Game Learning Analytics is not *informagic*!

Ivan Perez-Colado, Cristina Alonso-Fernandez, Manuel Freire, Ivan Martinez-Ortiz, Baltasar Fernandez-Manjon

Dept. Software Engineering and Artificial Intelligence
Universidad Complutense de Madrid, Facultad de Informática
C/ Profesor Jose Garcia Santesmases, 9 28040 Madrid, Spain
{ivanjper, crisal03}@ucm.es {manuel.freire, imartinez, balta}@fdi.ucm.es

Abstract—**Game learning analytics has a great potential to provide insight and improve the use of games in different educational situations. However, it is necessary to clearly establish what the learner's requirements are and to set realistic expectations about the learning process and outcomes. Application of game learning analytics requires pedagogically informed policies that settle the learning goals and relate them to analysis and visualization; and a supporting infrastructure that provides the mechanism on top of which it is executed. Both concerns can be addressed separated: on the one hand, there is a Learning Analytics Model (LAM) which describes how the analysis is carried out, interpreted as learning, and presented to stakeholders; and on the other hand, an underlying analytics system that concentrates on performance, security, flexibility and generality. An important advantage of this separation is that it allows LAM authors to concentrate on their area of expertise, limiting their exposition to the actual mechanism used underneath. However, LAMs built for a single game fail to account for the frequent case where games and their analytics are aggregated into larger, overarching plots, games or courses. This work describes an extension to an existing game learning analytics system, used in RAGE and BEACONING H2020 projects, which manages multilevel analytics through improvements to both policy and mechanism; and introduces meta-Learning Analytic Models, which characterize learning in hierarchical structures.**

Keywords—learning analytics; serious games; learning analytics models; multi-scale games; xAPI;

I. INTRODUCTION

One of the main goals of Learning Analytics (LA) is to allow educators to better understand how learners apply domain knowledge when using an e-learning system, helping educators to improve educational outcomes [1]. However, learning analytics needs several inputs at different levels before it can provide such understanding. The definition of these inputs can be described in a Learning Analytics Model (LAM). In the particular case of e-learning using serious games (SGs), which allows more immersive and engaging experiences for students than traditional contents [2], LAMs have the potential for providing very good insights into learning – at a corresponding cost in terms of LAM complexity. However, it is necessary to clearly establish what the LAM requirements are and to set realistic expectations. Too often, users of analytics expect them to be presented with such insights based exclusively on "shallow" interaction data, assuming that the system can infer the educational game design and what exactly counts as "learning" on its own. This is not game learning analytics: it is an unrealistic expectation that we call *informagic*.

Although many stakeholders may benefit from the information obtained from game learning analytics (such as game developers, game designers, and researchers), this paper mostly focuses on teachers, due to their central role in guiding the learning process of students. We consider two main approaches towards providing analytics to teachers: first, when delivered at near-real time, analytics can allow teachers to
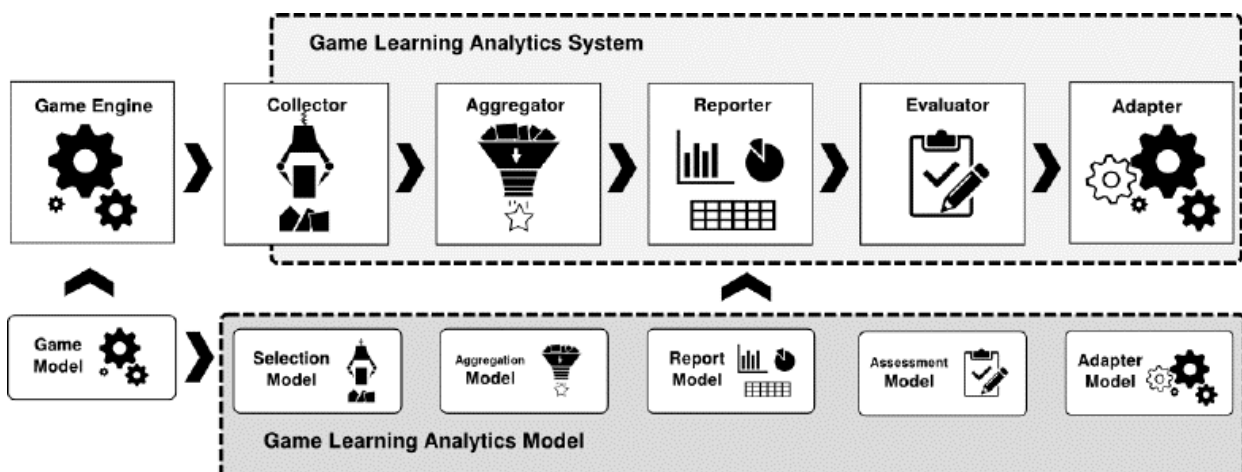

Figure 1. General architecture of a Game Learning Analytics Platform including the Learning Analytics Model and the Learning Analytics System [5].

maintain awareness of student actions in-game, and then help students that get stuck while playing [3]; and second, asynchronously, analytics can provide additional information to evaluate students and reflect or elicit comments on what was learnt during the game-play session [4]. Different stakeholders may be interested in other types of analytics; for example, managers could be interested in comparing the performance of groups against each other, rather than educational interventions. LAMs describe, for each stakeholder, what interaction information is gathered, how that interaction information is related to the educational design and how that correlation is interpreted and presented in the corresponding dashboards.

