CHAPTER 4

eAssessment System for TIMSS 2019

Mark Cockle
Heiko Sibberns

Introduction

As described in Chapter 1 of this volume, TIMSS 2019 marked the beginning of the transition to eTIMSS—the digital version of TIMSS designed for computer- and tablet-based administration. eTIMSS offered an engaging, interactive, and visually attractive assessment that enabled TIMSS 2019 to better assess complex areas of the mathematics and science frameworks and increase operational efficiency in translation, assessment delivery, data entry, and scoring. Although the aim is to switch completely to the new digital mode in future assessment cycles, in recognition of the different levels of preparation and infrastructure, countries had the option in 2019 of choosing either eTIMSS or paperTIMSS.

In addition to the overarching requirements for a computer-based system that could produce attractive and engaging assessment items while being reliable, flexible, and easy to use, there were a number of other conditions that had to be taken into account in choosing the system:

- The assessment should be capable of operating on tablets as well as on personal computers.
- Assessment delivery should be via USB memory sticks or through a local server approach whereby the assessment software is installed on a local server that can be accessed by a small number of clients (no more than 30). Full internet-based administration was not a requirement for the TIMSS 2019 assessment cycle.
- Because about half the countries were administering the paperTIMSS version, it was important that items developed in the eAssessment system be as similar as possible to their corresponding paper versions, while capitalizing on interactive computer-based features such as drag-and-drop, multi-select, and drop-down menus.
- Beyond the utilization of features just mentioned in developing individual items, the system should also accommodate more extended Problem Solving and Inquiry Tasks (PSIs) designed to simulate real world or laboratory situations in which students could integrate and apply process skills and content knowledge to solve mathematics problems or conduct virtual

scientific experiments and investigations. These tasks would be tailor-made and not have any counterparts in the paperTIMSS assessment.

- Great emphasis also was placed on clarity and ease of use of the student interface, which was to be kept as simple as possible. This meant, for example, that only those tools such as rulers or calculators that were necessary for processing a specific task were available.
- Since the TIMSS assessment has to be translated and adapted to the needs of each country and language while retaining the same user experience, it was important that the system incorporate a preview functionality for checking that the assessment content appears in exactly the way it is intended for the assessment situation.

To meet all of the design requirements and constraints and to adequately take into account the workflow that has been optimized by TIMSS during the last 20 years, it was decided to develop the eTIMSS computer-based assessment system in-house instead of using an existing commercial system or having it developed by an external company. Accordingly, the eTIMSS "eAssessment system" was designed and implemented by the software team at IEA Hamburg, with input from the TIMSS & PIRLS International Study Center on the user experience/user interface and from IEA Amsterdam on translation issues.

The TIMSS 2019 eAssessment system consisted of a number of integrated software and application modules as follows:

- The **Designer** is an item authoring system used by the TIMSS & PIRLS International Study Center to develop the eTIMSS achievement items
- The **Assembler** was used to group items into item blocks and item blocks into student "item block combinations" (student booklet equivalents)
- The **Translation System** was used by National Research Coordinators (NRCs) from each country and benchmarking participant to translate the items into their language(s) of instruction and by IEA Amsterdam and the TIMSS & PIRLS International Study Center for translation and layout verification, respectively
- The assessment **Player** was used to administer the eTIMSS assessment—present the items on tablet or computer, record students' responses, and upload the data to the IEA servers
- The **Data Monitor** was used by NRCs and test administrators to check the status of uploaded material and progress of the data collection
- The **Scoring System** was used by NRCs and their scoring staff to review students' written responses to constructed-response items and score them according to the eTIMSS scoring guides.

