A Pattern Language Based Learning Design Studio for an Analytics Informed Inter-Professional Design Community

Nancy Law[1], Ling Li[2], Liliana Farias Herrera[2], Andy Chan[2], Ting-Chuen Pong[3]

[1] Faculty of Education, University of Hong Kong, Hong Kong
nlaw@hku.hk
[2] Faculty of Education, University of Hong Kong, Hong Kong
[3] School of Engineering, Hong Kong University of Science and Technology, Hong Kong

Abstract. Over the past ten years, a lot of advances have been made in both the learning design and learning analytics research communities in terms of research outputs. On the other hand, these advances have not made visible impact on the everyday practices of the key stakeholders they aim to serve: teachers and learners. This paper argues that to advance both fields and to connect them into an iterative cycle of continuous improvement, we need a learning design pattern language in the Alexandrian tradition to guide learning design practice, and to connect designers' intuitive formulation of questions about learning processes and outcomes with evidence-based questions and operationalizable questions. It further describes the pattern language and the online tool, Learning Design Studio that the authors have developed to serve as an organically extensible language and a technology platform to support learning design, and inter-professional collaboration among learning designer and learning analytics communities.

Keywords: pattern language, learning design, learning analytics, learning design tools, design community

1 Introduction

As the call for this special issue indicates, there is a synergistic relationship between Learning Design (LD) and Learning Analytics (LA), and there have been increasing efforts to build the connection between these two areas to realize the anticipated synergy [1, 2]. Without explicit connections with the pedagogical design principles and intentions, the meaning and implications of learning analytics would be lost to teachers and other stakeholders. It is imperative that advances in these two fields can be articulated so as to contribute simultaneously to the enhancement of learning and learning design. LD can potentially offer LA a domain vocabulary, highlighting the elements of a learning system to which meaningful analytics questions can be applied. LA, when applied to address questions that are informed by the design assumptions and contexts, has the potential to guide feedback and intervention for learners and learning designers, to improve learning and learning design. However, there is not yet a commonly accepted domain vocabulary within the field of LD, or even who counts as a learning designer. For the field of LD to really prosper, we need to have a community of practitioners beyond researchers who identifies themselves as learning designers, who share the same vocabulary, language, and tools, and have a common understanding of what the kind of workplaces and roles of fellow practitioners in the same profession. Likewise, the field of LA is still very much dominated by computer science and engineering researchers interested in computational approaches to collecting, analyzing and visualizing the digital footprints of learners to make sense of what is going on in the learning process and to contribute to the enhancement of learning. However, the language used in this field is dominated by methods, techniques and algorithms, which does not speak directly to the concerns and issues that learning designers grapple with.

In this paper, we begin by reviewing the rationale and the need for developing a Learning Design Pattern Language in the Alexandrian [3] sense of the concept of pattern language in the literature, and a review of the state of development in both LD and LA fields. This is followed by a description of the Learning Design Pattern Language that we have developed to provide (1) a structure, vocabulary and interconnections across the conceptual elements of LD for practitioners that would also promote good pedagogical design principles, and (2) a structure and vocabulary for describing LA *questions* for different *stakeholders* and *purposes*. We also highlight where our work has been inspired by Alexander's pattern language, and where our work differs from current work on LD pattern language reported in the literature.

A living language is symbiotic with the host community it serves, supporting the community's communication and collaboration needs, and evolving with the activities and their advances. We have developed a technology platform, the Learning Design Studio (LDS^{HE}), to support learning and assessment design as well as design-aware learning analytics specification. In this paper, we provide a brief introduction to LDS^{HE} by describing: (1) how it embodies and supports LD work using the pattern language, (2) the empirical work we have done in using the platform and associated tools to document and visualize an actual MOOC course, and (3) how the pattern language can guide (a) the formulation of operationalisable learning analytics questions from learning designers' questions about students' learning that are sensitive to the design decisions made in the learning context, and (b) the possible actions that can be taken by different stakeholders such as students, teachers, instruction designers and institutional leaders, on the basis of the learning analytics outcomes. We hope that this paper can stimulate interest and adoption of LDS^{HE} such that both the pattern language and the LDS^{HE} tool itself can develop further organically while serving the learning design and learning analytics communities.

2 Learning Design and Learning Designers

Efforts to make learning designs explicit, visible, sharable and explorable date back to the '80s and '90s in the fields of intelligent tutoring systems [4, 5] and intelligent authoring [6]. Learning design as a recognizable field of research emerged in the beginning of 2000s [7, 8]. Learning design (LD) can be used to refer to the artefact or *product* resulting from the design process, which may be a plan or a formal description of the sequence of learning activities that can be realized on an e-learning platform [9]. LD is also used to refer to the *process* of developing a plan and the necessary environment and resources for a target set of learners to achieve specific learning outcome goals. When the focus is on the LD process, the purpose may be to facilitate the *coordination* of the different elements such as communication mechanisms, assessment strategies, collaboration mechanisms, which are carried out by different participants or stakeholders [10]. It could also be used to support teacher *inquiry* into students' learning as an integral part of teacher learning [11].

Irrespective of whether the focus is on LD as a process or a product, representational issues are central to LD, whether as a field of practice or research. When the focus is on LD as a product for the implementation of learning activities and resources on an e-learning platform, there is always an agreed formalism for representation in the form of technical languages and tools such that learning designs can be unambiguously implemented on technology platforms [12, 13]. The IMS-LD [12] and LAMS [14] are classic examples of this type of formalism that focus on representational issues at the implementation level. These formalisms are derived from the instructional design tradition [11] and aim to be pedagogically "neutral" in that these representations do not give preferential support to any specific pedagogical approach, nor are these designed to highlight the pedagogical differences among designs. An important consideration in the development of this type of formalisms and tools is the requirement for the designs to be reusable under different learning contexts, which often means that the designs are created independent of specific implementation contexts [15].

According to the Larnaca Declaration: "*The ultimate goal of Learning Design is to convey great teaching ideas among educators in order to improve student learning*". LD researchers sharing this priority generally take a process-oriented view of LD, and the representational issues are pedagogically motivated. As such the LDs generated serve as artefacts to mediate the professional discourse around pedagogical practices for the purpose of sharing and learning [8, 16]. The formalisms developed with this focus generally give priority to representations of the problem context and pedagogical goals targeted, as well as the rationale underpinning the pedagogical decisions. These usually also include efforts to identify *pedagogical patterns* that highlight the generic features of the problem context and the design principles that underpin the pedagogical decisions [17]. Examples of these representations include the Design Principles Database [18–20], the Australian Universities Teaching Council (AUTC) Learning Design project [21], the pedagogical patterns [17] and Pedagogical Pattern Collector [22] developed by Laurillard's team [23]. These representations are not designed to be pedagogically neutral, but highlight the intended learning context and the relevant design principles adopted, with the latter grounded on learning sciences research.