Educational games can be part of a more complex game learning approach. For example, a game may contain minigames; or a geolocalized game may launch different games depending on the location. Games that are not fully stand-alone, and are instead a part of a greater whole, require additional LAM-related information that describes how the users' learning and progress within each of these parts is to be considered into the context that contains them. The next section describes in greater detail the concept and advantages of LAMs, together with other formulations of the same basic idea. Section III then deals with meta-LAMs, that is, LAMs for mutliple levels (multilevel analytics); and describes changes to analytics systems that allow meta-LAMs to be supported. Section IV details changes to data collection, and Section V focuses on analysis and visualization for LAMs and meta-LAMs. Finally, Section VI provides conclusions and future work.

II. LEARNING ANALYTICS MODEL

As described in [5], a LAM provides the models on how information should be tracked, aggregated and reported to a Learning Analytics System (LAS). This relation inside a Game Learning Analytics platform is depicted in Figure 1. LAMs isolate learning analytics users from the implementation details of the underlying LAS. This allows both systems to evolve independently as long as the interface between the model and the system is well-defined and represents policy and mechanism respectively.

The LAS underlaying a serious game needs zero knowledge of game concepts or learning; and instead concentrates on blindly applying the LAMs for each of the games that report to it. This allows building generic analytics systems, where the allowable type of analytics is only constrained by the expressiveness of LAMs. Greater expressive power has the downside of increasing the difficulty of authoring in LAMs and supporting them at execution time. On the other hand, simplistic LAMs may prove too constraining to provide useful feedback. A reasonable compromise between both extremes is to provide a simple default LAM that delivers basic information on game completion and progress; and allow analytics' users to replace or extend it on a case-by-case basis. This way, (crude) analytics are available at zero cost, but more advanced insights can be obtained by investing additional effort where needed.

In [6], Chatti et al. propose a LAM-compatible reference model for learning analytics based on four dimensions: the data the system gathers, manages and uses for analysis (*what?*), the
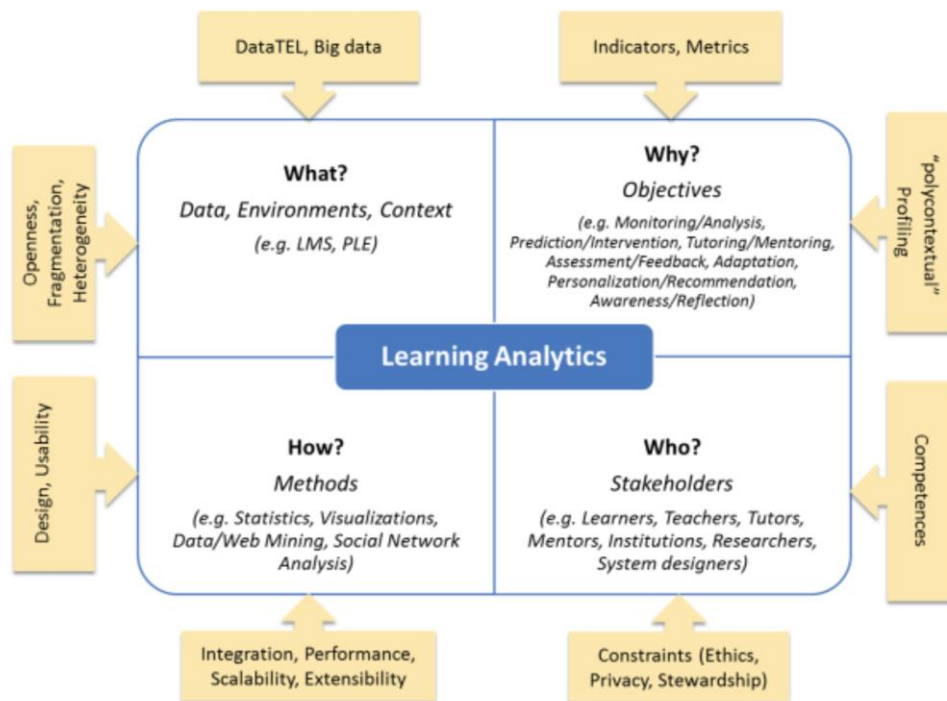

Figure 2. The LAM-compatible Learning Analytics Reference Model proposed in [6] describes four dimensions: data, objectives, methods and stakeholders.
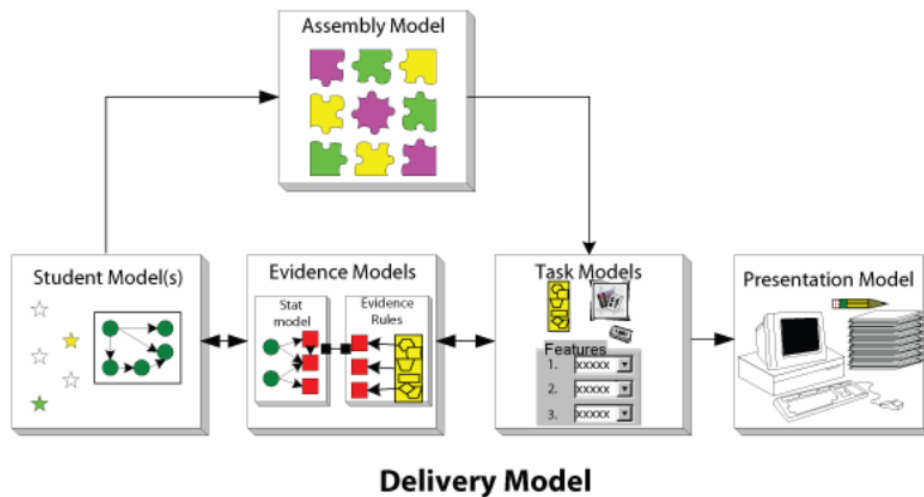
Figure 3. The Conceptual Assessment Framework layer of the evidence-centered assessment design described in [7] relies on student, evidence, task, assembly and presentation models.

stakeholders targeted by the analysis (*who?*), the reasons for the system to analyze the collected data (*why?*), and the way in which the analysis is performed on the collected data (*how?*). This reference model is illustrated in Figure 2. LAMs as defined in this paper attempt to answer 3 of these 4 questions, although motivating the specific answers for a LAM is welcome but non-mandatory: LAM authors must provide an executable definition of *what*, *how*, and *who* for the LAS to work on; the *why*, however, is only useful as documentation, since the LAS cannot interpret it. Of course, well-described LAMs will be much easier to maintain and adapt to evolving needs.