Design and Architecture of the eTIMSS Modules

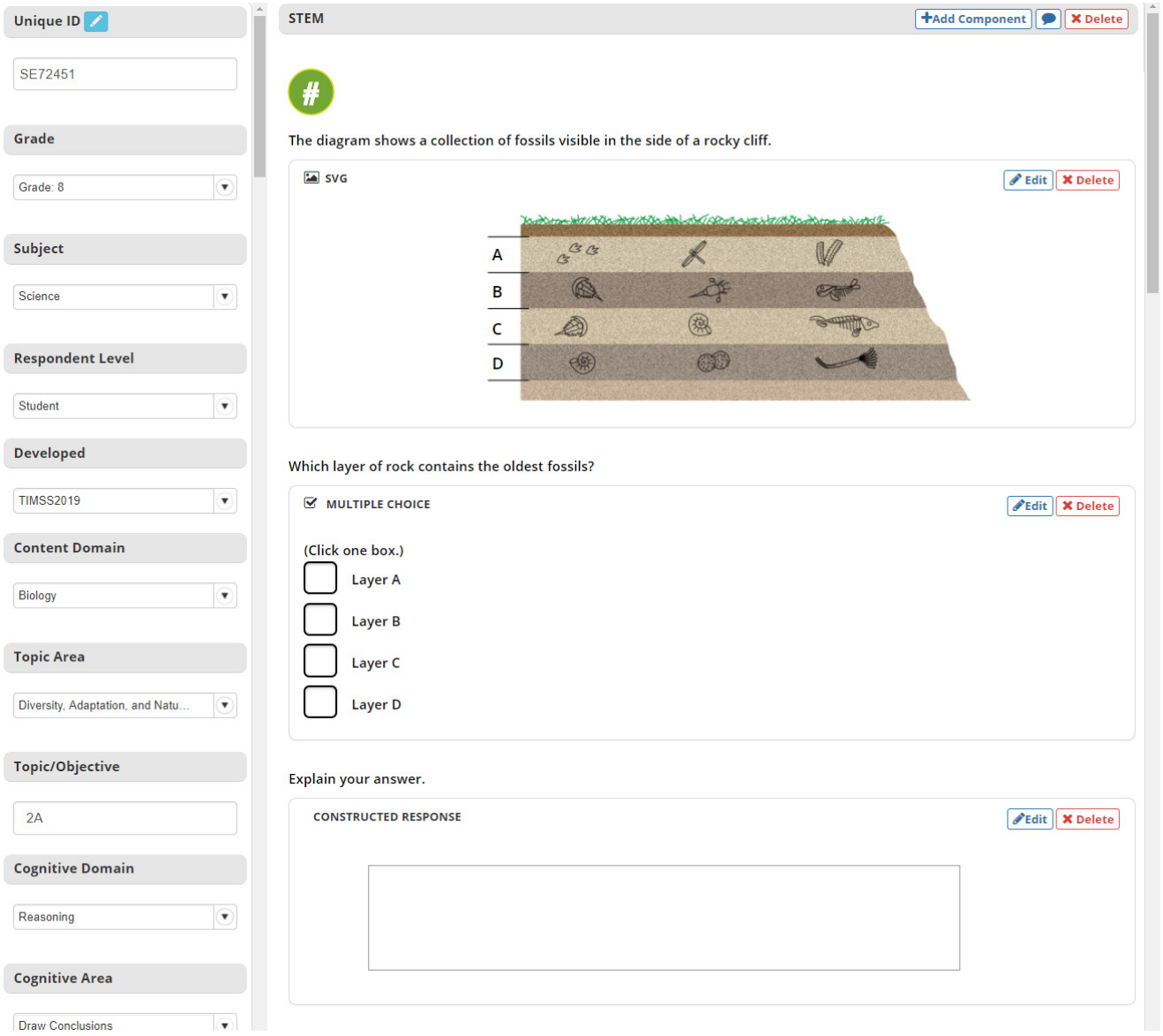
In considering the description of the TIMSS eAssessment system it is helpful to differentiate among three distinct subsystems: 1) the production system for creating assessment content (the Designer, the Assembler, and the Translation System); 2) the delivery system for administering the test in the test session (the assessment Player); and 3) the retrieval and processing system, for upload of the test data to the IEA servers, scoring, and further data processing.

The following is an overview of the various system components and their interaction.

Designer

The designer was used by staff at the TIMSS & PIRLS International Study Center to create the digital versions of the standard (non-PSI) items. Exhibit 4.1 shows part of the screen used for the creation of items. In this example, the item includes three separate elements: a Scalable Vector Graphic (SVG) image, a multiple-choice option, and a constructed response field. On the left is a column for item properties, including, amongst other information, the item ID number, testing grade, testing subject, and content domain assessed by the item. These were used for the selection and identification of the item later in the process of assembly and booklet creation, as well as within the Player.

Exhibit 4.1: eAssessment Designer Input Screen for Example Item



The objective in developing the Designer was to create an item authoring system that encompassed existing TIMSS paper item formats (multiple-choice, constructed response etc.), including the stem text, images and so on, but also new item types unique to the electronic environment. These included drop-down menus, drag & drop, selection (boxes or images), and sorting (boxes or images) item types (see *TIMSS 2019 Item Writing Guidelines*). To accommodate items where the student had to draw shapes or lines, a line-drawing grid also was introduced.

The Designer included various features that could be used by item developers in creating or customizing items, which was particularly important when dealing with trend items where a close match between the electronic item and the paper version was required. For example, the application of labels to images could be made above, below, or to the left or right of an image, or tables could be inserted with invisible lines in order to place objects within columns to obtain a more precise layout.

