguardian.com/technology/appsblog/2011/aug/08/ipad-the-waste-land-app

[3] M. Day, M. Pennock, C. Smith, J. Jenkins, and I. Cooke, "Preservation planning for emerging formats at the British Library," in Proceedings of the 15th International Conference on Digital Preservation (iPRES 2018), Boston, MA, USA, 24-27 September 2018. https://doi.org/10.17605/OSF.IO/58PJV

[4] Ian Pears, 'Arcadia', published by Amphio and available on the Apple app store at https://itunes.apple.com/gb/app/arcadia-by-iain-pears/id1003843523?mt=8

[5] Inkle Studios, '80 days', available for multiple platforms via https://www.inklestudios.com/80days/

[6] Project Aon, 'The Kai Chronicles' (text written by Joe Dever), available from the Google Play store at https://play.google.com/store/apps/details?id=org.projectaon.kaichronicles&hl=en_GB

[7] Ben Garrett, 'The 8th Continent', available for multiple platforms from http://www.8thcontinent.co.uk/

[8] Nosy Crow's 'Snow White', available from the Apple app store at https://itunes.apple.com/us/app/snow-white-by-nosy-crow/id934867093

[9] Nosy Crow's 'Jack and the Beanstalk', available from the Apple app store at https://itunes.apple.com/us/app/jack-and-the-beanstalk-by-nosy-crow/id796383629?mt=8

[10] The Nosy Crow Rounds series, described at https://nosycrow.com/series/rounds/

[11] Nosy Crow's 'Axel Scheffler's Flip Flap Pets', available from the Apple app store at https://nosycrow.com/series/rounds/

[12] Flitwits Ltd, 'The FlitLits: The Funny Fair', available from the Apple app store at https://itunes.apple.com/gb/app/the-funny-fair/id815547838?mt=8

[13] S4C, 'Cyw a'r Wyddor', available from the Apple App store at https://itunes.apple.com/gb/app/cyw-ar-wyddor/id643054131?mt=8

[14] Amphio Limited, 'Beethoven's 9th Symphony', available from the Apple App store at https://itunes.apple.com/gb/app/beethovens-9th-symphony/id601942399?mt=8

[15] Tido (UK) Ltd, 'Tido Music: learn, play & sing', available from the Apple App store at https://itunes.apple.com/gb/app/tido-music-learn-play-sing/id1050926313?mt=8

[16] Amphio Ltd, 'Steve Reichs Clapping Music', available from the Apple App Store at https://itunes.apple.com/gb/app/steve-reichs-clapping-music/id946487211?mt=8

[17] Oxford University Press / Mobi Systems, 'Oxford Dictionaries Quick Search' app, available from the Google Play store at https://play.google.com/store/apps/details?id=com.mobisystems.msdict.embedded.wireless.oxford.dictionaryofenglish&hl=en_GB

[18] GPC 'Welsh Dictionary' app, available for multiple platforms from http://www.welsh-dictionary.ac.uk/android-ios-apps/

[19] Penguin Random House, 'American Interior', available from the Apple app store at https://itunes.apple.com/bw/app/american-interior/id857802114?mt=8

[20] Faber & TouchPress, 'The 39 Steps' app, available from the Apple Mac store at https://itunes.apple.com/tz/app/the-39-steps/id999827755?mt=12

[21] TouchPress Inc, 'Shakespeare's Sonnets' app, available from the Apple App store at https://itunes.apple.com/gb/app/shakespeares-sonnets/id528646395?mt=8

[22] Johan van der Knijf, 'Running Android apps on a PC: first impressions', published on the OPF blog, October 2014, https://openpreservation.org/blog/2014/10/23/running-archived-android-apps-pc-first-impressions/

[23] Federico Viticci, 'App Preservation: Saving the App Store's History', July 2018, https://www.macstories.net/stories/app-preservation-saving-the-app-stores-history/

[24] S. Langley, T. Carter, M. Davies, and I. Gilmour, "Managing multi-platform materials: selected case studies," in Proceedings of the 19th International Symposium on Electronic Art (ISEA2013), Sydney, Australia, 2013. https://ses.library.usyd.edu.au/handle/2123/9706

[25] Library of Congress Recommended Formats Statement 2018 – 2019, https://www.loc.gov/preservation/resources/rfs/

[26] J. Ackroyd, N. Canty, A. Watkinson, 'Research to support the British Library's work on 'Emerging Formats', CIBER internal report to the British Library, March 2017.