A related framework that focuses on describing educational assessment can be found in [7]. This evidence-centered assessment design (ECD) defines the information that should be tracked, how this tracking should occur and how the resulting data should be interpreted. Of particular interest is the Conceptual Assessment Framework (CAF) layer, depicted in Figure 3, which links the variables in the student model (*what are we measuring*), the environment in which students complete the task in the task models (*where do we measure it*), the observations to be made depending on the purposes and context in the evidence model (*how do we measure it*), and the assembly model that merges multiple tasks into a single feature (*how much do we need to measure it*) [7]. While LAMs may opt for simple approaches to many of CAF's models, any CAF model can be represented by a suitable LAM.

The interplay between the LAM, the underlying analytics system, and some of the main stakeholders is depicted in Figure 4, where blue boxes represent the mechanism (LAS), and red boxes represent the policy as described in the LAM. When students play games or mini-games, the tracker component embedded in those games sends traces (in our proposal, as xAPI statements) that describe actions and outcomes, which are then collected by the analytics system for storage, analysis and visualization.

Completely defining a LAM for a particular game (or mini-game) requires several decisions, depicted in Figure 5 together with the main stakeholder in charge of their definition and the expected results of each activity. The key decisions that a LAM must address are [8]:

1. **Learning goals** to be achieved in the game (e.g. specific knowledge, procedures, and tasks) must be defined before any other LA procedure. These learning goals will result in a specific learning design that, as pointed out by Bakharia et al. in [9], need to be linked to LA to provide it with a semantic frame of reference.
2. **Game goals** (e.g. tasks, levels) that correspond to the learning goals. The correspondence may not be bijective: a single learning goal may be furthered only once several game goals are completed (e.g. several levels to achieve some knowledge); while a single game goal may contribute to multiple learning goals (e.g. a level that teaches various skills). The set of game goals will result in a specific game design.
3. **Traces to be sent,** as defined by game developers based on the established learning and game designs. The information to be traced and sent by the game should therefore be necessary and sufficient to, once analyzed, inform of the degree to which the game goals are being met. In our proposal, traces must follow the xAPI-SG Model [10], as described later in detail (see the Data Collection Section). Notice that it is also possible to define traces with any information that may be of interest to stakeholders, even if this information is not specifically required to gain insight regarding learning goals.
4. **Analysis model**, defining how traces from step 3 should be analyzed and interpreted. In general, this will require keeping an updated estimation of the extent to which each learning goal is being met every time a new trace is received. The most valuable educational insight is obtained from analysis that take into account both the learning goals and how those goals were reflected into the game goals.
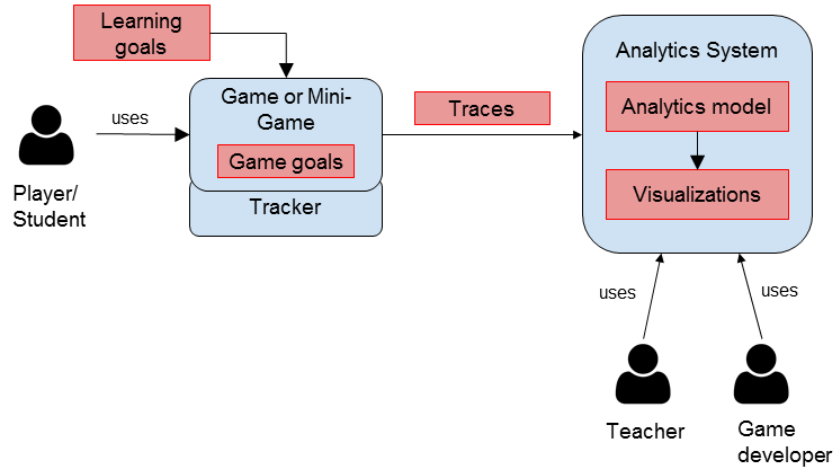
Figure 4. The Learning Analytics Model describes the information sent by the tracker, the analyses that will be performed and the visualizations later used to display results.

5. **Visualizations** that adequately represent the results of analysis, in the form of per-stakeholder dashboards. If a default LAM is being extended, then this LAM's default visualizations may be sufficient, or require only minor modification. Otherwise, entirely new visualizations may be required.

While analytics is usually performed after the game has been played, this does not need to be the case, since many systems can update their visualizations in near-real time. This opens the door to real-time alerts and warnings, which once configured can notify teachers of possible problems or interaction opportunities that arise during game-play, such as a learner becoming stuck in a particular level or advancing much more quickly than

expected. From a LAM perspective, real-time alerts and warnings can be seen as a specific type of visualizations which only present themselves in dashboards when triggered, and can be specified together with traditional always-on visualizations when designing those dashboards. In this sense, alerts and warnings focus on providing immediate information for teachers to act on during game-play time.