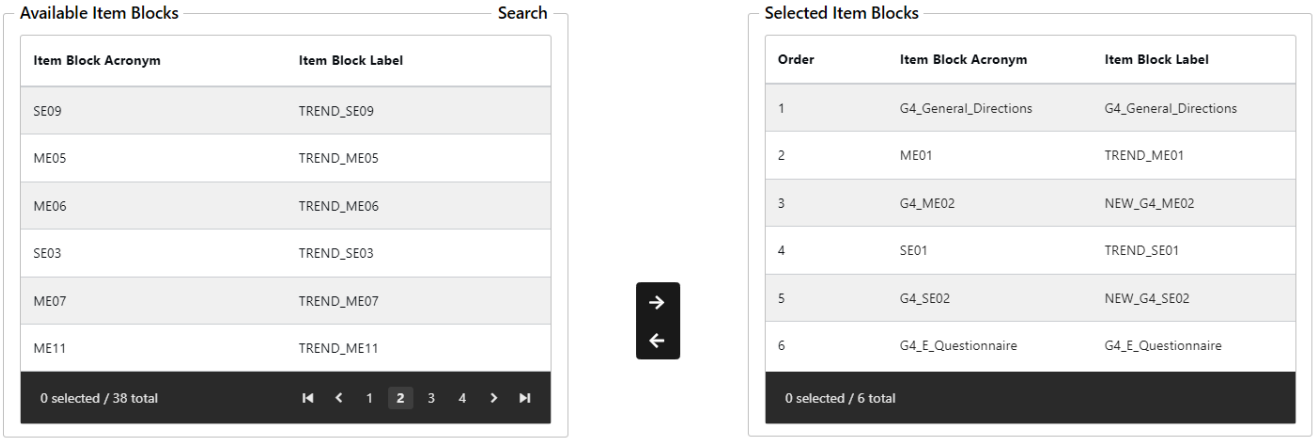
To accommodate items that included images with overlaid text, which are very common in TIMSS, an SVG feature was introduced. This provided great versatility in working with images such as line or bar charts that had overlaid text (e.g., axes labels) that later had to be translated.

It should be noted that the extended Problem Solving and Inquiry Tasks (PSIs) were substantially more complex and interactive than the standard eTIMSS items, and so were constructed independently of the Designer and subsequently combined into item block combinations (or eAssessment "booklets") by the Assembler.

Assembler

The Assembler module was used by IEA Hamburg and the TIMSS & PIRLS International Study Center to combine assessment items into blocks, and then item blocks or PSI tasks into item block combinations along with the assessment directions and eTIMSS questionnaire, in accordance with the TIMSS 2019 matrix-sampling booklet design (see TIMSS 2019 Assessment Design). It was also within this module that the allocation of the booklet number to the instrument was made. Exhibit 4.2 shows part of the Assembler window, and illustrates how a completed booklet has been constructed with six elements; on the left are available blocks that were not included in the construction of this booklet.

Exhibit 4.2: Combining Item Blocks and Directions in an Assembler Window



Once items were assembled into item blocks they could be released country-by-country into the Translation System, where countries and benchmarking participants could begin translation. Item blocks could be "released" for translation one by one as they were completed in the Designer. However, all items in the block had to be complete before the block could be released. If any item in the block was still in the state "In Progress," the release was prevented to ensure that only the approved material would be presented to those using the Translation System. A consequence of this "whole block" approach was that if a minor change had to be made to an item after the block had been released, the entire block had to be withdrawn from the released state until such time as the change had been made and the item could be reassigned to the block.

Translation System

The Translation System was a critical part of the eAssessment system and was used by multiple parties. These included IEA Amsterdam and IEA Hamburg during the setup, NRCs and translators from the participating countries, translation verifiers employed by IEA Amsterdam, and layout verifiers from the TIMSS & PIRLS International Study Center. Although the primary purpose of the system was to enable translation, verification, and documentation of any deviations from the original international English source version, the system also enforced the appropriate workflow by a combination of user rights applicable at different stages of the process leading to the final, approved, translations.

Exhibit 4.3 shows an example of the translation window, where the eTIMSS Questionnaire title has been translated into German. Note that three fields are displayed—the original English source text in the top field, the current translation in the bottom field (only this field is directly editable), and a "track changes" field to show the changes in a color-coded manner. Additional fields could be shown, for example, to compare a current translation with an earlier version submitted for translation verification.