While process oriented representations are conducive to scaffolding pedagogical conversations, they are not designed to provide the structural features and levels of detail required for machine implementation of the designs. More importantly, there is no commonly accepted *design language* used by the different research groups sharing this same research orientation. The representations or patterns developed by the researchers listed in the previous paragraph do not have sufficient details for unambiguous implementation. There is a need for a language for learning design analogous to the use of musical notation for the representation of music [24].

The emergence of a design language needs to be symbiotic with the development of the design community it serves. One obstacle to the establishment of a learning design language is the absence of a community that is publicly recognized as learning design professionals. There is a thriving community of instructional designers, but their foci are more on the design of artefacts to support learning. At the level of the key pedagogical decisions regarding learning goals, tasks, interactions, assessment and feedback, these are made by teachers, who unfortunately do not generally perceive themselves as learning designers. In fact, there are no compelling reasons for teachers to see themselves as a design profession. Design professionals such as engineers and architects have tools, formalisms, and templates that embody the expertise and knowledge accumulated in the profession to simplify and expedite their work, as well as to collaborate in the design and implementation process with other professionals in the design and implementation process. When the technology and tools for learning are relatively simple and the key pedagogical decisions are made primarily by an individual without the need for coordination or orchestration with other professionals, there is no compelling reason for conceptualizing teaching as a design profession. However, with the popularization of online learning, and especially the emergence and popularization of MOOCs, teaching is no longer an individual activity, and much of the design and orchestration work needs to take place before a course starts.

In the next section, we will provide a brief description of the design pattern language proposed by Alexander [3] before we describe the learning design pattern language and tool that we have developed that can support the design and implementation of fully online as well as blended courses.

3 The Alexandrian Concept of Pattern Language—Core Features

Alexander's [25] pattern language originated from his observation that 'even though there are a million different versions of these acts and processes, there is one fundamental invariants feature, which is responsible for their success.' (p. 8). This led to his focused efforts to identify the 'invariants' underlying successful designs, and to make such design knowledge explicit for others to use through design patterns. Alexander [3] argues that a design pattern 'describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice' (p. x). The key elements of a design pattern are the 'problem', 'context' and the semi-structured description of the 'solution'. These core design patterns can be operationalized as a functional statement: "For problem P, under circumstances C, solution S has been known to work" (ibid). The power of this design pattern format, he believes, lies in its capacity to externalize not only the design practice, but also the design thinking behind it, thus making it easier to critique designs and improve design practices [26]. Alexander [3] calls the whole assemblage of the 253 design patterns he collated a pattern language. In addition, he also refers to any selected sequence of patterns from this language as a pattern language for a small part of the environment.

The Alexandrian pattern language is underpinned by a strong value commitment that architectural designs are *living structures* that sensitively address the actual requirements of the society and the real demands of users. His 253 design patterns embody the rationale of fostering and facilitating design configurations that enliven and promote the wellbeing of communities and societies. These range from an 'Independent Region' to an 'Ornament', arranged in a multilevel hierarchy that begins with the largest for regions and towns, followed by neighborhoods and clusters of buildings, and down to details of constructions. Each pattern is connected to 'larger' patterns above it and 'smaller' patterns below it in the language [3]. Underlying this multilevel hierarchy is Alexander's holistic design thinking that buildings do not exist in isolation, and should be designed to make the larger world more coherent and whole (ibid). The multilevel structure of design patterns embodies a conceptual design framework that prioritizes coherence while facilitating the creation of innumerable alternatives and an infinite variety of details [27].

4 Alexandrian Pattern Language in Non-Architectural Design

While Alexander's pattern language originated in the context of architectural design, the merits of the pattern language approach have transcended disciplinary boundaries and inspired researchers in other fields, such as software engineering [28], hypermedia [29] and interaction design [30, 31]. Gamma and colleagues [24] developed a pattern format comprising four essential attributes: 1) the name of the problem; 2) a description of the problem context; 3) an abstract description of the solution to the problem; and 4) the consequences of applying the patterns that are the results and trade-offs. This standard format has been widely used by software engineers in documenting and communicating software design patterns, which are shared online as a storehouse of solutions to help each other solve recurring design problems. The adoption of the pattern language also facilitated the development of design communities [32, 33], resulting in dozens of patterns that simplify the complex expression of design models.

The concept of design pattern language has also attracted the attention of researchers in the learning design community. Peter Goodyear [34–36] is among the scholars who introduced Alexander's concepts of patterns and pattern languages to the learning design community through his systematic exposition of the potential value of these concepts to educational design and technology-enhanced learning. One pioneering project in this line of work is the Pedagogical Pattern Collector (PPC) developed as part of the project on Learning Design Support Environment led by Diana Laurillard [37]. PPC is an online software that collects and archives generic pedagogical patterns which contain the key elements of a learning design, such as the title, learning outcomes and methods. This standard format allows a group of teachers to share, build on and improve each other's work in an online community (ibid).

The Participatory Pattern Workshops initiated by Yishay Mor and colleagues [38] is another way to capture good design practice through patterns. By collaboratively reflecting on the challenges and solutions in teaching, groups of practitioners and researchers have together developed a set of good pedagogical design narratives, scenarios and patterns that can be used to enhance learning through technology (ibid). The pedagogical patterns put forward by Joseph Bergin and colleagues [39] are also a product of their collaborative work with educational practitioners. The patterns they identified, such as making failure a part of the learning process ("Build-in Failure"), and engaging students in active learning ("Active Student") (ibid), focus more on promoting the tactical skills that teachers need in everyday teaching and learning.

Design patterns have also been adopted in the development of holistic environments to support collaborative design work that comprise not only digital design tools, but also the broader physical settings and a wider range of tools. The Educational Design Studio (EDS), developed by Martinez-Maldonado and colleagues [40], provides a complex ecology of tools and resources (e.g., multi-touch surface devices, interactive whiteboard and writable walls) to facilitate the interaction and collaboration of a small design team with a mixture of expertise (e.g., teacher, instructional designers and educational technologies). The design patterns, which are implemented in one of devices-the CoCoDeS tabletop [41], provide a variety of candidate design solutions to scaffold the design process.

5 Overview of the LDS Pattern Language

To underpin our LD pattern collection and tool development work, we have invested an intensive effort to develop a LD pattern language. As this pattern language underpins the LD platform, Learning Design Studio (LDS), that we have developed, we will refer to this pattern language as the LDS pattern language for ease of reference.

Our review of research suggests that attempts in adopting pedagogical patterns in learning design has been fruitful, producing a wide range of patterns under different themes, at different design levels, and being implemented in different design environments. However, there is one major difference between these patterns and Alexander's design patterns in that the latter is underpinned by a well-structured design language that embodies the different levels of granularity in design, and highlighting the relationships between them, while the former only focus on patterns as standalone entities, and are represented in different formats and terms.

