Implementing large-scale computer-based assessment in schools

Maurice Walker

Australian Council *for* Educational Research

Abstract: This paper is directed at policy makers, administrators and assessment developers contemplating the introduction of computer-based assessment at a regional or national level. It draws heavily on ACER's experience in organising computer-based assessments for the OCED's Programme for International Assessment (PISA), which commenced with a limited computer-based assessment of science in 2006 and progressed through the computer-based assessments of digital reading in 2009 and of mathematics and problem solving in 2012. This experience is unique in that no other computer-based assessments of similar scope and complexity that assess students in schools have been organised to date.

Delivering computer-based assessment requires both hardware and software infrastructure. The choice of implementation model is constrained by a number of practical considerations, in particular whether the schools involved possess appropriate hardware and software, what alternatives to school-based infrastructure are available, and the level of security that is required. Three implementation models are described: internet delivery from an external host, portable applications that run under a computer's native operating system, and "live systems" that (temporarily) replace the computer's resident operating system. Each model is discussed with regard to its local and external infrastructure needs, the level of security it provides, and its relative advantages and disadvantages when compared with the other two models.

The paper includes an outline of architectural issues, such as assessment navigation, timing, and accessibility that impact are usefully considered from the onset of planning for a large-scale computer-based assessment.

Keywords: assessment methods; large-scale assessment; computer-based assessment; assessment design.

Introduction

As digital technologies have advanced in the 21st century the demand for using these technologies for large-scale educational assessment has increased. There are four widely recognised benefits to using computer-based methods for administering an assessment: i) computer-based methods facilitate a wider coverage of assessment content and can efficiently evaluate a wider range of cognitive processes than paper-based methods; ii) computer-based methods stimulate testee motivation through devices such as animation, interaction and surprise; iii) computer-based methods facilitate control of the assessment workflow and accommodate complexity of assessment design; and, iv) computer-based methods realise resource and administrative efficiencies (Walker, forthcoming; Yan Piaw, 2012)

Along with the advance of digital technologies, an abundance of technical solutions for assessment delivery has arisen, and the technology landscape in the school sector has diversified. This paper is intended to be a useful guide to those policy makers, administrators and assessment developers wanting to travel down the computer-based assessment route. The focus is on the initial design issues that will benefit program coordination when addressed early in the planning process. Three current alternative implementation models for large-scale computer-based assessments are described in the light of local infrastructure requirements and security considerations.

The examples presented in this paper derive from the Programme for International Student Assessment (PISA) run under the auspices of OECD and implemented in the first five cycles (2000 through 2012) by the Australian Council for Educational Research (ACER). The PISA studies provide highly relevant examples when considering implementation options for computer-based assessments as a large range of prior technical requirements and infrastructure across countries must be accommodated.

International comparative student assessment studies such as PISA are typically representative sample surveys[1] of school students from a broad variety of schools: spanning the geography of the countries, the languages of instruction within and across countries, with differing socio-economic contexts and with different levels of digital technology infrastructure. In some countries, all schools have computers and an internet connection but in others there are many schools that lack internet connections and some which do not even have computers. Existing infrastructure issues impact on the choices of whether and how to undertake computer-based assessments.

The level of test security required also impacts on the choice of implementation model. PISA, like many international and national assessments is a secure assessment. The goal is to keep the assessment material out of the public domain so that it may be used in subsequent assessments[2] without the contamination of prior engagement with the items.

[1] Very small countries take a census of students in order to attain the same degree of certainty as larger countries' population estimates.

[2] It is important that security is maintained in PISA because, firstly, the assessment survey window often spans many weeks within a country and students in deferent schools take the same test at different times; secondly, implementation windows for the same assessment vary across countries in a given assessment round; and, thirdly, the majority of material in any assessment round is reused in subsequent assessment rounds for equating/trend analysis.

Implementation models

There are currently several models available to implement computer-based assessment in large-scale assessments of school students. The choice of implementation model is guided by three practical considerations:

1. To what extent can participating schools be relied upon to provide appropriate hardware and software infrastructure needed to run the assessment?
2. What alternatives to school-based infrastructure are available?
3. How secure does the assessment need to be?

Delivering computer-based assessments requires a hardware and software infrastructure. At the very least, there needs to be some sort of computer (screen, keyboard, mouse, hard drive) and operating system. An internet connection, USB port, drawing tablet or other peripheries may be required. This infrastructure may be provided partly by the school, partly or wholly by the organising study centre, or wholly through specially set up local testing centres.