Exhibit 4.3: Example Translation from the Translation System

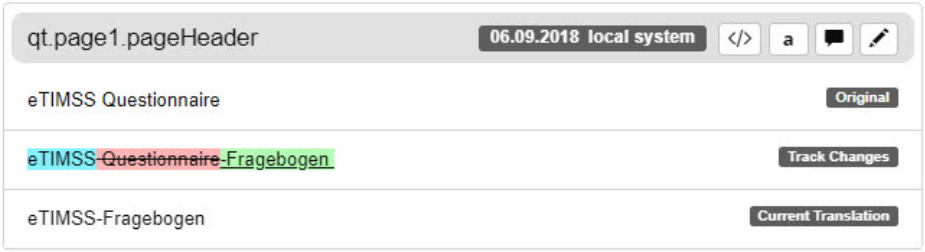
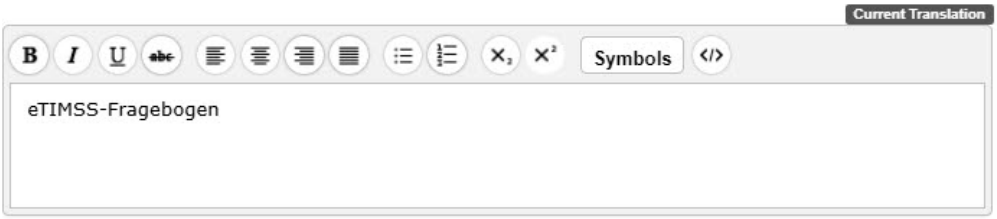


Exhibit 4.4 shows the translation editor, which enabled making a range of layout and font changes as well as inserting HTML commands, symbols, or mathematical structures such as fractions.

Exhibit 4.4: Translation System Editor



Translators also had the option to export the international source text elements in an XLIFF format, which could be used in standard translating programs for increased efficiency. This required the translations to be imported back into the Translation System and formatted for translation verification and layout verification.

The guiding principle in designing the Translation System was to define a process similar to that used in translating and verifying paperTIMSS assessment instruments. The process began by releasing a copy of the international English version of the achievement items into a separate language-specific folder for each country, followed by a workflow consisting of a series of status indicators indicating the progress of the translation and verification that were set by the various parties involved in the process. These status indicators were defined by IEA Hamburg and the TIMSS & PIRLS International Study Center as follows:

- **In Translation & Adaptation**: The initial, default status following release to the country
- **Ready for Translation Verification**: After completing translations, all blocks had to be set to this status when the translated materials were ready for translation verification
- **In Translation Verification**: Set by IEA Amsterdam when translation verification began, which locked the system for editing during the process
- **In NRC Translation Approval**: On completion of translation verification, the system was unlocked to allow the NRC to apply edits based on feedback from the translation verifier
- **Ready for Layout Verification**: After translation verification was complete and all edits applied, all blocks were set to this status to submit materials for layout verification
- **In Layout Verification**: Set by the TIMSS & PIRLS International Study Center when layout verification began, which locked the system for editing during the process
- **In NRC Layout Approval**: On completion of layout verification, the system was unlocked to allow further editing by the NRC based on feedback from the TIMSS & PIRLS International Study Center

- **Instrument Finalized**: The final status, assigned by the NRC, indicated that the materials had completed all verification steps and were ready for assessment Player production.

At each step along the way, comments could be left to document the process. The idea was to mimic the workflow and fields used in the National Adaptation Forms for the paper version. In the particular case of translation verification, additional labels were available to indicate specific errors or deviations found/corrected, including a "severity code" assigned by verifiers to each deviation to assist the NRC in deciding whether to accept or reject suggestions made by the verifier (see Instrument Translation and Layout Verification). In general, comments were labeled so as to indicate for whom they were intended. For example, a layout verifier could leave comments in the system for the attention of the NRC, in which case the label would have been "Layout."

A preview feature was available for all users to display items exactly as they would appear within the final Player. This was especially useful during layout verification, allowing as it did comparison with a preview of the original (untranslated) source version.