Our work on the LDS pattern language is very much inspired by the Alexandrian model that it should comprise a *value system* that provides guidelines for attributes of good learning designs; a *living structure* that facilitates the natural flow of systematic and coherent design; and a *communication system* that enables learning designers (e.g. teachers, instructional designers, learning technology developers) to articulate their design ideas and collaborate in the construction and delivery of the designs. On the other hand, there is one significant difference. We take the language as a formalism for the construction, operationalization and communication of designs. It comprises a vocabulary on the entities and relationships among entities in learning design, and rules (which can be construed as a grammar) on how different design elements can legitimately be connected. Design patterns are important elements in learning design, but the LDS pattern language is *not an assemblage of design patterns*. Instead, it is a language to be used for the construction of learning design patterns. We present the key features of the LDS pattern language in the remainder of this section.

5.1 A Value System to Guide Learning Design

Alexander used the phrase 'quality without a name' to describe the central quality of successful designs. From learning sciences research, we know that learning does not only take place through being instructed (or directed learning), and is likely to be more effective if (1) learners have more opportunities to exercise their agency in the learning process, such as in exploratory learning, the productive creation of artefacts or solutions, or reflective learning; or (2) interactions with peers and others are involved. Hence the quality of the learners' experiences is of primarily importance in influencing learning outcomes, and the quality of instruction matters only as far as the instruction's impact on learners' experiences. The LDS pattern language should facilitate design thinking as well as provide strategies and tools that foster and support designs that align with the following design principles:

- The learning tasks should be designed to provide learners with experiences that are more likely to bring about the targeted learning outcome goals;
- Learning designs should promote student agency for self-directed learning, encourage student sharing and collaboration;
- Good assessment designs should make learning outcomes visible, and would make more contribution to learning if assessment is integrated as an integral part of the learning process, and the assessment criteria are made explicit;
- Appropriate, just-in-time feedback should be designed to promote learning;
- The learning environment should empower learners by providing them with tools and resources needed to communicate and construct;
- Learning designs should focus on learners' experiences rather than the work of the teacher or instructional designer;
- Learning is primarily social, hence LD needs to take account of the social organization of learning, including the management aspects such as the formation/ assignment of groups;
- The learning environment, including the social, physical and digital dimensions are integral parts of the design

5.2 A Living Structure to Scaffold the Configuration of Design Patterns

A design language provides an overarching schema for designing and orchestrating the design elements at different levels of granularity, highlighting the connections within and across different levels of the design, and their alignment. Inspired by the holistic design thinking embedded in Alexander's hierarchically-structured pattern language, the LDS pattern language is designed to foreground the key granularities that comprise the design elements of a course. This is similar to the multi-level pedagogical framework that Goodyear [34] proposes in applying the pattern language concept to designs for networked learning. The LDS pattern language specifies three hierarchically nested levels of granularity: course, learning unit and learning task (Fig. 1). Learning design should arguably start at the program level to ensure coherence. However, at this stage, we start with a course as the highest level granularity of design in the LDS pattern language. The *course level design* sets out the overall course structure in terms of the intended learning outcome objectives, as well as the total learning time and the time allocated to each session. The *learning unit level design* specifies the set of learning units needed to achieve the course outcomes. Each learning unit comprises task sequences organized in the form of design patterns according to explicit pedagogical design principles that addresses a particular type or constellation of learning outcomes. Pedagogy/pedagogical approaches are realized through the learning design patterns encapsulated in the learning units. Finally, the *learning task level design* specifies the context, duration, tool(s), and resource(s) needed for satisfactory implementation of the task concerned. This hierarchically nested design sequence follows the 'living structure' concept in Alexander's pattern language, going from larger structures to smaller structures, to "ones that embellish those structures, and then to those which embellish the embellishments" [3]. The multilevel, hierarchically nested design structures provide teachers and other involved designers with easy navigation and control during the complex design process, allowing them to zoom in or out easily to focus on the design work at different levels from the entire course to detailed design of a specific learning resource without losing context and coherence.

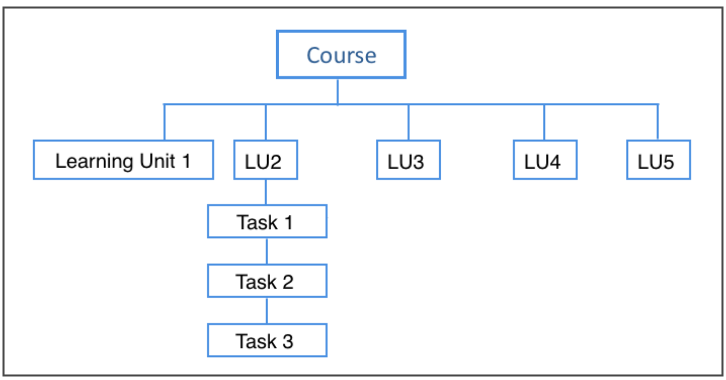


Fig. 1. The hierarchically embedded 3-level structure of a learning design for a course.

5.3 A Communication System among Designers

A design language is a communication system that allows designers to articulate their design artefacts and design thinking, for communication and dialog with fellow learning designers, learners, and other education professionals such as teacher educators and learning analytics researchers. The LD pattern language possesses a system of vocabularies that identifies design components at each level of granularity. This starts with a set of context descriptors that specify the context of learning at the course level, such as 'course title', 'class size', 'total learning hours', 'number of sessions', 'prerequisites', and 'learning outcomes'. There are three categories of outcome goals: 'disciplinary knowledge', 'disciplinary skills', and 'generic skills'. These context descriptors help to frame and communicate the overall requirements that the learning design needs to address. It is important to note that the total learning time includes structured "contact time" in face-to-face and blended learning modes, as well as the time that students are expected to spend on course-related study activities.

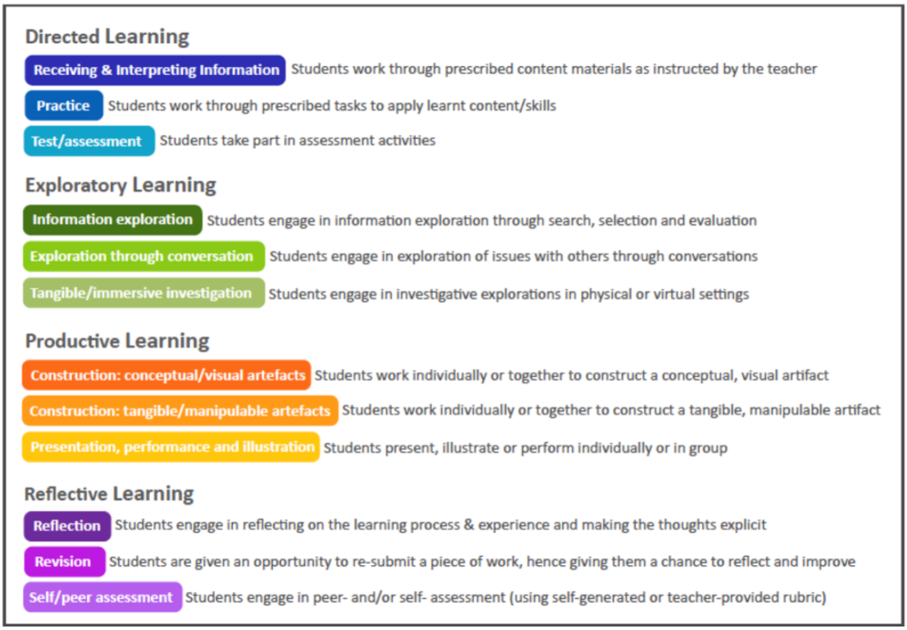


Fig. 2. The learning task taxonomy.

