The International Journal of E-Learning and Educational Technologies in the Digital Media (IJEETDM) 5(2): 64-72
The Society of Digital Information and Wireless Communications (SDIWC), 2019 ISSN: 2410-0439 (Online)

Decomposition, Adaptation and Assembly of Reusable Learning Objects with Multi-format Assets for Higher Education Instructional Design

J.E. Agaba[1], A. Nakakawa[2] and J. T. Lubega[3]

[1]jagaba@cis.mak.ac.ug

School of Computing and Informatics Technology, Makerere University

[2]anakakawa@cit.ac.ug

School of Computing and Informatics Technology, Makerere University

[3]jlubega@utamu.ac.ug

Uganda Technology and Management University

ABSTRACT

Instructional design is one of the key activities in an e-learning environment. There are various approaches used in instructional design and one of them is the E-Instructional design where content is considered to be a set of reusable learning objects. Quite a number of online repositories are stocked with reusable learning objects and usually each repository has a particular learning object metadata standard that it follows. Such learning objects can be retrieved, adapted and reused in instructional design. However, there still remains a challenge in adapting learning objects with multi-format assets. The available tools for adapting learning objects assume that all assets of the learning objects are of one data type yet it is not always the case. In this paper, we present a three-step approach of adapting a reusable learning object with multi-format assets. We evaluate this approach using a prototype of a learning object adaptor tool. The results of evaluation show that this three-step approach improves reusability of a learning object that has multi-format assets.

KEYWORDS

Learning Object Decomposition, Learning Object Adaptation, Learning Object Asset Assembly, Higher Education Instructional Design, E-learning Environments.

1 INTRODUCTION

1.1 Instructional Design in E-learning Environments

Berger and Kam[1] define instructional design process as the *"systematic development of instructional specifications using learning and instructional theory to ensure the quality of instruction"*. Usually, instructional design process varies from one higher education institution to another, and usually depends on a number of factors such as training requirements, client expectations, tools, development time, flexibility, complexity and budget[2]. Available literature shows that there exists a wide range of instructional design models applied in e-learning environments but most of such models are variations of the traditional Analysis-Design-Develop-Implement-Evaluate (ADDIE) model [3]. For example, Ryder[4] categorizes hundreds of instructional design models proposed by various scholars into behaviorist, cognitivist and prescriptive models. Similarly, Schneider[5] presents a sizable discussion on instructional design models. Whereas almost all the instructional design models presented by Ryder[4] and Schneider[5] can be used in e-learning environments (depending on the technological support needed to implement them), this paper focused on instructional design models that present instructional content as a set of items (or chunks) that have a pedagogical sense. An example of such instructional design models is the E-Instructional model suggested by Schneider[5].This instructional design model considers instructional content as a collection of reusable learning objects that are organized following existing sound behaviorist or

cognitivist design principles like Robert Gagne's nine steps of instruction[6].

Thus, in this paper, we focus on higher education instructional design using reusable learning objects and we aim at exploring how the reusability of such learning objects can be improved through adaptation.

1.2 Reusable Learning Objects Defined

The term learning object was first coined by Wayne Hoggins in 1994 [7]. However, through the years, a number of scholars have defined this term in various ways, basing on the context it is applied in instructional design [8]. In this paper, we define a Reusable Learning Object (RLO) as *a unit of electronic instructional content that can be used and reused to achieve a learning objective in a lesson, a module or the whole course*. In addition, we further narrow the scope to RLOs that have the following characteristics;

i. RLOs that are white-box transparent [9].
ii. RLOs whose assets are of different data types such as text, images, video/audio clips, code snippets, among others.
iii. RLOs that can be delivered in a web-based e-learning environment.
iv. RLOs that are decomposable into its assets to allow asset-level adaptation.
v. RLOs whose metadata structure is meant for higher education e-learning environments [8].

1.3 Structure and Granularity of RLOs

A standard RLO from a given repository usually has two parts, namely; the metadata (the definition of the attributes of the RLO defined by a given metadata standard) and the RLO content (the actual instructional content to be delivered to achieve a learning objective) [10].