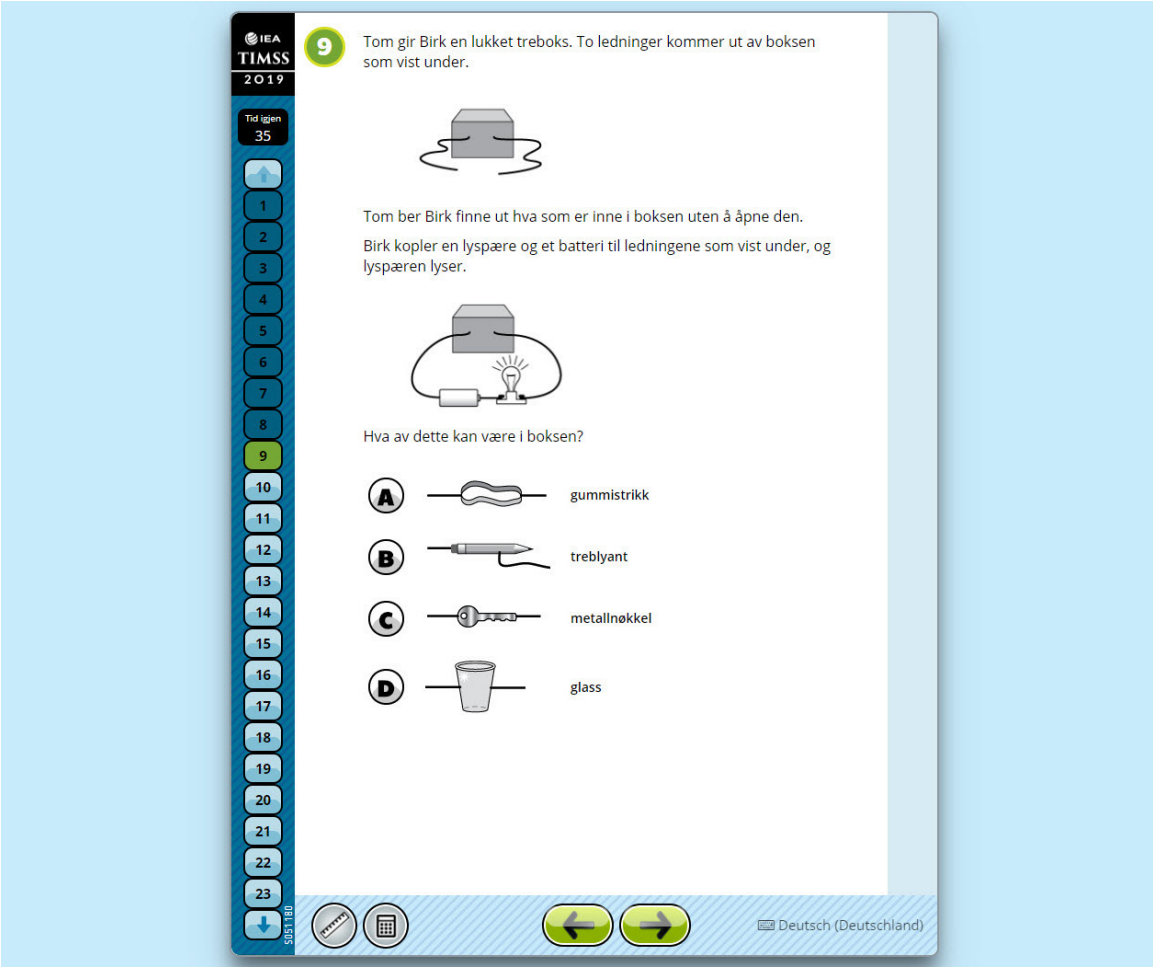
Player

The assessment Player is the software that the student interacts with while taking the eTIMSS assessment. The Player presents the assessment items to the student and uploads the student response data to IEA's data servers. After translation and layout verification were successfully completed, a customized version of the Player was produced for each language of instruction in each country. This sometimes required last-minute adjustments to the layout by IEA Hamburg before supplying the Player to the countries.

To access the Player, the student or the test administrator entered the login credentials assigned to that student. These consisted of a unique ID number and password which incorporated a two-digit code that determined the specific assessment item block combination assigned to each student. After entering the correct login credentials, the test administrator read aloud a test administration script that instructed students to enter a four-digit code to begin the test directions introducing students to the various types of items. After working through the directions, students were instructed to enter another four-digit code to begin working on the first part of the assessment. Following a short break, a third four-digit code provided access to the second part of the assessment. Finally, a fourth four-digit code allowed access to the short eTIMSS questionnaire.

Exhibit 4.5 shows the Player user interface, with an example science item. The students navigated through the assessment using the green forward/backward arrow buttons or via the navigation bar on the left side of the screen. The navigation bar records the students' progress through the assessment, showing which items have been completed and which have been omitted or not yet attempted. There also is a timer showing the remaining time.

Exhibit 4.5: eTIMSS Assessment Player User Interface



The software development criteria for the Player encompassed multiple, sometimes competing elements. The key elements were speed of operation, security, and consistency of user experience.

Speed of Operation

Several design factors were involved in producing a Player that responded smoothly and produced the minimum delay when navigating between items:

- The content (directions and items) to be displayed to the student was preloaded directly after the login screen, so that only response storage processes took place during the test session
- The Player database was mirrored in memory for faster performance
- The format of the data saved was as parsimonious as possible to maximize performance when saving and uploading data.