A. LAM example

To further clarify the concept of a LAM, this subsection describes some parts of the LAM used for the First Aid Game [11], designed to teach players between 12 and 14 years old first-aid techniques in three particular situations: chest pain, unconsciousness and choking. The following paragraphs
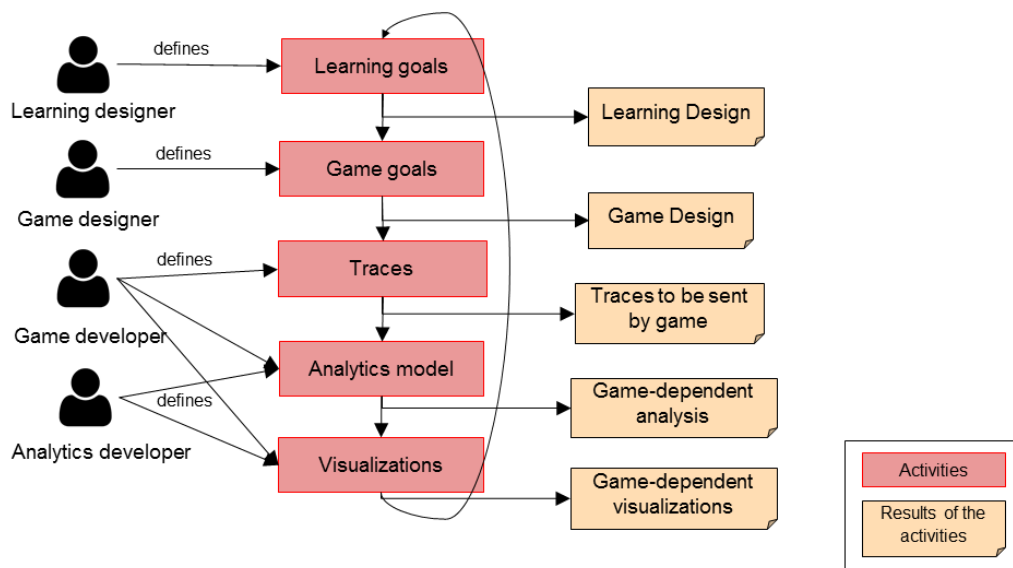

Figure 5. Learning Analytics Model steps, including the stakeholder in charge of each definition and the result of the activity carried out in each step.

describe a subset of the game's learning goals (LG), game goals (GG), traces to be sent, analyses and visualizations. This description covers only the chest pain situation, three of its main learning goals, and details of the consequent activities of the LAM that are of interest for those three learning goals in particular.

The main learning goal (LG1) of this game was to "*learn how to react in a first aid emergency situation*". There were also three learning goals related with each of the three game levels: for instance, for the first of them, the learning goal was to "*learn first aid techniques for chest pain situation.*" (LG2). Another lower-level learning goal was to "*learn the emergency phone-number*" (LG3). These three learning goals were reflected in three game goals: "*successfully complete all three levels of the game.*" (GG1, to satisfy LG1), "*successfully complete chest pain level.*" (GG2, to satisfy LG2) and "*successfully answer the question about the emergency number*" (GG3, to satisfy LG3).

Among the traces defined to be sent by the game, the ones that corresponded to the previous learning and game goals are those related to progress in each level (given as a real number from 0 to 1) and to answers provided for specific questions. Three types of traces were sent for progress for each level: the *initialized* trace at the beginning, the *progressed* trace with each progress from 0 to 1, and the *completed* trace when level is finished; including the level's score as an extension [3]. For specific questions, a *selected* trace was sent, together with the actual option that was selected, and whether it was correct or not. A more in-depth description of what *initialized*, *progressed* and *completed* means can be found in the Section on Data Collection.

The analysis model of these traces defined GG2 (and therefore, LG2) as achieved if the *completed* trace of the level had an extension with a score greater than or equal to five points; so that the progress of the whole game increased by a third of the total with each successfully completed level. GG1 (and therefore, LG1) was considered achieved when all three levels had been successfully completed. Finally, GG3 (and therefore, LG3) was achieved once the *selected* trace of the specific question had a successful result.

To visually communicate the results for LG1 and LG2 to teachers, the default visualization that reports progress in each of the three levels (*completables*) and in the complete game (also a *completable*), together with the scores obtained, are considered to be enough. The visualization that reports progress consists of a bar chart with all students in the x-axis and, for each of them, and in the y-axis, four bars representing their progress in each of the three levels and the complete game; with each bar ranging from empty (not started) to full (level or game completed. To communicate LG3, another bar chart showing the answers (y-axis) grouped by students (x-axis) to each question (individual bars within each student) was used.

To help teachers keep control of the class and help students having difficulty completing each level, a personalized warning with the message "*the user has failed the Chest Pain game mode*" was triggered whenever a student completed the level but failed it, to indicate that the knowledge had not yet been acquired. Another warning was set with the message "*the user has failed the Emergency Number question*" when the specific

question was answered incorrectly. Apart from those specific warnings, general warnings indicated which students had been inactive for over a given period of time, or could be stuck in a particular section could be easily enabled. In this particular case, students were detected as stuck when the teacher-specified expected level completion time was exceeded. A more robust "stuck" indication could have used the degree of deviation from the norm, once a sufficient number of gameplay sessions was collected.