The granularity of a RLO varies from one metadata standard to another. Battou *et al* [11] define granularity of a RLO as *"a process that involves breaking down educational content into a set of items or blocks having a pedagogical sense, also called grain"*. This definition implies that it does not matter how granular a RLO may be, provided it has a pedagogical sense. A RLO may contain assets (*which are indivisible units of instructional content such as text, images, video/audio clips, applet/servlets, code snippets, among others*). In some metadata standards, assets are also called Reusable Information Objects (RIOs). In Table 1 below, we present a summary of suggestions from different RLO metadata standards about the granularity of a RLO.

Table 1: Suggestions on RLO Granularity from metadata standards

Learning Object Design Standard	**Suggestions on Learning Object Granularity**
SCORM Content Model[12]	No particular size recommended. But it states that assets are joined to form a sharable content object which is the lowest level of granularity of a single learning resource and states that both assets and sharable objects may be reused.
IEEE's LTSC Content Model[13]	No standard size stated but provides 4 levels of learning object aggregation. Level 1(assets), Level 2(lesson or collection of assets), Level 3(course or collection of lessons) and Level 4 (a set of courses leading to a certificate).
NETg Content Model[14]	Provides a hierarchy of four levels of aggregation, as; a topic, a lesson, a unit and a course. The topic is regarded as the RLO with a single learning objective, learning activity and a corresponding assessment.
Cisco Systems RLO Content Model [11]	Attempts to dene the standard size of a reusable learning object as a combination of five to nine RIOs. But the actual size of the RIO is not stated, meaning that the size of the RLO depends on the size of the RIO.

1.4 RLO Adaptation Challenges with Current Tools

Agaba and Lubega [8] define RLO adaptation as *"the process of modifying an existing learning object in instructional design with the aim of achieving a new learning objective in the learning environment."* A number of tools and techniques for adapting RLOs for reusing in instructional design have been developed but the tools are based on the assumption that a RLO is made up of assets of the same data type, which is not always the case. In Table 2 below, we present a summary of the challenges with some of the existing RLO adaptation tools.

Table 2: An analysis of challenges with existing RLO adaptation tools

Learning	**Challenges with Technique /Tool**

Object Adaptation Technique/Tool	
The Wiki-type Content Editor [15]	1. Requires a technical instructional designer to edit HTML/CSS codes in the RLO. 2. Only edits HTML/CSS codes yet a RLO may contain other assets like Java applets which are linked to that RLO through HTML tags. In other words, it does not state how a RLO with multi-format assets can be adapted. 3. Works on the assumption that all the RLOs are always in HTML/CSS. 4. Does not support integration of RLOs with other objects after adaptation. 5. The RLOs supported are not meant for a specific educational level.
Transformation Augmentation and Substitution (TAS) service in LOTTI [16]	1. Does not state how a RLO with multi-format assets can be adapted. 2. The RLOs supported are not meant for a specific educational level.
The GiSHEO eLearning Environment[17]	1. The HTML/CSS editor requires a technical instructor to adapt the content. 2. Also assumes that all RLOs are written in HTML/CSS. 3. The editor in the tool is meant for HTML/CSS text adaptation only yet RLOs may contain other assets (like images and Java applets) where the learning may be centered but won't be adapted. 4. The RLOs supported are not meant for a specific educational level.
GLOMaker[18]	1. The RLOs supported are not meant for a specific educational level. 2. Can only allow adaptation of RLOs authored using this tool which reduces on RLO reusability.

This paper is based on the belief that the reusability of a learning object with multi-format assets can be improved if adaptation is done at asset level.

To achieve asset-level adaptation, the RLO must be decomposed into individual assets so that each asset gets a particular adaptor that matches its data type. The adapted assets are then assembled into a new RLO, ready to be used to achieve a new learning objective.

The rest of this paper is organized as follows; section 2 presents a three-step process of RLO decomposition, RLO asset adaptation and assembly of adapted RLO assets to increase reusability. Section 3 presents a methodology for evaluating this 3-step process, and section 4 presents results of evaluating the proposed three-step process and in section 5, we present a conclusion and future work.