Security

To ensure the security of the student data and test items:

- The Player was configured to run within a ”sandbox”—a virtual space in which software can be run securely–to isolate the Player in a restricted memory range
- Contents of the sandbox were automatically deleted after the testing session.

Consistency of User Experience

To ensure the Player operated the same way for all countries and languages:

- The Player exhibited a close to identical display on Firefox or Chrome browsers or with either the Android or USB Players
- Country- and language-specific CSS files were available to make final layout and font adjustments
- Right-to-left languages had automatically reversed layout, with the ability to revert individual elements back to left-to-right format.

The Player software consisted of an executable file and two or three additional files: 1) a country-specific “Player Model” SQLite database containing the translations and the item block combination structure; 2) a template SQLite database file as the basis of the results database; and, optionally, 3) a CSS file in case layout or font changes were required.

The results database was created for each student at login time, and included information about the particular culture (country/language combination), as well as the student ID and a reference to the country in the name of the database file itself. It should be noted that the results database is the repository of not only the students’ responses to the items but also the timestamped events that reflect the process of working on the assessment, such as navigating between screens, using interface tools, and changing responses to items.

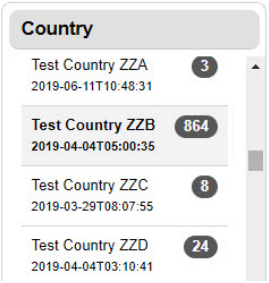
To upload data from the Player to the IEA servers, a menu option in the Player opened a separate upload page. A list of all the results databases in the default location (the same folder level as the Player executable) was displayed, along with buttons to “Upload Data” and ”Refresh.” For those using a Player to upload data from multiple USBs, it was possible to add additional results databases to the list. Clicking “Upload Data” triggered the upload process to start, and a color-coded bar showed the number of successful and unsuccessful uploads.

On completion of an upload, an acknowledgement was sent back to the Player client performing the upload confirming a successful (or rarely, unsuccessful) data transfer. Databases successfully uploaded were moved to an “uploads” subfolder and flagged to ensure that they would not be uploaded again. Databases not successfully uploaded remained in the list for a further attempt to be made.

Data Monitor

The Data Monitor was provided to enable NRCs and test administrators to further check the status of uploaded data and to monitor overall progress during the data collection. As shown in Exhibit 4.6, the Data Monitor enabled all records for a specific grade and country to be viewed, including information regarding the student ID, the record creation time (the time the student logged into eTIMSS Player), and the time of uploading.

Exhibit 4.6: Example Information from the Data Monitor



Culture	IDSCHOOL	IDCLASS	IDSTUD	Module	File Size	Record Created	Record Uploaded
en-ZZB	9998	999820	99982014	25	53 kB	3/27/2019 5:40:27 PM	3/27/2019 5:46:32 PM
en-ZZB	9998	999820	99982011	22	64 kB	3/27/2019 5:35:55 PM	3/27/2019 5:46:33 PM
en-ZZB	9998	999820	99982006	17	70 kB	3/27/2019 5:30:51 PM	3/27/2019 5:46:32 PM
en-ZZB	9998	999820	99982005	16	72 kB	3/27/2019 5:27:26 PM	3/27/2019 5:46:33 PM
en-ZZB	9998	999810	99981001	12	43 kB	3/27/2019 12:57:53 PM	3/27/2019 5:09:04 PM
en-ZZB	9998	999810	99981001	12	43 kB	3/27/2019 12:57:53 PM	3/27/2019 5:09:03 PM
en-ZZB	9998	999810	99981001	12	43 kB	3/27/2019 12:57:53 PM	3/27/2019 5:09:04 PM

Scoring System

The IEA CodingExpert software, consisting of an Administration Module and a CodingExpert Client, was the online scoring system used by NRCs and their scoring staff to score the eTIMSS constructed response items. The Administration Module enabled scoring administrators from each country and benchmarking participant to activate scorer accounts, assign scorers to items, set up and distribute training materials, distribute student answers, and monitor the progress and quality of the scoring. Scorers used the CodingExpert Client to score the student item responses assigned to them by the scoring administrator.

The Scoring System was an independent online system, working in tandem with local client software that supplied the students' responses to the scorers along with contextual information such as the translated item stem. In addition to the standard constructed response questions familiar from paper scoring, the eTIMSS Scoring System had to accommodate responses from unique, digitally-enhanced item types in the Problem Solving and Inquiry Tasks, as well as display screenshot images from the line-drawing items.

Preparing Data for Scoring and Processing

Some pre-processing steps were required to prepare data in a suitable format for import into the Scoring System and to enhance the efficiency of the human-scoring process. Data uploads from the eTIMSS Players were processed at IEA Hamburg by several data servers that received and then extracted the raw data from the uploaded SQLite databases into the "central" SQL database for all countries. This new structure contained a separate database for each country and grade, including all data from the original

SQLite databases with the addition of identifiers relating to the import of data and additional fields for scoring purposes.