Central to the pattern language is a taxonomy for teachers to construct task sequences to form coherent learning design patterns that address specific types of learning outcomes. The taxonomy is designed to capture the nature of the experience that each type of task will provide for the learner. A task describes the activity as designed for the learner to undertake, including the intended role of the learner and the nature of the anticipated experience through the activity. The task taxonomy in the LDS Pattern Language is designed to foregrounded 21st century pedagogies, which put emphases on the role of learners as active and self-directed, taking responsibility and agency for the learning process, with the instructor serving as a facilitator and motivator. Specifically, the task taxonomy comprises 12 task types (Fig. 2), which can be further grouped into four categories. The directed learning category includes three activity types: receiving and interpreting information (which can be provided in different forms of media), practice, and test/assessment. The exploratory learning category also has three activity types: information exploration (e.g. information search and evaluation), exploration through conversations (e.g. face-to-face and forum discussions), and explorations through tangible/immersive investigations (e.g. experiments, simulations, role plays). Learning activities that engages students in the creative construction of conceptual/visual artefacts (e.g. essays, concept maps, art work), tangible/manipulable artefacts (e.g. models, programs, robotic systems), or performances and presentations are grouped under the productive learning category. The fourth category is reflective learning, which includes three activity types: creating a reflection artefact (e.g. a reflection journal, an e-portfolio), revising earlier work, or doing peer- or self- assessment.

As Goodyear and Dimitriadis [42] point out, one cannot assume that learners are fully compliant in doing exactly what is designed for them. To differentiate between the designed tasks and the actual activities undertaken by students, this taxonomy is referred to as learning activity taxonomy when used to describe the actual types of activities observed during learning time.

The social organization of the learning tasks constitutes another important design dimension. Using conventional classroom situations as the reference, there are four types of social organizations of learning: whole class, individual, peer (e.g. peer assessment, peer teaching), and group work. There are fewer types of whole class learning tasks that are conducted online, examples of which include discussion forums and wiki construction designed for the entire class.

There is a set of vocabularies that specifies the features of the learning environment at each level of the design granularity. These include: mode of delivery (face-to-face only, fully online, or blended) at the course level; tools (e.g. a learning management system and the functionalities that it supports) and resources (e.g. videos and readings for a specific topic) at the task level. Assessment and feedback are important design elements, and the Language will prompt the designer to identify the format used, the type of outcome goals addressed and the level of granularity these serve.

5.4 Design-aware Learning Analytics and Visualization Specifications for Actionable Improvements

Learning analytics is a young field that has gathered momentum with the increasing adoption of virtual learning environments (e.g., blackboard, Moodle and MOOC platforms), and hence the ready availability of large sets of learning-related data (e.g., personal data, system information, learning behaviors and scores) [43]. Analytic tools such as Temporal Analytics, Cohort Dynamic Analytics and Comparative Analytics examine the patterns of individuals or groups of students' learning activities (e.g., assignment submission, participation in online forum or dropping out from a course) and provide generic analytics that are devoid of pedagogy specific links to the course designers' intentions [44]. There is another category of learning analytics tools that can be aligned with pedagogical approaches. For instance, Social Network Analysis can be used to explore and understand students' interactions in social constructivist designs [45]; Sentiment Analysis is relevant to designs aimed at increasing motivations or incorporating social emotional strategies to enhance learning [46, 47];

and discourse analysis can be helpful in finding out if students were able to construct better arguments [48]. This latter type of learning analytics tools can help us to understand students' learning behavior at a fine-grained level, and potentially flag problems that can be followed up in the teaching and learning process.

Despite the advances in the development of learning analytics tools, the deployment of tools for pedagogically relevant analytics about students' learning has mainly been confined to learning analytics researchers, computer scientists, data scientists and engineers, who often are the developers of the analytic tools and methods used. Education practitioners outside of this community sadly often can only name or request generic statistical descriptors belonging to the first type of analytics described above [44]. More importantly, the application of learning analytics have been done in an ad-hoc manner, often driven by the availability of certain tools rather than a systematic approach based on the specific learning context. The absence of an interpretative pedagogical framework to guide the analytics application renders the evidences gathered fragmentary, and the connection with the learning context obscure. This has led to the fact that, when being provided with learning analytics feedback, teachers tend to be uncertain about how to interpret and act upon them.

Following the philosophy that learning design should be an iterative process of improvement, and that learning analytics should be meaningfully connected to learning design, the LDS Pattern Language also provides a system of vocabularies to connect aspects of learning designs at different granularities to appropriate analytic and visualization tools as well as actionable recommendations. Depending on the specific design context, the pattern language will suggest relevant "*designer's questions*" such as whether students were able to understand the targeted conceptual knowledge for a specific learning activity sequence. Given a specific designer's question, one can identify "*evidence-based question(s)*" that can provide appropriate answer(s). If the necessary evidence (i.e. data) is available from the online system, then one can generate the "*operationalizable analytic question(s)*", which could then be addressed by relevant analytic tools. This approach also means that multiple analytics tools may be used to address the same *operationalizable analytic question*. While the questions designers want to ask can remain rather stable, the methods and tools for their solution can be updated as the learning analytics field advances.

By starting with learning designers' questions as an integral part of the LD pattern language, the operationalized learning analytics conducted would always be embedded within the context of a specific learning design, providing a coherent framework for the formulation of meaningful learning analytics questions, the interpretation of the results, as well as for subsequent follow-up actions. We are developing a taxonomy of designers' questions, which in turn can be mapped onto a taxonomy of evidence-based questions, and then further connected to operationalization through specific tools

The above is only a sketch of the key elements of the Pattern Language. Each type of design element will be associated with a full list of attributes that are needed to provide the context and purpose of the element as well as its implementation details. The whole system of vocabularies should give the teacher and/or the instructional designer 1) a disciplined method for design; 2) a language to express their design and design thinking; 3) artefacts and a well-structured formalism for sharing, communication and collaboration around learning design with other professionals; 4) a ‘library’ of design patterns from which teachers can select and customize in their practice as a design professional; 5) a pedagogical learning resource in the form of design patterns that exemplifies the creative application of pedagogic theories into learning designs that can be adapted for implementation in different settings and contexts, and 6) a taxonomy and catalog of questions for inquiry of student learning and learning design linked to appropriate learning analytics tools.