2 RLO DECOMPOSITION, ASSET ADAPTATION AND ASSET ASSEMBLY

2.1 RLO Decomposition

In this first step, we assume that the instructional designer has a RLO that meets the characteristics earlier mentioned in section 1.1 and also knows the instructional needs of all the stakeholders in the instructional design process(that is, the prospective learners, the prospective instructors of the designed content, the institution where the course will be taught, the industry where the graduate of the course will work, and any government educational policies that must be considered in instructional design).

Task I: Determine Granularity Level

This is necessary to determine how large or small the assets will be. It is always important that the assets be as small as possible provided they remain with a pedagogical sense. In other words, the smaller the asset the easier it will be for the instructional designer to adapt it. In this paper, the granularity of the RLO assets is set to be at the first level of RLO metadata structure for higher education presented in Figure I below.

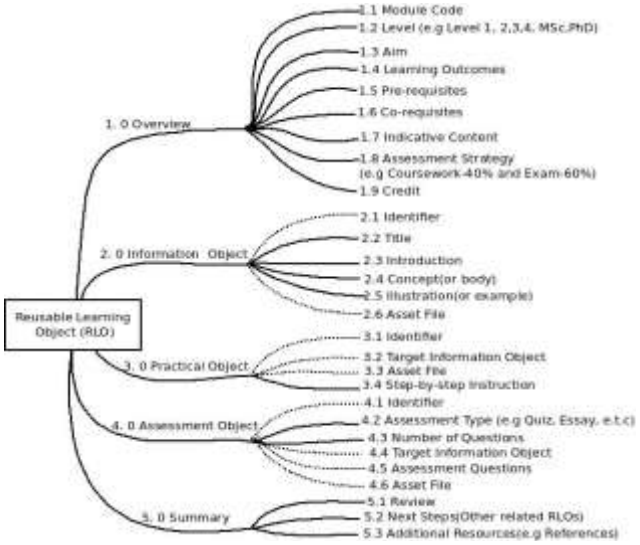

Figure 1: A Schematic Representation of the Metadata Structure for a Higher Education RLO (Adopted from Agaba and Lubega [8], page 70).

Task II: Decompose the RLO as per Granularity Requirements

As per the granularity adopted in Task I above, the Overview, Reusable Information Objects (RIOs), Practical Objects (POs), Assessment Objects (AOs) and Summary are the assets of the RLO. Each of these assets has metadata elements as presented in the schematic diagram in Figure 1 above.

If any of the assets of the RLO (that is, the Overview, RIOs, POs, AOs and Summary) has contents of different data types, then we further decompose the asset into sub-assets. These sub-assets will also have the same metadata elements as the 'mother' asset but for purposes of easy storage and retrieval, their identifiers could be marked in such a way that their origin (mother asset) can be identified. For example, if a RIO with identifier RIO_1 has two sub-assets in form of text and a Java Applet, the text can be identified with $RIO_{1\text{-}1}$ and the Java Applet can be identified with $RIO_{1\text{-}2}$. This makes it easy to have assets easily categorized into their different data types for adaptation. The decomposition process can continue as far as possible provided the assets remain with a pedagogical sense. The challenge that still remains in RLOs' research is the possibility of varying sizes of the assets when the RLO is decomposed. The existing standards on RLOs metadata are silent on this matter (see Table 1 above) neither does this research address it.

Task III: Categorize RLO Assets by Asset Type

Learning object assets can be categorized based on asset data types. For example, Wagner [19] suggests that RLO assets can be text, images, audio/video, code snippets and applets. Also, Raghuveer and Tripathy [20] suggest six categories of RLO assets as text, image, audio/video, animation, explanation and problem/solution. However, as earlier stated in the section 1.1, this research is limited to RLOs that are white-box transparent and which can be delivered in a web-based environment. This limitation implies that this research can only consider assets of type text, code snippets, applets and images (only those whose content and caption/title can be edited).

Therefore, under this task III, the we look at the data type of each asset and group assets of the same data type under one category. For example, if Overview and Summary of the RLO are all of type text, then we group them as text. If a given number of Practical Objects are Java applets, then group them as Java Applets. This categorization of assets makes it easier to determine the adaptors for each asset type during asset adaptation. Figure 2 below presents a flow chart to summarize the tasks under RLO decomposition.