III. META-LAM

When a game is composed of multiple games, their individual LAMs that focus on each of these games in isolation fails to describe the larger game as a whole. For example, a geolocalized game may involve launching different context-appropriate games depending on the location; and it makes perfect sense to ask for analytics on the overall progress in the overarching game. However, once several games are part of a larger aggregation, the problem of progress and completion as indicators arises: if there are several paths along the game, and some of them may require complex conditions that may or may not occur, it becomes very difficult to measure how much of the game remains to be completed. Analytics for composite games is also expected to include dashboard visualizations that show global information about one student, across several classes and/or games. To address the need for aggregating results, progress/completion calculation, and new aggregate-aware meaning, we have proposed the use of Learning Analytics Models for multi-scale games: meta-LAMs that stitch together the individual game LAMs into a larger whole. For example, returning to the geolocated game example, we can envision a main game that proposes tasks to players that require them to reach certain areas; which, when reached, cause different mini-games or activities to be launched. Each game or activity would retain its own LAM, and the meta-LAM would describe how these are to be joined together.

Any proposed meta-LAM, similarly to particular LAMs for games or mini-games, should define a set of learning goals to be achieved by the whole game, how they are going to be achieved by its different games or components, how the information is going to be traced and what analyses and visualizations should be performed on the data to have a complete understanding of the learning process in the complete platform. Once defined, the meta-LAM should allow access to essential information such as overall student progress in the general higher-level game.

For this meta-LAM to be correctly defined, both the hierarchy and the information flows must be clearly specified. The information flow includes transfer, aggregation and analyses that involve more than one level. For instance, a typical approach will be to have a tree structure where the general game is decomposed in several mini-games (as leaves) [12]. These leaves will then contribute towards the progress of their parent node as they are completed.

A. Meta-LAM structure proposal

Although the meta-LAM will depend on the structure of the hierarchy of games, a standard meta-LAM can be considered for certain general structures, and in particular games that are structured as simple hierarchies (trees). In the case of graph
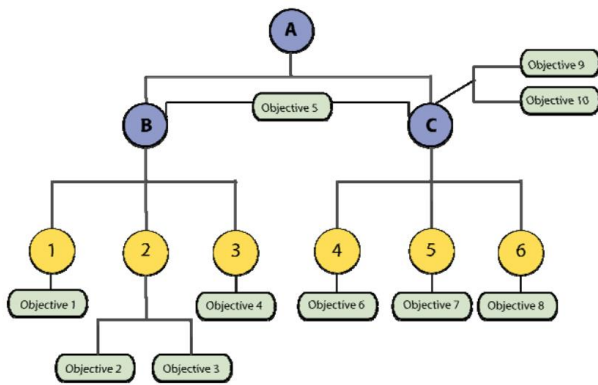
Figure 6. Activity Tree with Learning Objectives shared. Figure retrieved from ADL SCORM 2004 4th Edition Sequencing and Navigation (SN) specification [13].
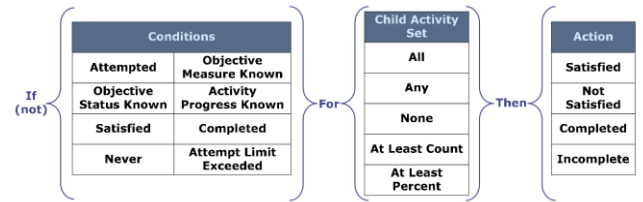

Figure 7. Rollup Rule Child Activity Set, Conditions and Actions. Figure retrieved from ADL SCORM 2004 4th Edition Sequencing and Navigation (SN) specification [13].

structures, a tree overlay could be defined to allow the use of this standard meta-LAM. However, meta-LAM should not be limited to this particular scenario, and be flexible enough to allow any reasonable model to be implemented. Notice that when it comes to systematizing LAMs for multi-scale games, we have found no standard or widely accepted model on the literature that covers this issue.

After an exhaustive revision of specifications used on similar fields like e-learning, we decided to build upon a greatly simplified version of the IMS Simple Sequencing specification as described in SCORM 2004 4th Edition Sequencing and Navigation (SN) [13] specification. SCORM Simple Sequencing SN defines an Activity Tree (AT) as the main hierarchical structure of learning activities, which may or may not correspond to the actual internal organization of activities. Clusters are defined as learnings activities with immediate sub-activities, and they can contain both leaf activities (which are not clusters) and other clusters.

Learning activities in SN (activities for short) are nodes in the AT. Activities have completion (with start and finish) and mastery conditions, and they can contain sub-activities to any depth. Any effort to complete an activity is called an attempt. Attempts may be suspended and later restarted, while abortion removes all attempt information. Learning Objectives have no specific semantics: they may refer to competencies, masteries, shared values, etc. There is no direct correspondence between activities and objectives: activities may have more than one objective associated and multiple activities may reference the same objective. Figure 6 shows an example AT with the learning activity "A" as root. Some learning objectives are also defined: all objectives are local to their associated activity, except objective 5 that is shared between activities "B" and "C" [13]. Some limit conditions may be stablished for activities to determine when they are not allowed to be delivered. In SN, the only mandatory condition is the Attempt Limit, which will correspond to the maximum number of attempts to complete an activity.

In our proposal, the AT will correspond to our meta-LAM, which is the highest level structure (i.e. the composed multi-scale game or the geolocalized-game both with mini-games). Activities in the AT will correspond to games, so that every activity will have an associated LAM as long as it has an associated learning content to track and analyze. If so, the associated LAM for those activities will define the conditions for its successful (or unsuccessful) completion. Notice that a single LAM may be sufficient to cover several games: it is very frequent to find related game activities, or even the same activity with slightly different configuration data being used in multiple places.