Three implementation models are outlined below: internet delivery, delivery via portable application, and live system delivery. Each model is discussed with regards to its infrastructure needs and assessment security.

Internet delivery

Delivering the assessment entirely through the internet initially appears a very attractive model. In this model there is no need to deliver any physical material to the school. Students simply log in to the assessment via the internet and they use their own school's hardware provided it has an appropriate web browser (if required) and a stable internet connection with appropriate dedicated bandwidth.

Internet delivery means that one or more of the elements of a test are transmitted to the computer via the internet at test-time. These elements are test content (items and stimuli) and a test execution environment (runtime). A test could be delivered using pre-existing software on the host computer, like a browser, or by downloading a specialised test runtime (e.g. a Java Web Start application) and executing it directly to run the test.

Internet delivery uses the host operating system, meaning that it can access the local font set and input methods, and it ensures that all peripherals that will be used (e.g. screen, keyboard) will be recognised. This is a significant advantage in an international study as there is an enormous diversity in:

- fonts–including Asian character sets, fonts with unique diacritics (e.g. Polish) and bidirectional text (as in Arabic and Hebrew);
- character input methods (especially for Chinese and Japanese); and
- hardware (especially keyboards, screens and video cards, all of which require device drivers, some of which may be non-standard).

Another advantage of internet delivery is that test results can be transmitted directly to a central data collection centre in real time. Items with a finite set of predictable responses (such as multiple choice or other closed response items) can be automatically coded[3] and

[3] Coding is the term for the first stage of the scoring process. For example, a response to a 4 option multiple choice item might be coded 1,2,3 or 4 (or 'missing' or 'invalid'); then the key is scored as 1 and the distracters (plus missing and invalid) are scored as zero.

processed without the student data ever having to be handled by the test administrator or national study centre. Items requiring expert judgement to score can be collated at an international centre and made available to national centres, for example through the use of an online coding system.

However, there are several caveats to consider when considering internet delivery. Perhaps the most important issue is test security. Internet delivery is the least secure of the implementation models outlined in this paper. Even with sophisticated keyboard lock-down procedures, students are often still able to access host applications, meaning, firstly, that cheating is possible and, secondly, the test material itself is not secure (i.e. it can be copied and stored). Computers running an internet delivered assessment must of course have access to the internet and this means that the transmission of secure information from tests to a worldwide audience is possible.

A further security consequence when delivering the test via internet is that the host system is subject to potential cyber crime. Threats include access to and theft of confidential materials, the installation of malware, and denial of service attacks.

It is likely that a computer-based assessment will be programmed for optimal execution in a limited range of browsers (perhaps even just one). If the test execution environment is the local browser, the issue of the diversity of browsers in participating schools should be considered. Apart from any technical reasons for a single browser delivery it is important that the tests are viewed consistently by students around the world: that is, the tests should have the same 'look and feel'. The degree to which such standardisation can be compromised to allow for a variety of browsers to view the test should be considered in the early stages of development.

A related issue here is the continual update of browser versions that may render the original test programming obsolete after just a few months. It would be wise to have the appropriate browser available for download for schools, though this may take some negotiation with those responsible for the school's IT infrastructure.

While simple computer-based assessment content, such as static stimuli with multiple choice options, demand relatively little processing power to render and manage. On the other hand, state-of-the-art assessments involving animations and complex interactions between the testees and graphical elements are resource intensive. To date, a significant hurdle to internet delivery of 'content heavy' assessments are requirements for a reliable and dedicated internet connection of high bandwidth. At the time of writing, meeting these requirements is not a realistic expectation for all schools within a country. Even schools having internet connections with high bandwidths can experience difficulties due to concurrent usage in the school or surrounding area and external interruption to service.

Another disadvantage, or at least challenge, is the technical infrastructure needed to host the internet delivery model in large-scale assessments. With several thousand students potentially online at any one time, the server resources and their efficient configuration is costly. Increasingly, cloud hosting options can mitigate these costs as large amounts of existing host resource can be hired for short periods. However, cloud based hosting does potentially come with an additional security risk in that the physical location of host servers is often not known and while security may be contractually guaranteed the security audit process is not often accessible to the client and cloud solutions have frequently been criticised for lacking the security they claim (Hickey, 2010; Jackson, 2013).