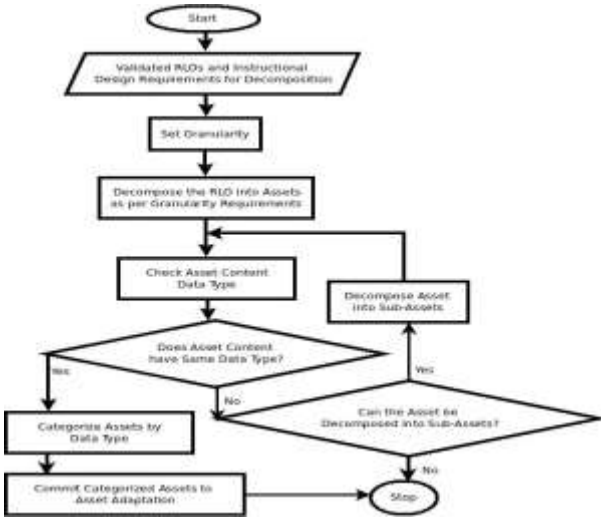

Figure 2: Sequence of Activities in RLO Decomposition

2.2 RLO Asset Adaptation

In this paper, we consider a RLO asset adaptor to be a tool, technique, software application or a development environment that can be applied (or used) by the instructional designer on a RLO asset (following instructional design strategies derived from the instructional design requirements) to enable the asset contribute to the RLO in achieving a new learning objective.

Generally, in RLO asset adaptation, given that A is a set of RLO assets obtained after decomposition of the RLO, you seek for a set T of asset adaptors to apply on set A to produce A' which is a set of adapted assets to be assembled/packaged into an adapted RLO. In other words, given that;

$A = \{a_1, a_2, \ldots, a_k\}$ and $T = \{t_1, t_2, t_3, \ldots t_k\}$,
Then, $A' = \{a_1', a_2', \ldots a_k'\}$,

Where t_1 is an adaptor for all RLO assets that have the same format as a_1, t_2 is an adaptor for all assets that have the same format as a_2, and so on.

Task I: Assign each asset type an asset adaptor

Each asset data type category (from RLO decomposition) is assigned an asset adaptor.
For example, all text-type assets of a web-based RLO would be allocated a Hypertext Markup Language (HTML) editor or any web authoring tool (to edit the text in the RLO asset. Another example would be that all applet-type assets would be allocated a Java development environment (to use in editing of source code, compilation and interpretation of the byte code for the adapted applets).

Task II: Adapt the RLO Asset

By following proper instructional design strategies derived from the instructional design requirements, the instructional designer uses the asset adaptor for the asset type identified in Task I above to make the necessary changes on the asset. The likely challenge in this step could be if the instructional designer does not have sufficient practical skills to do the adaptation. For example, the instructional designer might not achieve his/her goals for adaptation if he/she lacks the required programming skills to adapt an asset through changes in source code. Below we present an example of a set of activities that the instructional designer would be engaged in during asset adaptation.

Consider a RLO retrieved from a repository of instructional materials of a C programming language course for a first year Bachelor of Computer Science university program. The researcher assumes that this RLO aims at enabling the student to understand arrays in C programming language. The researcher assumes that one of the RIOs (the asset) of this RLO is about sorting array elements in ascending order. The asset type of such an asset could be text, implying that it can be edited in the HTML text editor. Suppose the learning objective is to teach the student of the course how to sort an array in C language in descending order, then, this RIO can be adapted by editing the C language code snippet in the web page so that instead of sorting array elements in ascending order, it does the reverse (descending order). This implies that the RIO can now be reused in a RLO that teaches the student how to sort array elements in C programming language in descending order.

Figure 3 below presents a summary of a sequence of activities under RLO asset adaptation.