Learning Objectives, as defined in Simple Sequencing, could easily correspond to learning goals to be achieved via playing. As with activities and objectives in Simple Sequencing, there is no direct correspondence in our proposal between games and learning goals. Simple Sequencing defines *Rollup* as the process of passing information from children nodes to parent nodes in the AT. Rollup rules define how progress for cluster activities is to be evaluated, and consist of a set of child activities to be considered, conditions to be evaluated against them and actions to be taken, as depicted in Figure 7. By default, all children activities are included in parent rollup, unless they are not tracked, or do not contribute to rollup (not mentioned in any activity set). Leaves activities are not affected by rollup rules. Weighted combinations of the information from child activities are used to determine progress, completion, and objectives met.

For the meta-LAM definition, rollup rules are essential as they define how information from single activities (games) is to be aggregated up for general multi-scale games. Following the Simple Sequencing proposal, we again consider that parent game nodes could define their progress and objectives as linear weighted combination of the progress and objectives achieved in the corresponding children game nodes.

This hierarchy also allows for multi-level status storage. This is required if players start a multi-scale game that launches other games in turn (as sub-activities in the AT), as the general status of the player, across all levels, should be stored. Similarly, multi-game variables may also be required. If they are global in the AT, they can be used and changed across games. Otherwise, communication of variables between games will be required for consistency. That means that there should be a clear and unique way to identify what a specific user is doing in the different games.
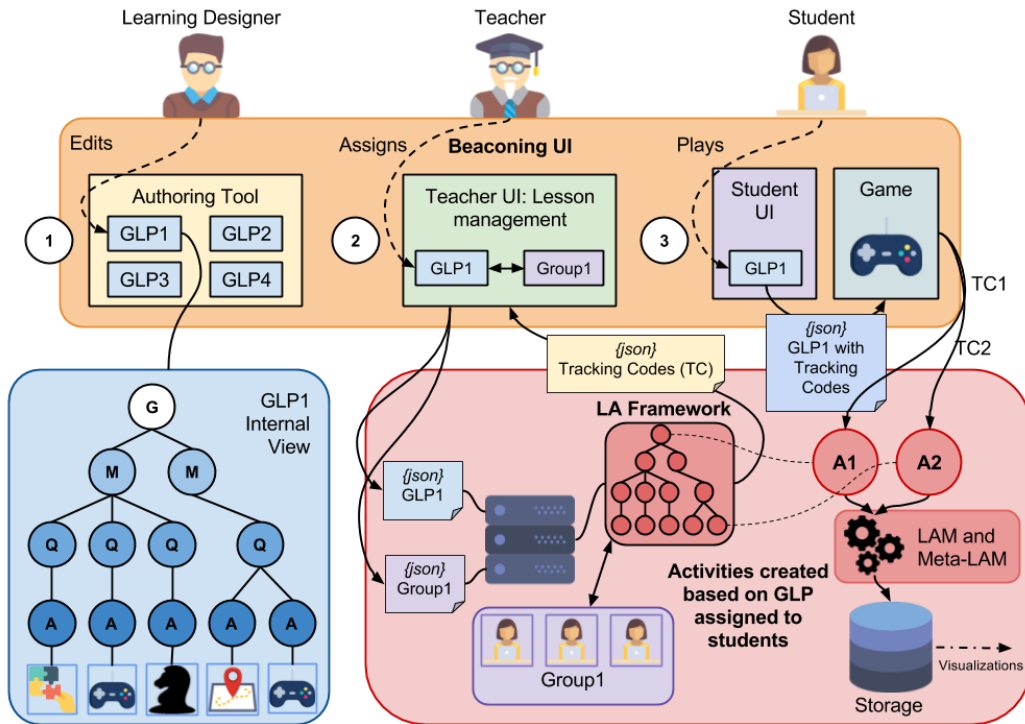
Figure 8. H2020 Beaconing Project structure proposal to apply LAM and meta-LAM using a model similar to the one proposed in SCORM Simple Sequencing SN. Game designers and data visualizations (dashboards) are omitted in this representation for simplicity.

The previously described structure for the meta-LAM is being considered for the case of the H2020 BEACONING Project, where a hierarchical structure of games and mini-games is required. In particular, learning designers define gamified lesson plans (GLP) that consist of missions, quests and activities. These activities may be games or mini-games. Figure 8 describes the first stages of the process: analytics definitions based on LAMs and meta-LAM, GLP creation and assignment to students, and students playing. For clarity, the role of game designers is being omitted in the figure.

IV. DATA COLLECTION

As defined in the previous chapters, the data collected from SGs is to be sent via traces for its later analysis and visualization. In our proposal, these traces must follow the xAPI standard. The Experience API (xAPI) is an e-learning standard used to track information from learning activities using *statements* composed of three main fields: an actor, a verb and an object. Additional fields may be included such as the context of the activity or its results.

Together with ADL [14], leaders of the community that created xAPI, the e-UCM Research Group developed the xAPI Serious Games Vocabulary (xAPI-SG) [15]. This profile, defined in detail in [10], defined concepts such as *completables* (e.g. tasks, quests or mini-games), *alternatives* and general *variables* to track interactions in the domain of SGs. It provides

a standard set of verbs (e.g. *initialized, progressed* and *completed* for *completables; selected* or *unlocked* for *alternatives*; etc.), activity types (e.g. *area, enemy, level, question,* etc.) and extensions (e.g. *progress*) general enough to cover the interactions of the player in most SGs but concrete enough to provide meaningful information of interest for the possible stakeholders. With its implementation in the xAPI standard, the interaction model provides a general, game-independent format for traces to model most of the players' interactions within a SG.

As required by the H2020 BEACONING project, an extension of the xAPI-SG model that vocabulary for geolocated games has been proposed, which includes verbs such as *entered,* objects such as *area* or *point-of-interest*, and an *orientation* extension that can be added to specific xAPI-SG statements.