Having mentioned the main advantages and disadvantages of internet delivery it is important to recognise that many of the disadvantages apply only when relying on school infrastructure. Computers in dedicated testing centres or carry-in laptops with internet access could be configured in such a way as to eliminate most threats to security and offer consistency of test experience. For example, browsers could be limited to the assessment URL address, keyboards could be locked down so that students can not escape the assessment environment and it could be ensured that no applications like spreadsheets and calculators are available. Such limitations cannot readily be applied to school infrastructure as they can require altering the host computers' systems, settings and applications.

Portable application

A portable application is software that runs on a computer's native operating system without being physically installed on the native system. Portable applications can easily be transported to the host school on USB flash drive (or other portable media). Assessment data are usually collected on a USB flash drive (or 'memory stick'). Using a USB drive to both deliver the portable application and collect the resultant data is the most practicable option. PISA 2012 used a portable application[4] to deliver its computer-based assessments in 44 countries to deliver tests in 56 language variants[5] to 145 431 students in 10 303 schools.

The software delivered included:

- TAO data collection and test management architecture
- Mozilla Firefox Portable Edition browser
- Flash Player plugin for the browser
- Apache HTTP server with PHP5
- TrueCrypt encrypted data container
- ClamWin antivirus scanner
- AutoHotkey to standardise and constrain the keyboard settings

The assessment items were written almost entirely in JavaScript but there were some Flash elements.

As with internet delivery, a considerable advantage of a portable application is that it can access features of the host computer's operating system without leaving a footprint on the computer. The application has access to the local font set and input methods, and driver recognition problems are precluded. Students interact with the application in exactly the same way as they normally would interact with any other application on their school's computers.

The interactivity between the portable application and the host operating system brings some disadvantages with it. First and most importantly, the portable application system is not totally secure. It is always possible that students will be able to leave the application and access other applications on the host computer. For example they could go to a spreadsheet and calculate an equation, or go to the internet and search on a topic. They could also conceivably copy and store or transmit the test material. The portable application model relies on vigilance of test administrators to avoid these problems.

[4] Developed through a collaboration between software developers at CRP Henri Tudor, the German Institute for International Educational Research (DIPF) and the Australian Council for Educational Research (ACER).

[5] There were 38 distinct languages, in 56 variants (e.g. French French & Canadian French; British English, US English, Australian English).

Another disadvantage is that portable applications can usually only be configured to operate on a single type of host operating system. For some applications this can be as specific as, say Windows XP; others might run on two or more versions of Windows. Developing an application to run on multiple operating systems such as Windows *and* Mac uses considerably more development resource.

Finally, although a portable application utilises all the advantages of the local operating system's features, it is also subject to that system's constraints. For example, in PISA 2012, the portable application was designed to run on Windows via USB and consequently it was not possible to circumvent the Windows User Management System. Windows imposed the constraint that the software had to be opened with administrative privileges (i.e. a user without administrative privileges was blocked). Where the school's IT infrastructure is managed externally (e.g. by a contracted company or at the school district level) obtaining administrative permissions can be very difficult.

The IEA's International Computer and Information Literacy Study (in 2013) will also use a Windows based portable application. Two physical methods will be used to deliver the application: a USB version delivered to single computers; and a version on a server (laptop) that is connected to the school's Local Area Network (LAN) to deliver to multiple computers simultaneously. Advantages of the LAN based system include that all results from a test session are stored on a single device (the laptop server) and although the test is Windows based, the Windows User Management System is bypassed to the degree that administrative rights are not required at the testee end.

Live system

The term 'live system' denotes an operating system that runs on a local computer without the need to install it on the local drive. The live system can be delivered to the local computer by means of portable data storage media. Like the portable system data are collected on a USB flash drive and so using a USB drive to both deliver the live system and collect the resultant data is the most practicable option.

As with internet delivery, and portable applications, the live system does not leave a footprint on the computer. A minor change to the host computer configuration may be required in order to boot directly from the USB drive but this is a relatively straightforward procedure and usually only a minor inconvenience, if it is required.

A major advantage of the live system model is that it is totally secure in operation mode. Testees cannot operate outside of the provided environment: they cannot access the internet, email, spreadsheets, dictionaries or calculators unless they are specifically provided in the assessment environment.

For the Digital Reading Assessment in PISA 2009, a live system was developed[6] that included:

- Knoppix for Linux Operating System,
- TAO data collection and test management architecture
- Fluxbox X window manager
- Iceweasel browser

[6] By collaboration between software developers at CRP Henri Tudor, the German Institute for International Educational Research (DIPF) and the Australian Council for Educational Research (ACER).

- Flash player plugin for the browser
- Apache HTTP server with PHP5

The assessment items were written in Flash.