6 Operationalizing the LDS Pattern Language through the Learning Design Studio

In this section, we provide an overview of the Learning Design Studio (LDS^{HE}), the technology platform we have developed to embody a learning design workflow underpinned by the values and structure of the Pattern Language, and facilitated by the design tools and repository of design patterns and resources provided by the platform. The superscript in the acronym indicates that this platform is customized to serve the Higher Education sector. To give readers a more concrete understanding of how the LDS^{HE} works, we illustrate how the platform can be used to capture the learning design for an actual MOOC course on *Introduction to Java Programming* that has been offered on the edX platform.

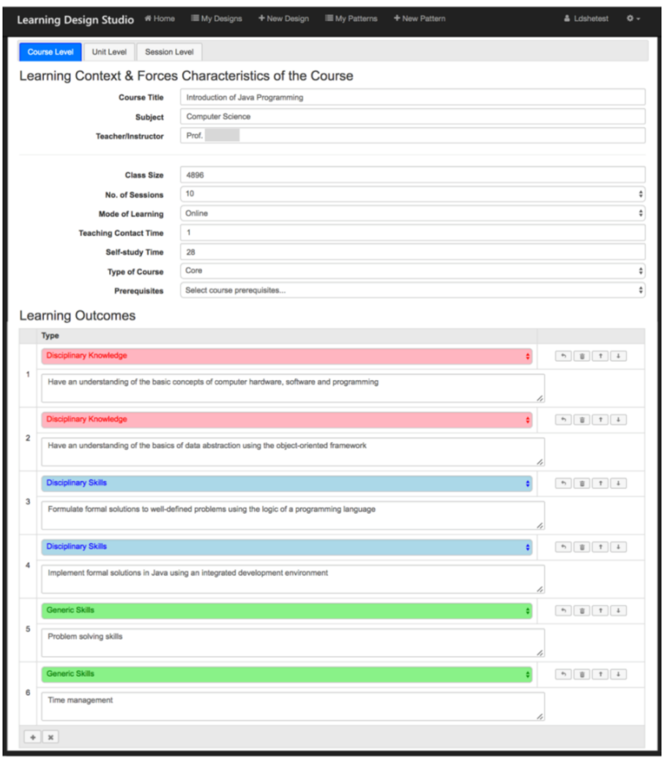


Fig. 3. The LDS^{HE} interface for the course level design.

A new course starts with a course level design page (Fig. 3), which invites the designer to enter basic course information as well as the intended course level learning outcomes listed under the three outcome categories.

The next level of design granularity is the Learning Unit (LU). Learning Unit 2 in this course addresses an important category of learning outcome in Java programming: the disciplinary knowledge connected with learning to program. The conceptual content is broken down into smaller chunks and sequenced to be addressed across the 10 weekly sessions. Each chunk is then addressed through the same sequence of learning activities, i.e. the same LD pattern (see Figure 4 a & b).

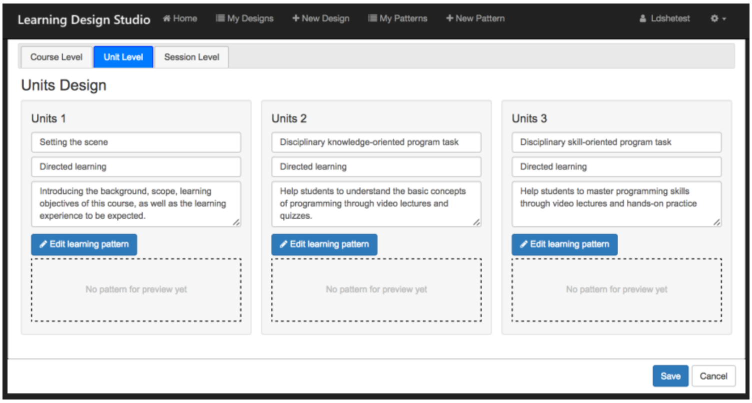


Fig. 4a. The LDS^{HE} interface for creating new learning units.

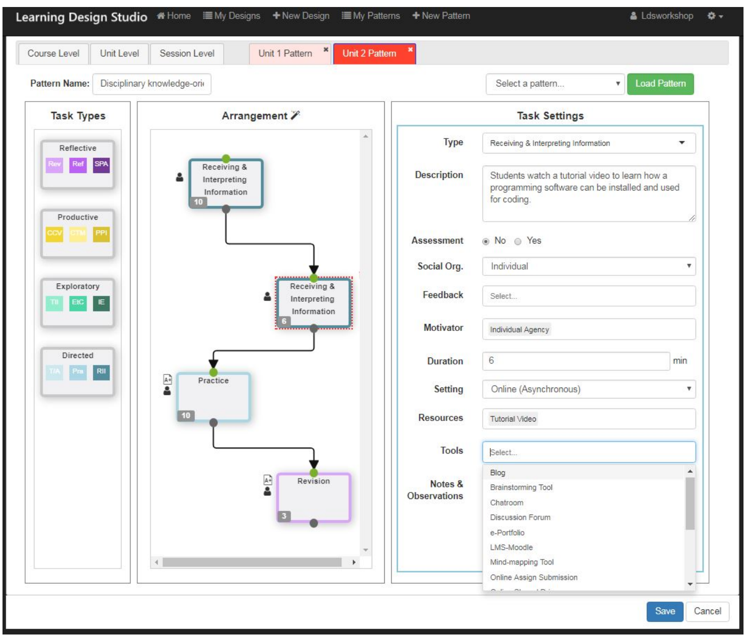


Fig. 4b. The LDS^{HE} authoring environment for constructing learning task sequences.

As shown in Fig. 4b, the user can drag the learning task types required from the 12 task types organized in four categories (directed, exploratory, constructive and reflective) on the left-hand panel column to the middle panel, and connect them according to the planned learning sequence. Activity sequences can be saved as a design pattern, to be instantiated for actual implementation. Further details for each activity such as time duration, social organization and resources required can be specified on the right-hand panel, which thus supports design work at the task level and below.

Courses in higher education are generally well-structured in terms of having a set number of sessions with an expected number of study hours per session. Each session may comprise learning tasks from several LUs, and the learning tasks within one LU may be distributed to different sessions. One logistic step in the course design process is to assign tasks or task sequences from the LUs to the different sessions. This is an interactive process, and the designer can fluidly move between task assignment to the sessions and the actual design work.

In developing LDS^{HE}, we are very mindful that learning design, like other design practices, is an iterative, continuing improvement process that should not be regimented to follow a strict workflow. Hence, although the LDS^{HE} provides a clear structure and logical workflow for the design process, we do not assume that course design happens in a linear fashion. Instead, LDS^{HE} is designed to allow teachers and instructional designers to navigate easily across levels and design elements and make modifications. Further, since the different levels of design are hierarchically connected, the system will also automatically synchronize the changes (e.g., the duration of a learning activity) so that changes made at one level and would be propagated to all affected levels in the course design.