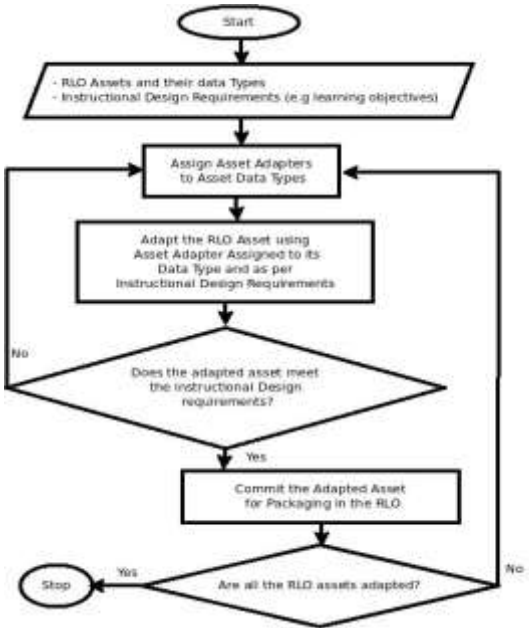

Figure 3: Sequence of Activities in RLO Asset Adaptation

2.3 RLO Asset Assembly

Based on instructional design strategies derived from instructional design requirements, the instructional designer assembles/packages the adapted assets of the RLO using existing instructional content sequencing techniques. For example, Sun *et al*[22] developed four instructional content sequencing techniques, namely; Information-Push(I-Push), Information-Pull (I-Pull), Information-Push-Pull (I-Push-Pull) and Information-Pull-Push (I-Pull-Push). Below, we present how and when such techniques can be used by the instructional designer in assembling RLO assets after adaptation.

i. **Using I-Push method**. With this instructional content sequencing technique, the instructional designer pre-selects the flow of content in the RLO. As far as the learners' personal learning styles[21] is concerned, I-Push is recommended for *Judgers (learners who like learning step-by-step while accessing materials)* while accessing the instructional content. This is because it

gives the learner a chance to access the content of the RLO step-by-step. I-Push method is also good in such a way that it protects the learner from losing the focus on the learning objective of the RLO. The I-Push method is not recommended for situations where the content of some of the assets of the RLO (like the content of RIOs) keeps referring to other assets of the RLO (unless such references are in the summary asset of the RLO). This is because the pre-selection of order and flow of content will limit the learner from jumping to the content that he/she wants.

ii. **Using I-Pull method.** With this instructional content sequencing technique, the instructional designer creates some kind of 'self-service' environment for the learner because it gives him/her "*some control over the presentation of the learning materials by choosing what they want to learn*" [22]. As far as the learners' personal learning styles [21] is concerned, I-Pull method is recommended for *Perceivers (learners who prefer a random global way of accessing learning materials)* while accessing learning materials.

iii. **Using I-Pull-Push method**. This method is a blend of both I-Pull and I-Push methods above. Using I-Pull-Push method, the instructional designer packages the assets of the RLO in such a way that gives the leaner some flexibility to access the content. The learner is given "*pre-defined information that has customized features*" [22]. As far as learners' personal learning styles[21] is concerned, this method is recommended for learners that are more of *Perceivers* than *Judgers*.

iv. **Using I-Push-Pull method**. Like the I-Pull-Push, this method is also a blend of both I-Pull and I-Push methods above. This method gradually allows the learner to access the content in the RLO by providing the learner with *"customized information that has pre-defined features*" [22]. As far as learners' personal learning styles [21] is concerned, this method is recommended for learners that are more of *Judgers* than *Perceivers*.

Below we present the sequence of tasks that an instructional designer can follow while assembling/packaging the adapted assets of the RLO.

Task I: Group the adapted assets as per standard structure of the RLO

It is important that the RLO metadata structure used at the time of decomposition is used to package the RLO assets, otherwise, more effort will be needed to adapt the old metadata structure to a new metadata structure. In this paper, we adopt the metadata structure presented in Figure 1 to assemble the RLO assets.

Task II: Choose Asset Sequencing Technique

Once the adapted assets have been grouped as per standard structure, the instructional designer must then choose the technique for asset sequencing that meets the instructional design requirements. The preferred method of accessing the content by the learners (learners' personal learning styles) largely influences the method of sequencing the assets of the RLO since the content is largely meant for the constructivist learner and not the instructional designer. Since the actual instructional content of the RLO is in RIOs, AOs and POs, the instructional content sequencing techniques discussed above can be applied on such assets of the RLO without affecting the standard structure of the RLO. For example, after asset adaptation, the instructional designer might realize that RIO number 3 (RIO3) depends on RIO2 and thus while packaging the assets, the learner should not be allowed to access the content of RIO3 before accessing RIO2. In such a case, the I-Push method would be applied (provided the learners are *Judgers*).

