The Leverage™ Approach:

Designing Virtual Performance Spaces for Data Collection, Analysis and Adaptive Applications

Peter Jakl and Dr. David Gibson

Leverage is a highly scalable "big data" platform for acquiring granular data based on a user's interaction with any digital application. Whether the virtual experience of the application is an adaptive learning program, a game, a simulation, or an informational web site, Leverage applies the power and scale of cloud computing to create meaning from user data via high-resolution data capture, automated analyses and responsive deployment of digital resources. Beyond the statistical meaning of data in traditional educational contexts, Leverage has advanced tools to make behavioral, personality and learning style inferences an integral part of finding meaning from the values of attributes in virtual performance data. Leverage-powered adaptive assessment engines are driven by a scripting language that can draw immediate conclusions about any attribute using the user's full path of previous activity while also making correlations to a segmented population – ALL in real-time.

Top-down, meet bottom-up

In traditional database design, there is an imposed structure of data concerning how to organize it into something meaningful. The problem is that such top-down design is typically done *without* user performance data. Top-down thinking drives designers to organize a domain view of data into tables such as people, locations, stakeholders, transactions, and activities. The design process then defines attributes within each table, for example, attributes that might constitute a "person" in the database - a name, a birth date, a location. Typical relational database design produces numerous interacting tables and knits them all together before any user performs any real interaction with the application. Each table requires designers to anticipate attributes that might contribute to something meaningful for analysis, which introduces the designer's expert conceptions, biases and predilections about the structure of knowledge, performance, and analytics.

Leverage, in dramatic contrast, turns this process upside down by dynamically creating structure and organization of data based on what is directly meaningful *during the process of analysis*. The Leverage data system approach initially focuses on raw data elements or attributes and, more specifically, their values. Structuring, organizing and analyzing tasks are not only done later as user data accumulates but are also undertaken as constantly evolving processes in Leverage. This approach simplifies integration of Leverage into digital applications because the only guiding rule is to *capture user actions*. Leverage creates a timeline of activity that builds a personalized path through an application, and performs this at scale for millions of users and trillions of events. Finding and making connections within this high-resolution activity is what tells the story and supports inferences of behavior, personality and learning tendencies. The significance of those inferences is drawn from correlations based on what others have done, as well as on what the user has done, within the same set of interactions.

Leverage's ability to dynamically summarize actions from the bottom up replaces the process of creating database tables from the top down. We can think of summaries as encoded, as well as evolving, questions of interest to analysts and other users of the data. Leverage summaries constantly create and update *post-hoc structure* related to actual performance information of real users that help data scientists to focus on finding and expressing meaningful patterns and relationships. Since they can be built at any time after data collection has begun, the summaries also highlight how data organization decisions and statistical findings result from an analyst's evolving set of questions as well as the changing performance profiles of individuals and groups.