7 The Learning Design Dashboard for Reflective Design and Improvement

LDS^{HE} provides a Learning Design Dashboard to facilitate teachers' self-monitoring and self-reflection during the design process. This dynamic and interactive dashboard provides simple visualizations on different aspects of the learning design under construction at different levels of granularity, including the percentage of learning time designated to different types of learning tasks, time allocation for tasks targeting the learning of different outcome goals, the ratio of tasks across the different forms of social organization, resources and tools to be specified, the ratio of targeted learning outcomes that are assessed, etc. Most importantly, the Dashboard is designed to heighten the designer's awareness during the design process of the alignment of their LD with the design principles underpinning the LDS Pattern Language, such as prioritizing student agency for self-directed learning, encouraging sharing and collaboration, developing reflective learning, and empowering learners by providing them with tools and resources needed to communicate and construct. The Dashboard infographics are designed to facilitate teachers' self-reflection and guide the refinement of learning designs towards the values that the pattern language advocates.

7.1 Course Level Dashboard Displays

The course level design progress chart (see Fig. 5a & b) helps the designer to monitor the extent to which the course level parameters set up at the start has been addressed by the design in progress. In Fig. 5a, the visualization at the top left shows the percentage of intended learning outcomes that has been addressed by the designed tasks. The table at the top right compares the learning time already designed with the scheduled learning time for each sessions in the course. The bar graph at the bottom shows the designed learning time devoted to each of the eight intended learning outcomes of this course.

It is important in learning design to ensure that there are ways to assess the extent to which various intended learning outcomes are achieved and appropriate feedbacks are given to students. The assessment and feedback alignment meter in the Dashboard (Fig. 5b) provides a simple visualization of this aspect of design coverage, which when clicked, will show the details.

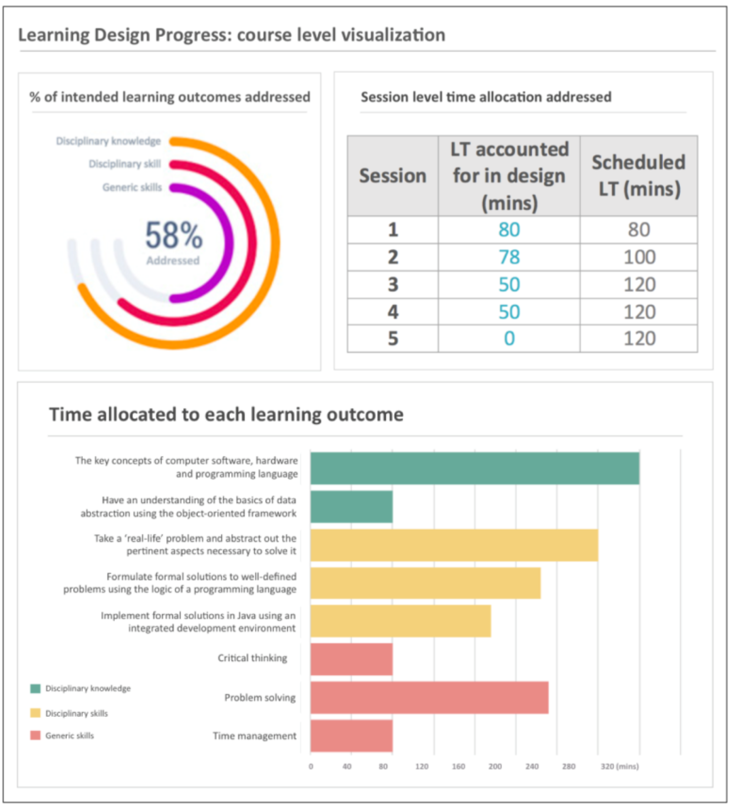
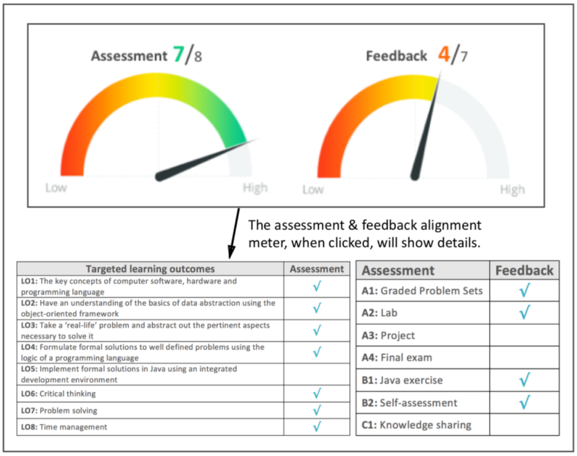


Fig. 5a. The LDS^{HE} course level design Dashboard.



Targeted learning outcomes	Assessment
LO1: The key concepts of computer software, hardware and programming language	√
LO2: Have an understanding of the basics of data abstraction using the object-oriented framework	√
LO3: Take a 'real-life' problem and abstract out the pertinent aspects necessary to solve it	√
LO4: Formulate formal solutions to well defined problems using the logic of a programming language	√
LO5: Implement formal solutions in Java using an integrated development environment	
LO6: Critical thinking	√
LO7: Problem solving	√
LO8: Time management	√

Assessment	Feedback
A1: Graded Problem Sets	√
A2: Lab	√
A3: Project	
A4: Final exam	
B1: Java exercise	√
B2: Self-assessment	√
C1: Knowledge sharing	

Fig. 5b. The LDS^{HE} Dashboard for assessment and feedback alignment check.

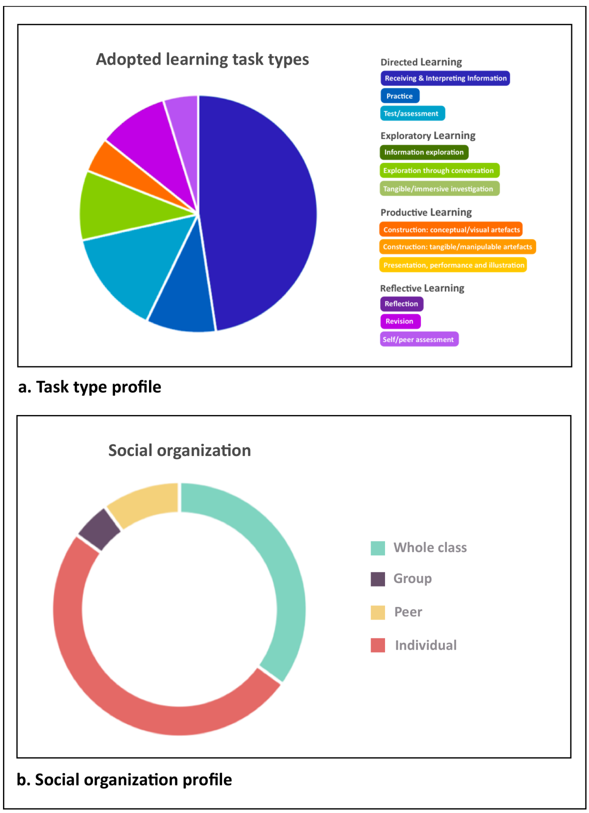


Fig. 6a&b. Pedagogical visualizations in the LDS Dashboard.