Being freeware, this live system bundle offered the greatest flexibility with respect to adaptability and could be used for no cost, but there were certain disadvantages. In particular, the Knoppix operating system did not recognise all hardware drivers, it was necessary to use uncommon text input methods (this is an issue for languages that use Chinese, Korean and Japanese input methods), and there was sub-optimal Flash player support for Linux (causing difficulties in display of Cyrillic fonts and input of right-to-left languages, for example). A universal open-source operating system may not support the variety of technical requirements from the diverse range of language groups involved in a large-scale study, especially when open written responses are required for comprehensive coverage of the assessment framework.

Other initial considerations

This paper has outlined three implementation models and discussed their relative suitability considering various infrastructure constraints and security requirements. The choice of implementation model, while critical, is not the only technical issue for consideration in designing a computer-based assessment. While elaboration of these issues is beyond the scope of this paper, it is worth briefly mentioning some of them, as their early resolution in the design process will be of great benefit to technical development.

No matter what implementation model is used, the architecture of the assessment should be carefully planned from inception (see ITC, 2005). Architecture here refers to fundamental structures of the assessment. In particular, the assessment interface is critical. Navigation and timing structures, accessibility considerations and the incorporation of multiple languages all impact on the design of the assessment interface. Testee registration and tracking are integral administrative considerations.

Assessment interface

The assessment interface refers to the visual and functional elements of the assessment other than the actual assessment items or stimuli.

The assessment interface should be coherent, intuitive and consistent. For example, testees should always find 'help' in the same place throughout the assessment, whether that is from a button, menu or keyboard shortcut. Navigation options, progression through the assessment and time constraints should be communicated in a clear manner. The student should not be distracted from the assessment by an overly complex testing environment.

The overall screen layout is an aspect of the interface that may impact considerably on the design of the assessment stimuli and items. In particular the overall assessment design should include a decision about the positioning of items or task instructions relative to stimuli. Will there be an area in which there is freedom for item writers to arrange stimuli and tasks ad hoc? Or will there be predetermined areas for tasks and stimuli?

Navigation and timing architecture

A major advantage of computer-based assessments is that complex workflows can be implemented, including controlling what the student is faced with, under what conditions, and for how long. Adaptive testing, for example, involves estimating the testee's ability at various intervals during the test (after a set of items, or testlet; or after each item) then presenting the student with an item or set of items that are near the testee's estimated ability estimate.

A useful workflow control used in PISA computer-based assessments was the imposition of a linear assessment flow – that is, testees could only move forward in the test and could not return to tasks with which they had previously engaged. This 'lockstep' approach was used to good effect in the Problem Solving assessment of PISA 2012 to independently measure the different processes involved in solving a problem. For example students were initially assessed for their ability to represent a problem situation by drawing the mental model that they had acquired during an interactive exploration of a system presented to them. In a subsequent task, a correct representation of the mental model was provided and the testee was then directed to transform a given system state into a target state. Thus, representation and formulation of the problem on the one hand, and planning and executing a problem solving strategy on the other, were processes that were separated out by the lockstep approach and measured independently.

Of course other navigational options are available. Navigation between items can be free, in that the student can navigate backward or forward and complete any item in any order. This is similar to what happens in a paper-based assessment. An argument for free navigation is that it allows students to complete the items with which they feel most comfortable, or are most proficient, first.

Whether item navigation is free or constrained in some way is an essential architectural parameter. It is not only important from a programming perspective, it also has implications for the assessment interface. Ideally, the interface will indicate the total number of items in the test, which item the testee is currently viewing, what items have been viewed by the testee, what items the testee can return to, what items have been answered, and what items are no longer available. The more complex the navigational architecture, the more complex the interface becomes.

Similarly, all manner of timing opportunities are afforded by computer-based assessment: a time limit can be imposed at the total test level, at a cluster or testlet level, at the item level, or any combination of the above. The more complex the timing options are, the more complex it becomes to communicate the time status and this can complicate the interface.

Accessibility issues

In considering a computer interface there are both accessibility challenges and opportunities that are not present in a paper-based medium. One challenge is the manual dexterity required to operate a mouse: for example, to accurately pinpoint an object, click, hold, drag and drop. Other accessibility challenges relate to the ability to type and to use character based input methods (for some Asian languages).

Accessibility options that can be facilitated by a computer medium include magnification of test and images (whole screen or partial), voice recognition, and on-screen readers.

Accessibility challenges and options should be considered when choosing or designing the software architecture *and* when designing assessment items, to factor in potential psychometric and administrative impact.