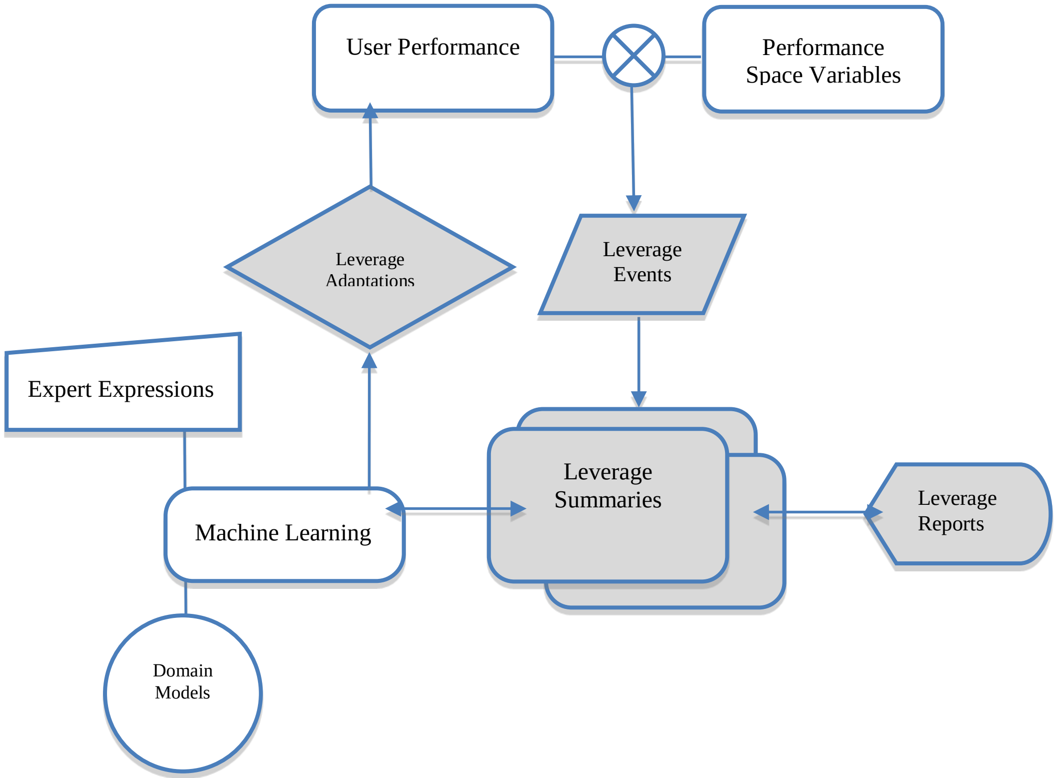

Figure 1. The Leverage Approach to Data Collection, Analysis and Adaptability

What replaces relational data tables in the design process?

Two things. First, prior to anyone using a Leverage-enabled adaptive application, there is *a list of users* simply consisting of basic registration information, which at a minimum is the user's email address and password. Second, *the structure of relevant potential actions* allowed by the interactive application is the only other pre-existing information. From the intersection of these two sources when a user plays the game, or uses the web site, or works with any Leverage-powered application, the meaning of the human performance information then arises from the data based on someone's actual use of the application.

The user table can be extended with new attributes at any time, which supports creating new cross sections through the data by filtering or grouping on attribute values. In educational applications, for example, adding new attributes such as teacher, grade, school, district and state, allows quick access to related group-based statistical summaries. Leverage has tools that support a seamless process of adding attributes and creating summaries that are integrated with applications. Leverage-powered applications can be created in any digital language or platform.

Real-Time Adaptive Assessment

Adaptability in an assessment or learning application has to be driven by knowledge acquired through data and used to make second-by-second decisions. Accumulated analyzed data then becomes actionable knowledge. Domain experts and other users of the data and results of an assessment need to trust the decision-making rules, methods and impacts of the system, and since these need to occur quickly, they need to trust that the machine's role in the system is accurately reflecting their judgment and making the same decision they would make, only faster and for millions of users on demand. Effective adaptability is thus a result of timeliness and transparency. The Leverage approach provides these as fundamental capabilities.

In the Leverage-powered adaptive assessment, experts can perform analyses at any time prior to, during and after an application is launched, and draw inferences and conclusions supported by evidence that, in the end, determine important features such as scales of proficiency and rules for making adaptive decisions. Moment-by-moment appraisals can be as simple as a static scoring and rule system (e.g. "If the student has scored a 10 on this section of the assessment, then provide the student with more practice.") or as dynamic and complex as desired when incorporating changing variables and thresholds for scores and rules and combining those with other sources of real-time input.

The components of adaptive assessment in Leverage outlined below show the full breadth of what is available to an expert community to formulate an assessment strategy:

1. *Raw data* demonstrating a user's individual responses, including high-resolution time-based samples of a performance (i.e. a performance pattern) as well as point data (i.e. an answer to a prompt).
2. *Application context,* including virtual environmental conditions, the context of other performers if needed, the specifics of the multimedia experience, and other factors that might influence a user's individual responses.
3. *Summaries* as described above that represent the individual and various user populations, including statistical summaries as well as qualitative re-representations and visualizations.
4. *Subsets and supersets of summaries* that cross-section, cross-tab, and data-mine the attributes
5. *Historical information* at individual levels and aggregated by extensions to the user table.

6. *Expert and domain knowledge* in structures that can have static as well as malleable content.

The mechanical processes to define clear assessments may vary, but all conclusions come in the form of rules and collections of rules. See for example (Holland & Reitman, 1978; Holland, 1995) for a discussion of the *subsumption architecture* for rule systems leading to machine inference.