Task III: Submit the Adapted RLO for Vetting

Once the instructional designer is satisfied with the changes made in the RLO (as per instructional design requirements), the RLO can then be submitted for vetting (done by instructional content and subject experts) so as to ensure that only quality RLOs are published to the e-learning environment.

Figure 4 below provides a sequence of the actions that an instructional designer can follow in RLO asset assembly.

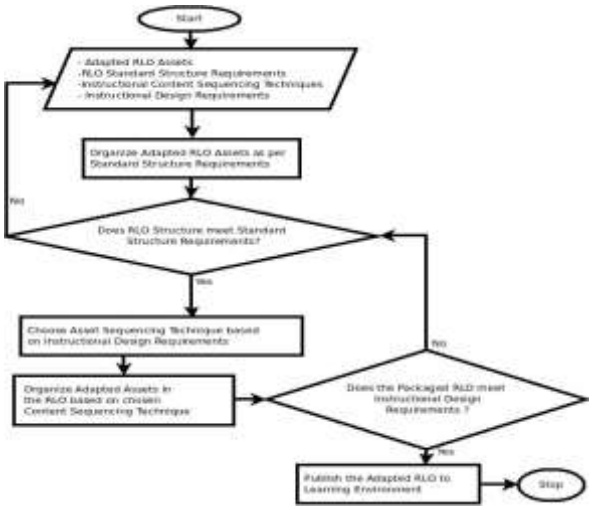

Figure 4: Flow chart showing a sequence of actions during RLO asset assembly.

3 METHODOLOGY FOR EVALUATION

We evaluate the above three-step process of RLO Decomposition, RLO Adaptation and RLO Assembly by developing a prototype (that instantiates the activities in the three-step process above) and using this prototype in experiments of adapting higher education RLOs with multi-format assets.

3.1 Development and Use of the Prototype

The prototype is a web-based application hosted online at www.loadaptor.com and it enables the instructional designer to decompose a RLO (selected from a repository) into assets following the metadata structure presented in Figure 1 above. The instructional designer is then allowed to adapt the assets and the repurposed assets are assembled into a RLO following the I-Push method of instructional content sequencing. The adapted RLO is then ready to be published into the e-learning environment after it has been vetted. Figure 5 below provides a sample of a RLO with its assets in this prototype.

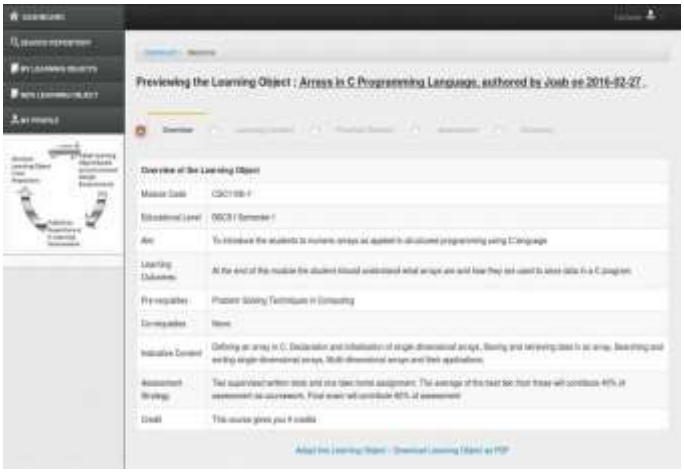

Figure 5. Sample of a RLO under preview in the prototype

3.2 Using the Prototype for Evaluation

We evaluate the functionality, correctness and usability of this prototype as far as RLO decomposition, RLO asset adaptation and RLO assembly are concerned. The choice of these three quality attributes is based on suggestions by Hevner *et al*[23] about evaluation of research artifacts.

Expert sampling technique[24] was used to select 8 experts in instructional design for higher education from 3 different universities in Uganda. Table 3 below is a summary of key demographic statistics about the characteristics of the experts in the sample.