While game developers using the xAPI-SG Profile can create their own trackers, e-UCM has also released a range of open-source tracker implementations for different platforms. They include: JavaScript[1], C#[2], Dot NET[3], Unity (adapted from the C# tracker)[4], Unity (specifically developed in Unity)[5] and LibGDX (Java) (outdated)[6].

V. ANALYSIS AND VISUALIZATION

This section briefly introduces the analytics system developed, as part of the H2020 RAGE Project, and tested,

---

[1] https://github.com/e-ucm/js-tracker
[2] https://github.com/e-ucm/csharp-tracker
[3] https://github.com/e-ucm/dotnet-tracker
[4] https://github.com/e-ucm/unity-tracker
[5] https://github.com/e-ucm/unity-tracker/tree/0.5.0
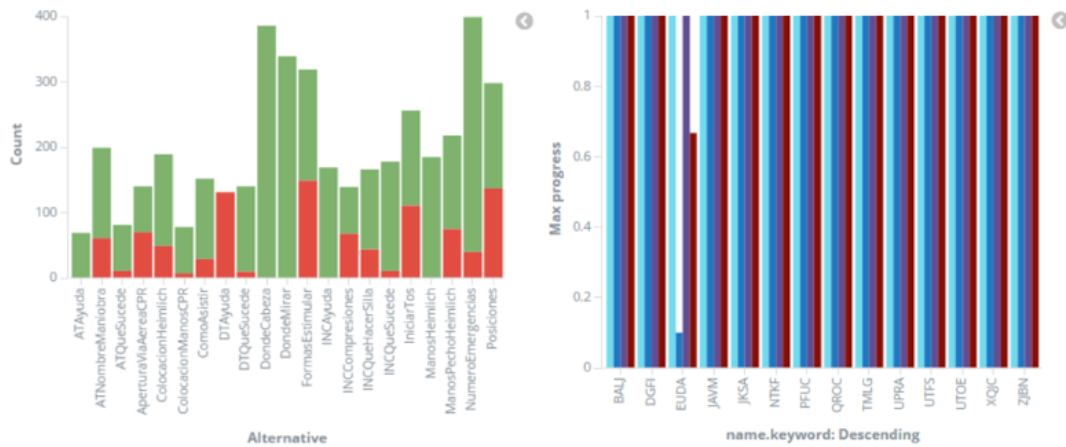[6] https://github.com/e-ucm/libgdx-tracker

Figure 9. Bar chart of number of correct and incorrect answers in each alternative (left) and bar chart with progress (with 1 representing 100%) in each level and in the complete game by each player (right). Names of players have been replaced with random identifiers.

extended and improved as part of the H2020 BEACONING Project. Full documentation of the current state of the system can be found at the Official GitHub Wiki Page[7]. The meta-LAM described in this work is, as of this writing, currently being implemented into this system, with validation soon to follow.

When it comes to analyses and visualizations, this LAS provides a fixed set of analyses and visualizations corresponding to the "default LAM", which is used for all new games until replaced or extended with a game-specific LAMs. While the default LAM is mainly focused on teachers, some of its default visualizations are targeted at game developers. The current set of analyses and visualizations for teachers provides information including scores, progress, alternatives selected, correct and incorrect answers, active sessions, and games started and completed; which are depicted in different types of visualizations, such as bar, line, or pie charts. Figure 9 shows two of these visualizations: number of correct and incorrect alternatives chosen by each player, and the maximum progress achieved by each player in each of 3 game levels and in the complete game; both for the First Aid Game example from Section II.

The previously defined set of analyses and visualizations is provided for both teachers and game developers. Meanwhile, students may require a different type of feedback that does not break the game flow and does not make them feel observed while they play (which may change their behavior). To avoid breaking the game flow of the students, the system provides an API that allows games to query analytics directly. Games wishing to use this API to provide in-game visualizations for students only have to make the corresponding calls to the analytics system; or, if they rely on the open-source tracker implementations described in section IV, simply request the information through the tracker itself. Finally, learning dashboards for students also present problems such as not providing recommended actions or focusing more on competition rather than on goal achievement [16]

The system allows game-specific LAMs for particular games. Their development must involve game developers, which can link in-game actions (mechanics) with game goals and to the information that is to be analyzed and displayed. As previously stated, the most valuable educational insight is obtained from analyses that take into account both the learning goals and how those learning goals are related to the game goals; and this is simply not possible for the default LAM, which must necessarily be generic. Note that game-specific LAMs do not need to address all of the steps depicted in Figure 5 to be useful; for example, choosing a visualization that makes player actions easier to relate to learning goals, without changing incoming data or how it is analyzed, can already provide significant improvements to insight as compared to a generic visualization.

Mirroring the distinction between the default LAM and game-specific LAMs, work is currently under way to provide a default meta-LAM for composite games, based on the structure proposed in Section III, and reflecting commonly used metrics such as progress or score; while particularly composite games will still be able to replace or extend it with their own specific meta-LAMs. These analyses must be aligned with the hierarchical structure of the game, and describe exactly how aggregations are to be performed between the different levels of the hierarchy. In the simple but common case of a tree structure, parent nodes may define their fields as the aggregation or average of their children's fields. However, other options are possible; and whichever is chosen, must be unambiguously described in order to correctly define analyses and visualizations at the meta-LAM level.