- *"If a student response is correct for item A, apply score of 5"*
- *"If a student response is incorrect for item B but correct for item A, apply a score of 2"*
- *"If a student response is correct for item C, apply a score of 5. If student also responded correctly on items A and B, apply an additional score of 2"*
- *"Based on a student score of 75, student shows adequate proficiency"*
- *"Based on a student score of 25, student requires remediation"*

Psychometricians working with domain content experts can bring additional expertise and provide another level of interpretation.

- *"Students that score above 50 on items A-G and score below 75 on items H-M demonstrate higher proficiency in visual acuity"*
- *"Students that average a response time below 4 seconds on items A-D and score higher than 80 in the performance section demonstrate higher proficiency in mental math"*
- *"Students that fall above the statistical average of student responses on all items, that have special needs, demonstrate proficiency in comprehension"*

Conclusions such as these can be programmed as rule collections into Leverage so that each student's proficiency on all measurable attributes can be updated in real time based on complex algorithms as determined by experts. The Leverage approach to rule formation and processing enables a panel of experts to view student responses in real-time, create new analyses at varying levels of expertise and target a series of tasks to accelerate an individual or group's speed of learning by combining accumulated with dynamic knowledge.

Because Leverage constantly executes rules triggered by user activity, the application has the ability to construct a course of learning appropriate for each individual student while simultaneously documenting aggregated performances relevant to both individual and group histories. For example, Leverage is able to monitor and process large-scale raw data that represents a path of responses based on thinking time, actual response time, and (most uniquely!) changes in patterns of responses. Leverage can process statistical summaries simultaneously as a whole and as segmented by attribute, and compute in-the-moment machine decision outcomes that personalize the digital experience as well as update the rule systems controlling future experiences.

Leverage stores an application's rules in the cloud. The rules are downloaded by the client application either at the start of the application, after a section or 'on demand,' providing ultimate flexibility in rule use and updating. The benefit of this flexibility is that all rules can be refined with increased knowledge. The refinements can be prepared 'manually' as experts directly adjust rules or they can be programmed into the rules themselves as a comparison against shifts in statistical and clustered performance results. For example, rather than defining a rule to update a student proficiency with a hard value of 80 if the student responds correctly 8 out of 10 times, such a rule can be an incremental update (such as +3) if the student responds correctly 2 above the average of students in the same school district. These rules can be extremely complex in order to reflect a more realistic environment.

Leverage provides reporting on how often rules are executed, the kinds of user interactions that trigger their execution and the outcome of the rule performance. Leverage is also able to stream rules by priority and dependency. Although highly complex rule sets can be created, each individual rule begins with a simple starting point of a basic scored appraisal. This simplicity at the most basic level brings transparency and trust to domain experts who are responsible for translating what they know into rules, which Leverage then uses in real-time on a massive scale to drive adaptability on evidenced-based decisions.

New Psychometric Considerations

It seems natural to ask questions about data after it has been collected and processed, rather than pre-forming answers in the data by structuring it before users perform real actions in a virtual space. However, there are several references in the literature that argue for first planning a chain of evidentiary reasoning in order to have a clear focus for data collection, avoid collecting too much information, or collecting the wrong information needed to support the inferences that need to be made. So which approach is best? Which of the two approaches - top-down or bottom-up – is best when *designing*, *deploying* and *analyzing data* for virtual performance spaces? Does the best approach depend on whether we think of these as *evolving* infrastructures rather than *static* structures?

We claim is that it is sufficient to have users and a potentially relevant action structure collecting data at the most atomistic level (with or without a fully determined chain of evidence), in order to support all *evolving, adaptive* contexts: iterative design development, a continually updated deployment model needed for adaptive learning, and an evolving higher-level analysis of emergent phenomena. This claim implies that from a design standpoint, the action structure only needs to be *potentially relevant*, not fully detailed in the initial design of evidentiary claim rules, in relation to the inferences we wish to make concerning learning or performance. If our claim is supported, then the Leverage data system approach has already pioneered, and can now fully support, a new measurement paradigm for the digital age, when virtual performances, online assessments, and digital game-based learning are on the rise.