Translation

If multiple languages are to be implemented, the management of a translation workflow is critical. At the very least there should be a method to replace text elements in the source/development language with translated/target equivalents. This holds for both the assessment interface and the assessment items, although different methods may be used for each. With regard to the assessment interface, it is useful to keep on-screen text to a minimum, using symbols, icons and intuitive graphic elements instead. However, it is almost inevitable that some language elements will be required: mouseover text, help pages and error messages for example.

Testee Registration and Tracking

In any complex assessment that involves multiple instruments the need to link the students with their various results datasets is vital. In addition to the computer-based assessment in PISA 2012 each testee was administered one of 13 different paper-based test forms and one of three forms of a questionnaire to gather information about the student's background and attitudes. The three instruments administered to the students were linked with a unique 13 digit student identifier. An additional 5 digit a cyclic redundancy check (CRC) was used in the computer-based component at log in to ensure the student did not mistype the ID which might cause the data to be linked to the wrong students (or to no student at all).

Assignment of students to particular test forms is often important in a survey situation to ensure adequate form rotation and linkage between related instruments (for example, a test form containing mathematics can be linked to a questionnaire on mathematics attitudes, while another form containing reading comprehension can be linked to a questionnaire on reading behaviours). In PISA 2012 there were up to 24 computer-based test forms. Assignment of students to test forms happened a lot later than when the computer-based assessment software had to be produced for each participating country. It was therefore impossible to incorporate a final list of all sampled students and their assigned forms into the computer-based assessment. Instead, at log in, the student typed in their assigned form number followed by a simple checksum, both of which were provided by the test administrator on the day of assessment. If validated, the software assigned the appropriate form to the student.

Summary

Computer-based methods offer significant benefits to the landscape of assessment options.

However, to realise all the benefits from a state-of-the-art computer-based assessment involves technical planning from the onset of assessment design. The choice of implementation model needs careful consideration as it has implications for budget, timeframe and the technical expertise required.

In particular, the type and variety of technical infrastructure present in or available to participating schools impacts on the implementation model. Heterogeneity in hardware and software infrastructure presents challenges to development as does any multiple language requirement. Where there is a wide diversity of infrastructure, an internet delivery or portable application implementation model may be preferred as these use the features of the local environment (school computers).

The level of assessment security required is also an important planning consideration. If relying on school infrastructure, there is relatively more risk to security involved with internet delivery, while less risk is posed with portable applications. The self contained, secure assessment environments provided by live system models pose the least risk to assessment security of the three models. However, security risks associated with internet delivery and portable applications can also be completely overcome by providing pre-configured hardware/software options such as those afforded by carry-in laptops or testing centres.

Other considerations that are vital include those relating to the assessment architecture. Of the multitude of navigational and time constraint features available in a computer-based environment, what ones are to be implemented and how will these be communicated in the interface? What accessibility options are required and which of these are practicable in the chosen implementation environment? How will testee registration be managed? Will multiple assessment languages be required and will this have implications for the assessment interface?

The nature of the content of computer-based assessment items has barely been touched on in this paper. State-of-the-art items can include complex interactive elements, animations, audio, video, and voice recording. Such features may assist measurement and motivate the testee. For technical planning, however, it is important that test and system developers have a mutual understanding of how complex the items are likely to be so that the choice of implementation model and the design of the assessment architecture can be informed by the level and type of technical resources the items will require.

References

Hickey, K. (2010) Dark Cloud: Study finds security risks in virtualization. *Government Security News*. Retrieved from http://gcn.com/articles/2010/03/18/dark-cloud-security.aspx, 29 August 2013.

Jackson, R. (2013) NZ segment of Cloud Security Alliance calls for transparency, security. *Computerworld*. Retrieved from http://www.computerworld.com.au/article/524392/nz_segment_cloud_security_alliance_calls_transparency_security/, 29 August 2013.

ITC (2005) International Test Commission Guidelines on Computer-Based and Internet-Delivered Testing. Retrieved from http://www.intestcom.org/Downloads/ITC%20Guidelines%20on%20Computer%20-%20version%202005%20approved.pdf

Walker, M. (forthcoming) An international comparative perspective on computer-based assessments. In Adams, R, Cresswell, J, Lietz, P and Rust, K: *Implementation of Large-scale Education Assessments*. New York: John Wiley and Co.

Yan Piaw, C. (*2012*) Comparisons between computer-based testing and paper-pencil testing: testing effect, test scores, testing time and testing motivation. *Computers in Human Behavior,* 28(5), 1580-1586.