Although scoring supervisors controlled the distribution of responses to scorers within countries, the responses themselves became available in the system soon after upload (with some delay due to the asynchronous handling of the import to the central database and thence to the scoring system). To avoid unnecessary scoring, therefore, it was essential that any duplicates in the central database were dealt with before import to the scoring system. In addition to measures to prevent a database from being uploaded a second time from the client side, checks were made to the results database creation date and content to ensure any possible duplicates were flagged before import. There were, however, some kinds of duplicate records that could be legitimate. Two databases with the same student ID but with different creation times could have originated in several scenarios. For example, this could be simply a case of the test administrator mistakenly using the same ID twice for two different students, or an interruption in the assessment may have led to part 1 being conducted from one USB stick and part 2 from a second. Such cases needed to be reconciled by IEA Hamburg's data processing procedures.

When scoring was completed, the student response data were transferred to tables prepared for import into the data processing system (DPE) employed at IEA Hamburg for all large-scale international assessments. Here data from the various other TIMSS sources, such as the student questionnaire or online context questionnaires, were merged together, using the IDs from the WinW3S database as the key. Following an intensive series of quality control checks to identify and reconcile any inconsistencies, the data were exported to SPSS and SAS data files for distribution to countries as part of the International Database (see Chapter 8: Creating the TIMSS 2019 International Database).

eTIMSS Assessment Delivery Methods

Countries participating in eTIMSS could choose from the following three methods for delivering the assessment:

- **USB delivery** involved running an executable file from a USB flash drive preloaded with the eTIMSS Player
- **Tablet delivery** involved running the assessment Player directly from an application on an Android-based tablet
- **Local server method** involved the use of the same Player as for the USB delivery, but run from a server on a local area network, with the client computers running a browser to connect to the server.

In practice, most countries focused on one standard method of delivery, with other options only used as exceptions due to special circumstances within schools.

USB Delivery

USB delivery involved running an executable file from a USB flash drive preloaded with the eTIMSS Player. Test administrators were instructed to run the executable, which would open the program on a main menu. On clicking the menu "Start eTIMSS" the program would present a login screen in "kiosk" full screen mode—a semi-locked-down state where some key strokes are blocked and students are unable to access or see the browser address bar.

The suitability of computers for this mode of delivery was determined by running a "system check" program, which returned a clear yes/no indication on parameters based on screen resolution, operating system, CPU speed and available memory, as well as a USB transfer rate check. This system check was provided as a stand-alone program for checking computer compatibility ahead of administration, but was also a module of the USB eTIMSS Player itself, for use on the day of testing.

Following the test session, the test administrator could use an escape code to return to the main menu in order to upload the results. It was recommended to perform the upload as soon as possible following the assessment, but it was also possible to conduct consecutive test sessions for several students and then upload these together at once. Further, it was possible to copy the results databases from several student USB flash drives to one single drive and use that to perform the upload function.

Tablet Delivery

Tablet delivery involved running the assessment Player directly from an application on an Android-based tablet. This application needed to be first installed on the tablet from an .apk (Android Application Package) file. Once installed, the application was available from the tablet home screen. The icons for these were labeled in such a way that fourth grade and eighth grade versions could be distinguished.

On tapping the appropriate application icon, a login screen would appear in full screen. In contrast to the USB version, it was not possible to prevent operation of the home button due to restrictions of the Android operating system.

A system check for tablets was provided via the Google Play Store as a separate application; there was no system check within the standard application. Minimum requirements comprised screen resolution (identical to the minimum for the USB application), version of operating system (Android 5.0.2 or higher), available storage, CPU speed, and available memory.

Following the test session, the application would return to the login screen. A button at the bottom of the screen allowed the upload of the assessment data to take place given the entry of a four-digit password. If multiple students had taken the eTIMSS assessment since the last upload, data for all these students would be transmitted.

Local Server Method

The local server method was a feature of the USB Player. With the Player program stored on a local drive of a PC, it was possible to start the program as with the standard USB from the main menu. An option then enabled the test administrator to set up the PC as a server and enable computers connected to the local network to connect to this, displaying the assessments in a browser (the Chrome browser was the preferred option).

The minimum specifications for the server computer were above those for standard USB delivery, and no system check was available to test suitability. Therefore the following minimum requirements for the server PCs were defined in order to determine if a machine was able to run the Player successfully:

- OS: Windows 8 or higher
- Processor speed: 2.2 GHz
- Memory: 8GB
- Available storage space: 10GB on SSD drive
- Administrator rights.