Table 3: Summary of key demographic characteristics of the sample of experts

Highest Academic Qualification	**Post-Doctoral**	**PhD**	**Masters**
	2	5	1
Experience in Instructional Design	**Over 5 Years**	**3-5 Years**	**1-3 Years**
	5	2	1

These experts were requested to use the prototype to adapt RLOs with multi-format assets by following the above three-step process. After this experiment, a feedback questionnaire was given to each participant. In the next section, we present the results of the experiment.

4 EVALUATION RESULTS

1. **Functionality of the prototype.** Key functions tested were;
   A. The system can decompose a RLO into assets for adaptation.
   B. The user can adapt the RLO assets after decomposition.
   C. The user can assemble the adapted assets into a new RLO.
   D. The user can publish the new RLO to an e-learning environment.

   Each of the 8 experts was requested to rate the functionalities of the system (A to D) based on a five-point Likert scale of Strongly Agree, Agree, Not Sure, Disagree and Strongly Disagree. Figure 6 below presents a summary of responses from the 8 experts.

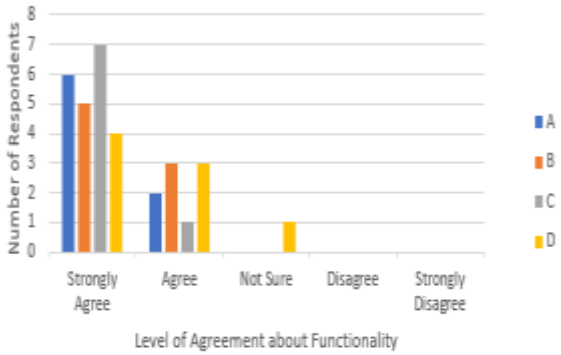

Figure 6: Summary of responses about functionality of the prototype

2. **Correctness of the RLO Adaptation Process in the prototype**
   The participants in this experiment were requested to evaluate the correctness of the prototype regarding the tasks presented under RLO decomposition, RLO adaptation and RLO asset assembly in the previous section. On a 5-point Likert scale of Very Correct, Correct, Not Sure, Wrong and Very Wrong, 5 out of 8 participants (62.5%) indicated that the process was Very Correct while 3 out of 8(37.5%) indicated that process implemented by the prototype was correct in RLO adaptation.

3. **Usability of the Prototype**
   The participants of the experiment were also requested to evaluate the usability of the prototype as far as the adaptation of RLOs with multi-format assets is concerned. On a 5-point Likert scale of Very Easy To Use, Easy to Use, Not Sure, Not Easy To Use, Very Hard To Use, 4 out of 8 participants (50%) indicated that it was Very Easy to use and another 4 out of 8(50%) indicated that the prototype was Easy to use in adapting RLOs with multi-format assets.

5 CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this paper, we proposed that reusability of RLOs with multi-format assets in instructional design can be improved through asset-level adaptation. We demonstrated how asset-level adaptation can be achieved through RLO decomposition, RLO asset adaptation and then RLO asset assembly. This three-step process was experimented using a prototype and the feedback results from the participants indicate that indeed this three-step process can enable instructional designers to adapt RLOs with multi-format assets.

5.2 Future Work

Although this research has demonstrated how RLO asset-level adaptation can be achieved through a three-step process as a way of improving reusability of RLOs with multi-format assets, there still remains challenges especially in RLO decomposition into assets. There is still no agreed standard size of an asset which would help to determine when decomposition should effectively end. Secondly, there is still a challenge of adapting assets using separate tools/applications where the instructional designer lacks technical expertise. An example here is when the RLO has source code as part of the assets and the instructional designer does not know the programming language the source code is written in. Attempts to adapt an asset using a tool that the instructional designer does not understand well will lead to authoring poor quality content. In addition, the learnability of the RLO created using such a three-step process needs to be measured. Such issues need to be investigated in future work.