7.2 Pedagogical Visualizations

An important aspect of learning design is the pedagogical approach adopted. Currently, the dashboard provides two types of pedagogical visualization: the task type profile (Fig. 6a) and the social organization profile (Fig. 6b). The designer can choose the level of granularity for displays to visualize: the entire course, a specific learning unit or a learning design pattern. When the system has accumulated more design patterns, one can also visualize the proportion of time allocated to different types of design patterns.

8 Empirical Evaluation of the Pattern Language and LDS^{HE}

A preliminary evaluation of the beta version of the LDS^{HE} prototype was conducted on a MOOC course titled *Introduction to Java Programming* that has been offered on the edX platform. This was carried out in two studies:

In Study 1, a group of six Master students majoring in IT in Education, and with no background in Java programming was given a tutorial on using LDS^{HE}, was invited to take this MOOC course and then document the course design of this course independently as experienced by them in the capacity of a learner. The six designs were compared for similarity and differences, and a focus group interview was conducted to find out if they found any difficulty or ambiguity in the process. The students were given three weeks to complete their course design documentation. From their submitted work, we could see that the LDS^{HE} enabled them to effectively capture and communicate 1) the learning design patterns adopted in this course, 2) the pedagogical approach embedded in each pattern, 3) the learning context in which each pattern was situated, and 4) the particular learning outcomes that each pattern tackled. We are also encouraged by the observation that the six representations are consistent with each other, which indicates that the LDS pattern language can be used appropriately by non-experts in the representation of the learning design as they experienced during the course. The six students indicated during the focus group interview that the formalism and structure of the LDS pattern language, as well as the features of the LDS^{HE}, are easy to understand. They did not report any critical difficulties in documenting the course design. However, they helpfully pointed out some missing vocabularies for fuller representations of the task types and task settings. The LDS pattern language and the LDS^{HE} were refined based on their feedback.

In Study 2, two research team members with no Java programming knowledge took the MOOC, documented the course independently on LDS^{HE}, compared and agreed on a consolidated representation, then presented the learning design captured to the course teacher and his two instructional designers separately to seek their feedback on (i) whether the learning design captured was accurate, (ii) whether they perceive any advantage in using this form of representation, and (iii) if they have any suggestions for the improvement of LDS^{HE}. The instructor and instructional designers agreed with most of the documented learning designs of their course. Their major disagreement lied in the matching we made between the course's targeted learning outcomes and the learning outcome types of the LDS pattern language (i.e., whether they fall within the category of 'disciplinary knowledge' or 'disciplinary skills'). The mismatch was mainly due to our research team members' lack of domain knowledge. Regarding the usefulness of the course representation in LDS^{HE}, there were some key aspects that the instructor and instructional designers found valuable. First, they appreciated the systematic approach encapsulated by LDS^{HE} in the documentation of the course design. "*In the process of developing a course […], I don't have the time to focus on the entire course structure and entire learning outcomes. Through this pattern language, I can […] view how the course is delivered in a much more structured way*" (Instructional Designer 1). Second, they appreciated the pedagogical values embedded in the LDS^{HE}, which promote a stronger pedagogical orientation in their design thinking: *"[I used] to hold a simple model of course design […] now using the activity taxonomy, [I think] I can create more sophisticated [designs]*" (Instructional Designer 2). Third, the instructor was stimulated to modify his learning design through reflecting on visualizations presented on Design Dashboard. For example, on seeing the visualization on the profile of learning task types adopted in the course (Fig 6a), the instructor was surprised that his course had such a high proportion of time devoted to directed learning tasks (the portions in blue), and he indicated that *"I will increase 'explorative learning' (the portion in green) in the future"*. The research team is working on a fuller reporting and analysis of these two studies.

9 Discussion and Conclusion

We are encouraged by the pilot evaluation responses to the LDS pattern language and LDS^{HE}. Both the language and LDS^{HE} will be further developed, and more evaluation studies of LDS^{HE} in different course contexts are ongoing. Pending increased adoption of LDS^{HE} by teachers, instructional designer and other learning design related professionals, we hope that LDS^{HE} can eventually serve as a knowledge management system for archiving and sharing of good learning design patterns, as well as a platform to scaffold meaningful LD and LA discourses. As a formalism for constructing LD patterns, the LDS pattern language can serve as a framework to examine possible synergies across LDS^{HE} and existing LD tools and libraries such as the Pedagogical Patterns Collector [22] and the Design Principles Database [20].

Any language or tool can only thrive if there is an active community that uses and extends them. Our hope is that the Pattern Language and LDS^{HE} can serve as a leverage point for the LD & LA communities to work collaboratively together to realize the potential synergies to advance theory and practice in both fields. We foresee innovative pedagogical designs and learning analytics and visualization technologies to advance even faster in the years to come. Hence the LD Pattern Language is designed to provide a stable structure that is extensible to capture new design knowledge and practices, and the LDS^{HE} will also be open for further technological extensions by user communities. In particular, we hope that LDS^{HE} can serve as a repository of knowledge, tools and resources to support sharing and collaboration among members of the LD and LA communities. By providing a language and a technology platform that serve and connect both researchers and practitioner communities in LD and LA, we hope that advances in these two fields can contribute towards achieving the impacts promised by advances in pedagogical theories, learning technologies and big data analytics.

Acknowledgments. We acknowledge the Innovation and Technology Commission of the HKSAR Government for funding this study (Innovative Technology Fund, grant number ITS/206/15FP). We also thank the six MSc ITE students at HKU who helped to pilot the LDS pattern language, and the instructor and instructional designers of the Java MOOC course, for their kind support to the study. We are also grateful to the two anonymous reviewers who gave valuable feedback to guide the revision. Finally, we would like to express our deep appreciation to Professor Peter Goodyear at The University of Sydney for his insightful comments on an earlier version of this paper.