VI. Conclusions and Future Work

Learning Analytics Models encapsulate and describe how data from serious games is to be gathered, analyzed and displayed. When used correctly, they provide relevant information about the students' learning experience to teachers and other stakeholders; feedback that is key to guide the correct application of serious games in education. Effective LAMs should be developed based on the learning design, and their

---

[7] https://github.com/e-ucm/rage-analytics/wiki

development should ideally start even before that of the game itself, clearly establishing their requirements and their (realistic) expected outcomes. If the information from interactions obtained is limited and does not relate to the learning design, expecting analytics to provide insights into learning is to believe, as this paper's title states, in *informagic*.

Development of LAMs should follow a linear process where each step bases its development on previously established definitions and outputs, and preventing cases where late steps such as analysis or visualization are expected to provide learning insights from insufficient or context-free data.

While standard LAMs fit the needs of a single game, they fail to provide adequate insight in cases where games are part of more complex structures. For these cases, meta-Learning Analytics Models need to be defined to fulfill the needs of those structures. In our meta-LAM structure proposal, based on SCORM Simple Sequencing, games follow a tree structure where they can launch other mini-games, which can then launch further mini-games in a hierarchical fashion. Learning goals may be shared across different sub-games; and similarly, single sub-games may help achieve multiple learning goals. Aggregation from sub-games to their parent games consists of linear weighted combination of the fields of their children, which may include scores or progress.

Future work includes the improvement of the current default LAM, including both its default analyses and visualizations; and implementing and testing the proposed meta-LAM structure as part of current projects that require support for complex hierarchies of games and mini-games. Results of the assessment could also be used for adaptation in games, as authors have pointed out in [17].

Finally, it is also important to notice that the adoption of learning analytics can greatly benefit from its direct integration into game authoring tools that simplify costs and knowledge required for its application [18].

REFERENCES

[1]   D. Gašević, S. Dawson, and G. Siemens, "Let's not forget: Learning analytics are about learning," *TechTrends*, vol. 59, no. 1, pp. 64–71, 2015.

[2]   T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," *Comput. Educ.*, vol. 59, no. 2, pp. 661–686, Sep. 2012.

[3]   M. Liu, J. Lee, J. Kang, and S. Liu, "What We Can Learn from the Data: A Multiple-Case Study Examining Behavior Patterns by Students with Different Characteristics in Using a Serious Game," *Technol. Knowl. Learn.*, vol. 21, no. 1, pp. 33–57, Apr. 2016.

[4]   C. S. Loh, Y. Sheng, and D. Ifenthaler, *Serious Games Analytics*. Cham: Springer International Publishing, 2015.

[5]   M. Freire, Á. Serrano-Laguna, B. M. Iglesias, I. Martínez-Ortiz, P. Moreno-Ger, and B. Fernández-Manjón, "Game Learning Analytics: Learning Analytics for Serious Games," in *Learning, Design, and Technology*, Cham: Springer International Publishing, 2016, pp. 1–29.

[6]   M. A. Chatti *et al.*, "Learning Analytics: Challenges and Future Research Directions," *E-Learning Educ.*, no. 10, 2015.

[7]   R. Mislevy, "Evidence-Centered Assessment Design: Layers , Structures , and Terminology," *Princ. Assess. Des. Inq. Tech. Rep. 9*, vol. 9, no. July, p. 46, 2005.

[8]   M. Freire, I. Martinez-Ortiz, C. Alonso-Fernández, I. J. Perez-Colado, and B. Fernandez-Manjon, "D4.7 Game analytics and adaptation components reference implementation - H2020 BEACONING Project Deliverable." 2017.

[9]   A. Bakharia *et al.*, "A Conceptual Framework linking Learning Design with Learning Analytics," *Proc. LAK16 6th Int. Conf. Anal. Knowl. 2016*, no. April, pp. 329–338, 2016.

[10]   Á. Serrano-Laguna, I. Martínez-Ortiz, J. Haag, D. Regan, A. Johnson, and B. Fernández-Manjón, "Applying standards to systematize learning analytics in serious games," *Comput. Stand. Interfaces*, vol. 50, pp. 116–123, Feb. 2017.

[11]   e-ucm, "First Aid Game," 2012. [Online]. Available: http://first-aid-game.e-ucm.es/.

[12]   W. Westera, R. J. Nadolski, H. G. K. Hummel, and I. G. J. H. Wopereis, "Serious games for higher education: A framework for reducing design complexity," *J. Comput. Assist. Learn.*, vol. 24, no. 5, pp. 420–432, 2008.

[13]   ADL, "SCORM 2004 (4th Edition)," 2009. [Online]. Available: http://www.adlnet.org/adl-research/scorm/scorm-2004-4th-edition/.

[14]   ADL, "ADL." [Online]. Available: https://www.adlnet.gov. [Accessed: 22-Mar-2016].

[15]   eUCM Research Group, "xAPI Serious Games Profile," 2016. [Online]. Available: http://w3id.org/xapi/seriousgames. [Accessed: 24-Apr-2017].

[16]   I. Jivet, M. Scheffel, H. Drachsler, and M. Specht, "Awareness is not enough: Pitfalls of learning analytics dashboards in the educational practice," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10474 LNCS, pp. 82–96.

[17]   V. Shute, F. Ke, and L. Wang, "Assessment and Adaptation in Games," in *Instructional Techniques to Facilitate Learning and Motivation of Serious Games*, P. Wouters and H. van Oostendorp, Eds. Cham: Springer International Publishing, 2017, pp. 59–78.

[18]   I. Perez Colado, V. Perez Colado, I. Martínez-Ortiz, M. Freire, and B. Fernandez-Manjon, "uAdventure: The eAdventure reboot - Combining the experience of commercial gaming tools and tailored educational tools," in *IEEE Global Engineering Education Conference (EDUCON)*, 2017, pp. 1754–1761.