The upload procedure was similar to the USB method, with the additional step of stopping the server-client service. Once this was done, the results from all students could be uploaded at once.

Description of eAssessment Data

The assessment Player recorded student item responses as well as other actions taken by the student and the data were stored in a SQLite database. Student actions were broken down into timestamped events that recorded process data such as navigation behavior and tool use, but also messages to the student that were created by the system (e.g. time remaining towards the end of the test). The student responses and event data were stored separately, with the item responses in a "response table" and the events in an "event table." There were also auxiliary tables containing the student ID together with the language in which the assessment was administered and information about whether the data had already been uploaded to the IEA server. Other tables were used for error handling.

Each item response or event was stored with both general attributes and attributes specific to that response or event. The following general attributes were recorded:

- Two timestamp parts: The first recorded events and item responses in Unix time and gave the elapsed time in seconds since January 1, 1970. Since a more precise time information was needed for event data, the second timestamp added the milliseconds.
- A sequential number recording the correct sequence of actions: This number reflected the exact order of events and responses and had to coincide with the sequence obtained using the timestamp information.

- A screen ID number: This number indicated the specific screen (or item) on which the response was saved or event occurred.
- A page identifier: Due to the rotation of item blocks within booklets, an item could be displayed in different positions in the assessment. Therefore it was necessary to also include a "page number" as a general attribute.
- An item ID number: For recording responses, the item identifier referred to the particular item or item input (e.g., keyboard field) on the screen. This number corresponded to a given "raw variable name" specified by the TIMSS & PIRLS International Study Center.
- An event-type ID number: For recording events, using ID numbers instead of names helped to minimize data traffic during the assessment administration. A separate reference look-up table held the actual event names that corresponded to the event-type ID numbers.
- A response ID number: For recording responses, this identifier indicated if a response was changed later during the response process. It showed the sequential number (ID) under which the subsequent answer was saved. The final answer the student gave to an item was marked with a "NULL" value for this field.

Item Responses

In the response table, each response was stored in a separate record. The response table held the entire response history of each item the student worked on. All item responses were stored as one or more records with string of characters indicating the student response. This could be a single number, but also an extended string containing information about drawn lines or the dragging and dropping of objects. In addition, the student response table contained typed student responses that were later transferred to the Scoring System for human scoring, along with screenshot images of responses from the line-drawing tool. Responses that did not need human scoring were machine scored. For these responses, a set of detailed scoring rules provided by the TIMSS & PIRLS International Study Center were incorporated in a scoring algorithm and applied to each response to determine the appropriate score.

Event-Specific Attributes

In addition to the general attributes, attributes specific to each event were stored as JSON objects. JSON objects in general hold for each attribute the name of the attribute (property) and the value of the property. Exhibit 4.7 shows an example extract from the event table for the "UI:IsLoaded" event type. This event indicates that the appropriate test form was loaded with the first item presented to the student. The event-specific attribute is the "index" which is set to zero for the first page of the test, stores as the JSON object {"index":0}.

Exhibit 4.7: Extract from the Event Table for Event Type "UI:IsLoaded"

Event-Type Id	Screen ID	Page Identifier	Information
26	13617	0	{"index":0}

Results, Challenges, and Lessons Learned

In retrospect, it was the right decision to set up the eTIMSS system modularly and to differentiate between the phases of content and item creation, translation, instrument assembly, assessment delivery, monitoring of the data retrieval, and scoring of the responses. In each phase, different roles with the corresponding rights were required. Administration was comparatively easy due to the modular structure.

For the translation and translation verification, it turned out to be very helpful that the eTIMSS system supported the XLIFF format. With the help of XLIFF exports, translators could easily import the texts to be translated into standard translation programs and thus carry out the translations very efficiently.

The preview function, which made it possible to display the translated content as it is displayed in the specific assessment situation, was of great help. In this way, it was possible to react very early if the space allotted for the translation was not sufficient and translations were not displayed at all or incorrectly. In these situations, often manual intervention was necessary through CSS files.

In particular, the right to left (RtL) languages (Arabic, Hebrew) presented multiple challenges. A lot of effort went into producing a standard RtL template that could be applied on request in the Translation System. This template had for example certain images flipped or moved to fit to the style. Despite this template, a lot of manual work needed to be done at IEA Hamburg for adjustments of texts, images and input boxes by way of CSS files to finalize players.

The large number of Players that had to be produced in a very short possible time posed a particular challenge. In total, more than 100 player variants were created, all of which had to be tested before distribution. This work was all managed conforming to the timelines for producing paperTIMSS assessment materials.