References

1. Bakharia, A., Corrin, L., de Barba, P., Kennedy, G., Gašević, D., Mulder, R., Williams, D., Dawson, S., Lockyer, L.: A Conceptual Framework Linking Learning Design with Learning Analytics. In: Proceedings of the Sixth International Conference on Learning Analytics & Knowledge. pp. 329–338. ACM, New York, NY, USA (2016)
2. Lockyer, L., Heathcote, E., Dawson, S.: Informing Pedagogical Action: Aligning Learning Analytics With Learning Design. Am. Behav. Sci. 57, 1439–1459 (2013)
3. Alexander, C., Ishikawa, S., Silverstein, M.: A Pattern Language: Towns, Buildings, Construction. OUP USA (1977)
4. Goodyear, P. ed: Teaching knowledge and intelligent tutoring. Intellect Books (1991)
5. Self, J. ed: Artificial Intelligence & Human Learning: Intelligent Computer-aided Instruction. Routledge, Chapman & Hall, New York, NY, USA (1988)
6. Tennyson, R.D. ed: Automating Instructional Design, Development, and Delivery. Springer Berlin Heidelberg, Berlin, Heidelberg (1994)
7. Agostinho, S., Bennett, S., Lockyer, L., Harper, B.: The future of learning design. Learning, Media and Technology, 36(2), 97-99. (2011)
8. Mor, Y., Ferguson, R., Wasson, B.: Editorial: Learning design, teacher inquiry into student learning and learning analytics: A call for action. Br. J. Educ. Technol. 46, 221–229 (2015)
9. Oliver, R., Harper, B., Wills, S., Agostinho, S., Hedberg, J.: Describing ICT-based learning designs that promote quality learning outcomes. Rethink. Pedagogy Digit. Age Des. Deliv. Elearning. 64–80 (2007)
10. Conole, G.: Designing for learning in an open world. Springer Science & Business Media (2012)
11. Hansen, C.J., Wasson, B.: Teacher Inquiry into Student Learning:-The TISL Heart Model and Method for use in Teachers' Professional Development. Nord. J. Digit. Lit. 10, 24–49 (2016)
12. Koper, R.: Current research in learning design. Educ. Technol. Soc. 9, 13–22 (2006)
13. Masterman, E.: Activity theory and the design of pedagogic planning tools. In: Handbook of research on learning design and learning objects: Issues, applications, and technologies. pp. 209–227. IGI Global (2009)
14. LAMS: https://www.lamsinternational.com/
15. Larnaca Declaration: https://larnacadeclaration.wordpress.com/full-document/, (2012)
16. McAndrew, P., Goodyear, P., Dalziel, J.: Patterns, designs and activities: unifying descriptions of learning structures. Int. J. Learn. Technol. 2, 216–242 (2006)

17. Persico, D., Pozzi, F.: Informing learning design with learning analytics to improve teacher inquiry. Br. J. Educ. Technol. 46, 230–248 (2015)
18. Kali, Y.: Collaborative knowledge building using the Design Principles Database. Int. J. Comput.-Support. Collab. Learn. 1, 187–201 (2006)
19. Kali, Y., Levin-Peled, R., Ronen-Fuhrmann, T., Hans, M.: The Design Principles Database: A multipurpose tool for the educational technology community. Des. Princ. Pract. Int. J. 3, 55–65 (2009)
20. Design Principles Database, http://www.edu-design-principles.org/dp/designHome.php
21. Learning Designs, http://www.learningdesigns.uow.edu.au/
22. Pedagogical Pattern Collector, http://web.lkldev.ioe.ac.uk/PPC/live/ODC.html
23. Ljubojevic, D., Laurillard, D.: Theoretical approach to distillation of pedagogical patterns from practice to enable transfer and reuse of good teaching. Teach. Engl. Technol. 11, 48–61 (2011)
24. Mor, Y., Craft, B., Maina, M.: Introduction: Learning design: Definitions, current issues and grand challenges. Art Sci. Learn. Des. (2015)
25. Alexander, C.: The Timeless Way of Building. Oxford University Press (1979)
26. Alexander, C.: Notes on the Synthesis of Form. Harvard University Press (1964)
27. Salingaros, A.: The Structure of Pattern Languages. Archit. Res. Q. 40, 149–162 (2000)
28. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software (Adobe Reader). Pearson Education (1994)
29. German, D.M., Cowan, D.D.: Three hypermedia design patterns. In: 2nd Workshop in Hypermedia Development: Design Patterns in Hypermedia (1999)
30. Erickson, T.: Lingua Francas for design: sacred places and pattern languages. In: Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques. pp. 357–368. ACM (2000)
31. Borchers, J.O.: A pattern approach to interaction design. In: Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques. pp. 369–378. ACM (2000)
32. Gabriel, P.R.: Patterns of Software: Tales from the Software Community. Oxford University Press (1998)
33. The Hillside Group: http://hillside.net/
34. Goodyear, P.: Educational design and networked learning: Patterns, pattern languages and design practice. Australas. J. Educ. Technol. 21, (2005)
35. Goodyear, P., Yang, D.F.: Patterns and pattern languages in educational design. In: Handbook of research on learning design and learning objects: Issues, applications, and technologies. pp. 167–187. IGI Global (2009)
36. Goodear, P., Retalis, S.: Learning, technology and design. In: Technology-enhanced Learning: Design Patterns and Pattern Languages. pp. 1–28. Sense Publishers (2010)
37. LDSE, https://sites.google.com/a/lkl.ac.uk/ldse/
38. Mor, Y., Mellar, H., Warburton, S., Winters, N.: Practical Design Patterns for Teaching and Learning with Technology. Springer (2014)
39. Bergin, J., Eckstein, J., Manns, M.L., Sharp, H., Chandler, J., Marquardt, K., Wallingford, E., Sipos, M., Völter, M.: Pedagogical Patterns: Advice for Educators. Joseph Bergin Software Tools (2012)
40. Martinez-Maldonado, R., Goodyear, P., Carvalho, L., Thompson, K., Hernandez-Leo, D., Dimitriadis, Y., Prieto, L.P., Wardak, D.: Supporting collaborative design activity in a multi-user digital design ecology. Comput. Hum. Behav. 71, 327–342 (2017)
41. Martinez-Maldonado, R., Goodyear, P.: CoCoDeS: Multi-device support for collocated collaborative learning design. In: Proceedings of the 28th Australian Computer-Human Interaction Conference. pp. 185–194. ACM, Australia (2016)
42. Goodyear, P., Dimitriadis, Y.: In medias res: reframing design for learning. Res. Learn. Technol. 21, 19909 (2013)

43. Ferguson, R.: Learning analytics: drivers, developments and challenges. Int. J. Technol. Enhanc. Learn. 4, 304–317 (2012)
44. Kennedy, G., Corrin, L., Lockyer, L., Dawson, S., Williams, D., Mulder, R., Khamis, S., Copeland, S., others: Completing the loop: returning learning analytics to teachers. Rhetor. Real. Crit. Perspect. Educ. Technol. Proc. Ascilite Duned. 436–440 (2014)
45. Dawson, S.: "Seeing" the learning community: An exploration of the development of a resource for monitoring online student networking. Br. J. Educ. Technol. 41, 736–752 (2010)
46. Shum, S.B., Ferguson, R.: Social learning analytics. Educ. Technol. Soc. 15, 3–26 (2012)
47. Wen, M., Yang, D., Rose, C.: Sentiment Analysis in MOOC Discussion Forums: What does it tell us? In: Educational Data Mining 2014 (2014)
48. De Liddo, A., Shum, S.B., Quinto, I., Bachler, M., Cannavacciuolo, L.: Discourse-centric learning analytics. In: Proceedings of the 1st international conference on learning analytics and knowledge. pp. 23–33. ACM (2011)