To explore these issues, we raise an initial set of questions about design, deployment and analysis to invite comments and dialog:

DESIGN

> *What makes a digital performance space potentially relevant to the purpose of an educative experience or a performance opportunity, as well as to the intersection of both purposes?*

By "potentially relevant" we mean that the virtual performance space has to either provide a learning opportunity or performance opportunity (or both) that would be of value, or fulfill a purpose, in some community. A learning opportunity helps someone acquire knowledge, skill or expertise, and a performance opportunity gives someone a chance to show what he or she knows and can do. We have in mind more complex performances than say, choosing an optional answer from a list, and more complex knowledge than say remembering or recognizing a right answer.

For example, one community might be "speakers of English," with a performance opportunity of "ordering lunch at a roadside diner," and the virtual performance space would provide a learning opportunity if appropriately provided feedback is intended to improve how someone orders lunch in English. Relevance is ultimately established by a chain of valid evidence that can be assessed by a performance expert of the community.

We claim that designers of such applications do not need a fully outlined sequence of relations in the evidentiary argument prior to collecting user performance data. In addition, whether the purpose of the application is primarily educative, primarily performance-oriented, or both, we claim that the timing of the construction of the evidentiary argument doesn't matter, as long as the space is potentially relevant to the performance. We keep an open mind to the possibility that a space created for one purpose, might potentially provide evidence of other valued skills and knowledge, and we maintain that these other purposes can be discovered after the space has been designed, constructed and used.

We reason that since the data created by the user interface is finite and bounded by the sensor net of the devices for user interactions, the space of potential performance data is, or can be approximated to, a finite space that can be mapped to a network of relationships. Even if the combinatorial possibilities of the space are so huge as to be practically intractable, the real performances won't be. So *the possibility space network holds all of the designed possibilities for performance* and can be analyzed with sufficient computing power and machine intelligence shaped over time by expert knowledge and user experience.

Individual trajectories and groups or bundles of similar paths are created when people perform and each path has a unique time-based history; for example, unique durations spent at points in the space, resources used, sequences of actions, and time stamps. Based on these trajectories, sub-nets representing any performance are relatable to externally validated performances. For example, as long as the space is non-trival and sufficiently complex, there will be a range of performances. Some people will perform badly and others will perform well according to the standards for relevance. We hold that *the construction and validation of these performance sub-nets is best determined after* *creation of the space and its use by people, rather than before and in the absence of user data*. If the subnets can be fully determined beforehand and in the absence of real performance data, then the space, we feel, probably does not support highly complex and subtle performances.

Note that the order of constructing the evidence chain (whether before or after virtual space construction) does not in either case guarantee targeting some pre-specified knowledge or skill; this is a matter for external validation. Those who advise pre-planning the space and its analysis structure based on domain knowledge in the absence of performance data have a point, for at a minimum the space must contain at least the potential to be relevant. We are simply cautioning against over-thinking the evidence chain without the aid of real performance data, because there is a danger of creating unnecessary variables and an overly rigid structure of the performance construct.

These dangers arise because a design phase disconnected from user performance can create *hypothetical* variables for which there will be no interesting or useful data connected to a future claim. For example, variables for the x,y,z space coordinates of a user can be defined whether or not the user will ever interact with the application in a way that relates to that spatial information. If spatial action and decision-making is NOT a part of the interaction mechanics of the space, then creating and collecting that data will NOT be useful to an analysis of learning, and probably only distantly related to performance if at all. So during design, it is better to *create variables only as needed for enabling user interactions*, (and not some pre-determined theory of the data, or on the off-chance that the variable might be useful to a future analysis). The variable list in this case will be a near-minimal set needed to characterize performance trajectories as well as make reasonable estimates and inferences about the user performance data in the context of this virtual performance space. This will be true, we believe, whether or not the space has been effectively designed with a clear purpose of educating, prompting performance, or both. In contrast to design-first and domain-only planning, when user data is engaged, variable values and even emergent variables are empirical rather than hypothetical.