REFERENCES

1. Berger, C., & Kam, R. (1996, October 18). Definitions of instructional design. Available online at http://www.umich.edu/~ed626/define.html
2. elearningDOM(2017). Instructional Design Process: A Step-By-Step Guide. Retrieved from https://elearningdom.com/instructional-design process-guide-step-by-step/
3. Piskurich, G.M.(2006). Rapid Instructional Design: Learning ID fast and right.
4. Ryder, M. (2014). Instructional Design Models. Available online at: http://carbon.ucdenver.edu/ mryder/itc/idmodels.html.
5. Schneider, D.K. (2006). Instructional Design Models and Methods. Online Learning in Diplomacy Workshop, Geneva, 30th May 2006.
6. Durso, C.(2013).Calibrate Research Base. Medallion Learning, July 2013.
7. Stoilescu, D. (2008). Modalities of Using Learning Objects for Intelligent Agents in Learning. *Interdisciplinary Journal of E-Learning and Learning Objects*, 4.
8. Agaba, J. E. and Lubega, J. T. (2018). Reusable Learning Object Metadata Adaptation for Higher Education Instructional Design, *International Journal of E-Learning and Educational Technologies in the Digital Media (IJEETDM)* 4(3): 66-74, The Society of Digital Information and Wireless Communications (SDIWC), 2018 ISSN: 2410-0439 (Online), September 2018.
9. Agaba, J. E. and Lubega, J. T. (2014). On Addressing Design Issues that Hinder Reusability of Learning Objects in Instructional Design, *International Journal of Information and Education Technology, (IJIET)*, Vol. 4, No. 1, pp. 25-28. ISSN 2010-3689.
10. Wiley, D.A. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In *D. A. Wiley (Ed.), The Instructional Use of Learning Objects: Online Version*. Available online at: http://reusability.org/read/chapters/wiley.doc.
11. Battou, A., El Mezouary, A., Cherkaoui, C., & Mammass, D. (2011). An Adaptive Learning System Architecture based on a Granular Learning Object Framework. *International Journal of Computer Applications*, 32(5), October 2011, 18-26.
12. Funda, D., & Kadir, E.(2011). A Personalized Learning System Based on Semantic Web Technologies, In *Slavomir, S., Vlado, G., & Marko, R.: Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation, and Evaluation*, pp. 258-284.
13. IEEE(2002). IEEE Standard for Learning Object Metadata. Standard No.: 1484.12.1-2002.
14. Noraniah, M.,Y.(2005). Learning Object Content Models: A Comparative Analysis. In *Proceedings of the Postgraduate Annual Research Seminar* 2005; pp. 87-91.
15. Devshri, R., Sudeshna, S., & Sujoy, G. (2010). A Comparative Study of Learning Object Metadata, Learning Material Repositories, Metadata Annotation & an Automatic Metadata Annotation Tool. In *Advances in Semantic Computing*, Vol. 2, pp. 103-126, 2010.
16. Gkatzidou, S. & Pearson, E.(2008). A vision for truly adaptable and accessible learning objects. *Proceedings of ascilite 2008, Melbourne*.
17. Gorgan, D., Stefanut, T., & Bacu, V.(2010), eGLE GiSHEO eLearning Environment, 6 September 2010, *ESA-ESRIN, Frascati, Italy*.
18. Boyle, T. & Bradley, C.(2009). File Formats Compatible with GLOMaker, London Metropolitan University, August 2009.
19. Wagner, E.D. (2002). The New Frontier of Learning Object Design, in *The E-Learning Developers' Journal*, June, 1-8.
20. Raghuveer, V. R., & Tripathy, B.K. (2012). An Object-Oriented Approach to Improve the Precision of Learning Objects Retrieval in a Self-Learning Environment, *Interdisciplinary Journal of E-learning and Learning Objects*, Vol. 8, 2012; pp 193-200.
21. Sun, L., Ousmanou, K., & Williams, S.(2004a). Articulation of Learners Requirements for Personalized Instructional Design in e-Learning Services. In *Liu et al.(Eds.): ICWL 2004, LNCS 3143*, pp. 424-431, 2004.
22. Sun, L., Ousmanou, K., & Williams, S.(2004b). Building Personalized Functions into Dynamic Content Packaging to Support Individual Learners, in *E-Learning Europa*, April 2004. Available online at :http://www.ais.reading.ac.uk/papers/con41buildi ng%20personalised.pdf
23. Hevner, A.R., March, S.T., Jinsoo, P., Sudha, R.(2004). Design Science in Information Systems Research. *MISQ 2004*, 28(1), 75-105.
24. Lund Research Ltd (2012). Purposive Sampling. Available at: http://www.dissertation.laerd.com/purposive-sampling.php.