When designing for prompting performance and without concern for educating the user (e.g. for a summative assessment or a practice environment intended to observe performance without feedback to the user), then a relatively static model of performance is adequate, but even in this case, the model needs to be informed and shaped by actual past user performances rather than a pre-existing conception of the domain alone. The domain model can change afterwards but should not evolve during an individual performance; otherwise there will be a shifting standard for establishing relevance and validity. But when designing for an educative purpose, the model can and should change during interaction with the user as *all user actions imply a need to update the model*, even when those actions exactly reproduce an existing trajectory - a remote possibility in a complex application. An application that combines both educative and performance purposes has to decide moment-to-moment, in a kind of dynamic dance, whether the immediate analytic need is serving one or the other purpose.

Beyond these considerations, the discipline-specific content involved in either an educative or performance assessment purpose places additional external requirements on the design (e.g. thresholds of validity, realism, effectiveness), but these are not constitutive for the data collection and analysis design. These external requirements, (e.g. does the space represent something in the real world, does it provide an “opportunity to learn” or an “opportunity to perform” that some expert community validates?) while placing a top-down constraint on the virtual performance space design, do not constrain the bottom-up approach to data gathering and analysis.

DEPLOYMENT

> *What are the data collection and analysis issues if an adaptive application goes beyond providing alternatives to users and adapts itself to changing circumstances, including altering its structure over time as new theories and performance data arises?*

This question draws a contrast between so-called “adaptive” applications, which are primarily decision-tree structures that have pre-set alternative pathways selected based on domain knowledge and past user interactions and data, and applications that may do that but also *automatically evolve over time*. The former applications can be said to personalize the digital experience based on pre-existing models of variation in the population, where the latter applications also *evolve the models* based on user input including experts working with the system as well as user performances. One challenge to statistically based psychometrics is the problem of norms that arises when underlying models evolve. Norms assume that the models remain constant, but what role do norms play when models are evolving, and how does the change in the role of norms impact the educative versus performance versus combined digital application purpose?

We argue that when the purpose is educative, both the individual and group norm can evolve as rapidly as naturally occurs with each new learning and performance trajectory, because the main metric for change is the past, not a current comparison status. A challenge in dealing with past performances is to how to choose an appropriate sliding time-window for comparison. However, at the same time, computing a current status in comparison to some group of users or a model of the domain is also useful in an educative application in order to select a pre-set alternative pathway, so in this case, the challenge is the choice of group for comparison. The domain model can change as long as all the group statistics change with it, because the objective of performance improvement is not to compare against some benchmark but to offer helpful scaffolding advice. We also maintain that it is not important in low-stakes educative applications to constrain the group selection process to any particular point in time or any particular kind of group similarity. This is because, regardless of the selection process, so long as the adaptive advice given to the user propels their trajectory toward some sub-net that is closer to the ideal model (which itself can be evolving even during the process of this user’s interactions with the application) then the goal of educating toward better performance has been met. In fact, the closer the selected alternative path is to the current performance (rather than to the ideal performance), the more likely the user has the capacity to meet the new objectives without much scaffolding.

When the purpose is performance assessment, then the traditional role of norms of a comparison group remains and the individual's own past performance is only of secondary interest. Using the individual's past performance might be useful in estimating learning gain, but the primary objective is to classify the performer within some appropriate population of other performers, or against the domain knowledge itself. In this case, the prior structure of the domain is important and evidence claims that refer to it do need to be settled ahead of the performance and then held constant for some period of time if a comparison with peers is desired.

ANALYSIS

> *How does order matter in terms of top-down and bottom up approaches to data analysis? What are the barriers and facilitators when undertaking a top-down analysis phase AFTER the construction of a virtual performance space?*

When analyzing from the bottom-up *data mining* and *machine learning* techniques are used to first discover patterns and the rules that explain those patterns. When analyzing from the top-down, *a model or hypothesis* guides the creation of *statistical tests* that can potentially validate the model.

A primary barrier to undertaking an analysis after the creation of a virtual performance space is the possibility that the space was created with few potentially relevant user actions for making useful inferences. Even if the domain model was well determined, if the space of possible performance actions does not intersect with the domain model – and this is often left to programmers who may not be domain experts - there will be no point of reference for understanding the data. Machine learning and data mining techniques will find clusters, rules and relationships, but those will not be immediately meaningful to the analysis without an understanding of the middle ground between the bottom-up data and an expert's view of performance in the virtual space. Domain experts, on the other hand, may not have a view of "performance in this space" since they are not programmers, but they do know what should matter and not in a real-world analog of the performance. This is where the ECD approach is helpful.

A second barrier for an analyst is thus not having a mid- to high-level view of the clusters and dynamic patterns of atomistic actions that an external expert would count as a valid, relevant performance. Here, the ECD approach provides helpful bridging concepts that structure the search for meaning in the performance data. However, since experts do not generally have a view of the atomistic level of data created by a virtual performance (e.g. the microsecond interactions of an action), it seems most unlikely to us to expect that a full ECD evidentiary chain can be produced ahead of time.

In the Leverage approach, there are two places for mid-level analytic constructs to be created: complex *Events* that aggregate multiple atomistic events and *Summaries* that capture more complex data manipulations and dynamically report on the data subsets formed by sorting, grouping, and joining related variables. These constructs can be created both before and during deployment and can be updated by both domain experts and other users of the data as well as machine learning methods (Figure 1).

The "code script" for analysis is thus a living and evolving documentation of the relationship of users of the application that includes all decision-makers, both performers and observers (e.g. students, teachers, administrators, parents). With any reasonable design effort that attempts to provide an authentic virtual analog of some real-world interaction, once users have begun to interact with that application, the events generated are then automatically summarized and reported and those insights plus any new insights arriving from experts interacting to create new summaries and reports, along with machine learning algorithms that uncover emerging patterns and relationships, and the domain knowledge representation already encoded in the virtual space are all considered input for adaptive decision-making. Adaptive changes to the program can then occur for the user as well as for the entire event-driven system (Figure 1).

SCALABILITY

> *Suggesting highly granular data collection is easy, but is it feasible? What limitations are imposed on such a platform?*

Although this document is not intended to be an under-the-hood technical reference to Leverage, the issue of scalability is important to address, because we are making lofty and perhaps remarkable claims about collecting and storing highly granular data at scale, processing the data into sets of statistical summaries with multiple cross-sections, and then dynamically referencing this knowledge in order to produce outcomes that can be used for adaptive applications and assessments.

Leverage has evolved from technology developed for the U.S. Army to support *America's Army 3.0*, a "AAA game title" (i.e. a title developed by a large studio, funded by a massive budget), which is used by as many as 25,000 concurrent users generating 10,000 raw data events per second that triggers the execution of over 1,000 appraisal rules per second. The application is a great example of adaptive assessment since the scoring model is based on the Army's seven core values; higher-order knowledge, skills and attitudes that include loyalty, duty, respect, selfless service, honor, integrity and personal courage.

Subject matter experts from the Army evaluated user-generated in-game actions (e.g. accuracy, adhering to mission objectives, skill proficiency from training) and assisted in defining Leverage rules that adjusted value representations for the Army core values. Integrity or respect, for example, was earned (or penalized) based on a series of actions that might have involved long periods of time and many actions in a diversity of contexts.

The data approach described in this document tells the big-picture story of how Leverage persists these action records so that, once connected, a Leverage rule could apply the proper appraisal and make an appropriate determination, action or adaptation to the application *while running in real-time*. Adaptations to the individual user's experience are then made based on the composite profile of a virtual soldier and his or her adherence to the core values. The player's virtual performance is then accelerated or remediated to reflect the consequences of his or her actions according to the requirements implied for additional training and practice.

Our goal here is to translate this experience for teams of people who are interested in building adaptive digital applications and assessments powered by Leverage.

Invitation

We invite you to learn more about Leverage and our overall methodology. If you are part of a team of people exploring or actively building adaptive applications and assessments, we hope you'll be in touch to inform us of your efforts and to learn more.

Bios

Peter Jakl is Chief Pragmatist and President at Pragmatic. He has used his 32 professional years in technology to make effective use of data in a variety of business sectors, with an emphasis on inference and predictive modeling. His chief motivation is to hire amazingly talented people and create a working environment to achieve.

Dr. David Gibson serves as Chief Scientist overseeing and guiding ongoing research and development of the platform. His research and publications include work on complex systems analysis and modeling of education, web applications and the future of learning, and the use of technology to personalize education. Dr. Gibson's books *include Games and Simulations in Online Learning*, which outlines the potential for games and simulation-based learning, and *Digital Simulations for Improving Education*, which explores cognitive modeling, design and implementation.